INTERNATIONAL STANDARD

ISO/IEC 19763-12

First edition
2015-03-15

Information technology — Metamodel framework for interoperability (MFI) —

Part 12: Metamodel for information model registration

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

Partie 12: Métamodèle pour l'enregistrement du modèle d'information

Reference number
ISO/IEC 19763-12:2015(E)

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

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.

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

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

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

For an explanation on the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the WTO principles in the Technical Barriers to Trade (TBT) see the following URL: Foreword - Supplementary information

The committee responsible for this document is ISO/IEC JTC 1, *Information technology*, SC 32, *Data management and interchange*.

ISO/IEC 19763 consists of the following parts, under the general title *Information technology — Metamodel framework for interoperability (MFI)*:

— *Part 1: Framework*

— *Part 3: Metamodel for ontology registration*

— *Part 5: Metamodel for process model registration*

— *Part 6: Registry Summary*

— *Part 10: MFI Core model and basic mapping*

— *Part 12: Metamodel for information model registration*

The following parts are under preparation:

— *Part 7: Metamodel for service registration*

— *Part 8: Metamodel for role and goal registration*

— *Part 9: On demand model selection*

— *Part 13: Metamodel for forms registration*

Introduction

There is an increasing demand for systems to interoperate by exchanging data. For these data exchanges to be meaningful, it is essential that the business information requirements that are met by the data stored in these systems are understood so that suitable data exchange mechanisms can be developed.

Business information requirements, including the semantic meaning of the information, are often represented by information models before the databases that are an integral part of the systems are designed. These models are often called logical models. The subsequent design of the database structure can also be considered to be another form of information model.

Where there is an overlap of the universe of discourse of two systems, the information models for these two systems can be registered using the facilities specified by this part of ISO/IEC 19763. The mappings between these two models can then be registered using the facilities specified by ISO/IEC 19763-10. An interface between the two systems can then be designed, enabling the two systems to interoperate by exchanging information.

Information technology — Metamodel framework for interoperability (MFI) —

Part 12: Metamodel for information model registration

1 Scope

The primary purpose of the multipart standard ISO/IEC 19763 is to specify a metamodel framework for interoperability. This part of ISO/IEC 19763 specifies a metamodel for registering information models. This metamodel was developed taking into account two distinct types of information models such as

— those that are used to document the information requirements of a particular area of interest, and

— those that represent the structure of a database which are often expressed using a Database Definition Language (DDL).

Information models that represent information requirements can be developed using a number of different common diagramming techniques and notations. The metamodel specified in this part of ISO/IEC 19763 was developed to cover the registration of models expressed using the following techniques and notations:

— Express-G, an ISO standard entity-relationship modelling notation, as described in Reference [8];

— IDEF1X, a US Federal standard entity-relationship modelling notation, as described in References [6] and [12];

— the entity-relationship modelling notation first developed by Harry Ellis and Richard Barker and later adopted by Oracle for its CASE*Method and by the UK's CCTA for SSADM (Structured Systems Analysis and Design Method), as described in Reference [2];

— the UML Class Diagram notation, as described in References [13] and [14];

— the original entity-relationship modelling notation proposed by Peter Chen, as described in Reference [4];

— the Information Engineering entity-relationship modelling notation, as described in Reference [11].

It is understood that these selected techniques represent all of the essential features of all information modelling techniques used to represent information requirements.

The registration of information models that represent the structure of a database is limited in the metamodel specified in this part of ISO/IEC 19763 to those database structures that conform to the Core SQL specification. Core SQL is the set of features defined in the conformance requirements specified in ISO/IEC 9075-2 and ISO/IEC 9075-11.

The registration of information models that are expressed using notations such as Object Role Modeling (ORM) and "Natural language Information Analysis Method" (NIAM), collectively known as fact-based models, is out of scope for this part of ISO/IEC 19763.

2 Conformance

2.1 General

An implementation claiming conformance with this part of ISO/IEC 19763 shall support the metamodel specified in Clause 5 depending on the degree of conformance as described below.

2.2 Degree of conformance

2.2.1 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 19763 describes specifications that promote interoperability. Extensions are motivated by the needs of the users, vendors, institutions, and industries, but are not specified by this part of ISO/IEC 19763.

A strictly conforming implementation could be limited in usefulness, but is maximally interoperable with respect to this part of ISO/IEC 19763. A conforming implementation can be more useful, but could be less interoperable with respect to this part of ISO/IEC 19763.

2.2.2 Strictly conforming implementation

A strictly conforming implementation

a) shall support the metamodel specified in Clause 5, and

b) shall not use, test, access, or probe for any extension features nor extensions to the metamodel specified in Clause 5.

2.2.3 Conforming implementation

A conforming implementation

a) shall support the metamodel specified in Clause 5, and

b) as permitted by the implementation, can use, test, access, or probe for any extension features or extensions to the metamodel specified in Clause 5.

NOTE 1 All strictly conforming implementations are also conforming implementations.

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

2.3 Implementation Conformance Statement (ICS)

An implementation claiming conformance with this part of ISO/IEC 19763 shall include an Implementation Conformance Statement stating

a) whether it is a strictly conforming implementation (2.2.2) or a conforming implementation (2.2.3), and

b) what extensions, if any, are supported or used if it is a conforming implementation.

3 Normative references

The following documents, in whole or in part, are normatively referenced in this document and are indispensable for its application. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

NOTE One or more terms and definitions of the referenced International Standards listed below are used in Clause 4 Terms and definitions.

ISO/IEC 9075-1:2011, *Information technology — Database languages — SQL — Part 1: Framework (SQL/Framework)*

ISO/IEC 9075-2:2011, *Information technology — Database languages — SQL — Part 2: Foundation (SQL/Foundation)*

ISO/IEC 11179-3, *Information technology — Metadata registries (MDR) — Part 3: Registry metamodel and basic attributes*

ISO/IEC 19763-10, *Information technology — Metamodel framework for interoperability (MFI) — Part 10: MFI Core model and basic mapping*

4 Terms, definitions, and abbreviated terms

4.1 Terms and definitions

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

4.2 Terms for concepts used in this part of 19763

4.2.1
aggregation
special form of *relationship* (4.2.37) that specifies a whole-part relationship between the aggregate (whole) and a component part

Note 1 to entry: Adapted from ISO/TS 19103:2005, 4.2.2.

4.2.2
association
semantic *relationship* (4.2.37) between two *object classes* (4.2.35)

Note 1 to entry: Adapted from ISO/IEC 11179-3:2013.

4.2.3
associative entity type
information model element (4.2.25) that has both *relationship* (4.2.37) and *entity type* (4.2.21) properties

Note 1 to entry: May also be known as an association class.

Note 2 to entry: An associative entity type can be seen as a relationship that also has entity type properties or as an entity type that also has relationship properties.

Note 3 to entry: Adapted from ISO/IEC 19501:2005, 5.46.1.

4.2.4
attribute
named characteristic of an *entity type* (4.2.21) whose values serve to qualify, identify, classify, quantify or express the state of an instance of an entity type

4.2.5
attribute unique identifier element
unique identifier element (4.2.45) that is a statement that a particular *attribute* (4.2.4) is part of a particular *unique identifier* (4.2.44)

4.2.6
cardinality
number of elements in a *collection* (4.2.8)

Note 1 to entry: Adapted from ISO/IEC 9075-2:2011.

4.2.7
catalog
named *collection* (4.2.8) of *schemas* (4.2.42) in a database environment

Note 1 to entry: Within this part, this definition applies only to the registration of database structures that conform to the SQL Core specification as described in ISO/IEC 9075-2:2011.

Note 2 to entry: Adapted from ISO/IEC 9075-1:2011.

4.2.8
collection
aggregation of similar *objects* (4.2.34)

EXAMPLE set, bag (or multiset), list, array

Note 1 to entry: Adapted from ISO/HL7 21731:2006.

4.2.9
column
component of a *table* (4.2.43) that is a *collection* (4.2.8) of values all of the same defined *data type* (4.2.11)

Note 1 to entry: Within this part, this definition applies only to the registration of database structures that conform to the SQL Core specification as described in ISO/IEC 9075-2:2011.

4.2.10
composition
form of *aggregation* (4.2.1) which requires that a part instance be included in, at most, one composite at a time and that the composite object is responsible for the creation and destruction of the parts; composition may be recursive

Note 1 to entry: Adapted from ISO/IEC 19501:2005, 5.48.1.

4.2.11
data type
set of representable values

[SOURCE: ISO/IEC 9075-1:2011, 3.1.1.4]

4.2.12
described domain
domain (4.2.15) that is specified by a *description* (4.2.13) or specification such as a rule, a procedure, or a range (i.e. interval)

Note 1 to entry: May also be known as a non-enumerated domain or a continuous domain.

Note 2 to entry: Adapted from ISO/IEC 11179-3:2013.

4.2.13
description
property of an *information model element* (4.2.25) that is a statement explaining the significance of this information model element to the business and/or organisation that is the subject of this *information model* (4.2.24)

Note 1 to entry: May also be known as a significance statement.

4.2.14
diagram
technical document showing part of an *information model* (4.2.24) using graphical symbols

Note 1 to entry: A model can consist of one or more diagrams.

Note 2 to entry: Adapted from ISO 29845:2011.

4.2.15
domain
collection (4.2.8) of values from which the instance of an *attribute* (4.2.4) must take its value

Note 1 to entry: A domain provides a set of business validation rules, format constraints, and other properties for one or more attributes.

Note 2 to entry: The term domain is used in this part of ISO/IEC 19763 purely in the sense that the term is used in modelling information requirements using techniques such as entity-relationship modelling and object oriented modelling. The term should not be confused with its use in contexts such as "business domain" and "domain of discourse".

4.2.16
entity
concrete or abstract thing that exists, did exist, or might exist about which information may need to be held in support of business operations

Note 1 to entry: Some *information modelling methods* (4.2.27) use entity for their main information modelling construct, whilst others use *entity type* (4.2.21); in this part of ISO/IEC 19763 the term entity type is used in preference to entity.

Note 2 to entry: Can also be known as an entity instance (in those information modelling methods that use entity for their main information modelling construct) or an *object* (4.2.34) (in those information modelling methods that use *object class* (4.2.35) for their main information modelling construct).

Note 3 to entry: Adapted from ISO/IEC 11179-3:2013.

4.2.17
entity relationship model
information model (4.2.24) based on *entity types* (4.2.21) and their *attributes* (4.2.4) and *relationships* (4.2.37)

4.2.18
entity role
role that an *entity type* (4.2.21) is playing in a *relationship* (4.2.37)

4.2.19
entity specialisation hierarchy
means by which instances of an *entity type* (4.2.21) can be classified or specialised

Note 1 to entry: Can also be known as an entity generalisation hierarchy, an entity subtype hierarchy, an entity type classification or an entity classification.

4.2.20
entity subtype
subset of the instances of an *entity type* (4.2.21), known as the supertype, that share common *attributes* (4.2.4) and/or *relationships* (4.2.37) distinct from other subsets

Note 1 to entry: Can also be known as a subtype, an object subclass, or a subclass.

4.2.21
entity type
set of characteristics common to a *collection* (4.2.8) of *entities* (4.2.16) that are instances of the type

Note 1 to entry: Some *information modelling methods* (4.2.27) use entity type for their main information modelling construct, whilst others use entity. In this part of ISO/IEC 19763, the term entity type is used in preference to entity.

Note 2 to entry: Can also be known as an entity (in those information modelling methods that use entity for their main information modelling construct) or an *object class* (4.2.35) (in those information modelling methods that use object class for their main information modelling construct).

4.2.22
enumerated domain
domain (4.2.15) that is specified by a list of all its *valid values* (4.2.46)

Note 1 to entry: Can also be known as a discrete domain.

Note 2 to entry: Adapted from ISO/IEC 11179-3:2013.

4.2.23
foreign key attribute
key attribute (4.2.28) whose value contributes in some way to the identification of the one related instance of that *entity type* (4.2.21) involved in the associated *relationship end* (4.2.37)

Note 1 to entry: A foreign key attribute provides or contributes to an alternative representation of the *relationship* (4.2.37) concerned. Its value must be drawn from the *domain* (4.2.15) of the corresponding key attribute of the related entity type.

4.2.24
information model
graphical and textual representation of *entities* (4.2.16) and the *relationships* (4.2.37) between them

Note 1 to entry: Can also be known as a data model, a conceptual data model, a logical data model, an *entity relationship model* (4.2.17), an object class diagram, or a database definition.

4.2.25
information model element
element of an *information model* (4.2.24) that can be represented graphically and/or textually

Note 1 to entry: Typical information model elements are *entity types* (4.2.21), *relationship ends* (4.2.37), and *unique identifiers* (4.2.44).

4.2.26
information modelling language
language or notation that is used to model information requirements in an *information model* (4.2.24)

4.2.27
information modelling method
approach to developing an *information model* (4.2.24) using a particular *information modelling language* (4.2.26)

4.2.28
key attribute
attribute (4.2.4) whose value contributes in some way to the identification of individual instances of the host *entity type* (4.2.21) or of some related entity type

4.2.29
link phrase
statement that explains the nature, expressed in business terms, of a *relationship* (4.2.37) from the perspective of one of the associated *entity types* (4.2.21)

4.2.30
maximum cardinality
statement of the maximum number of elements that can exist in a *collection* (4.2.8)

4.2.31
minimum cardinality
statement of the minimum number of elements that can exist in a *collection* (4.2.8)

4.2.32
native key attribute
key attribute (4.2.28) whose value contributes in some way to the identification of individual instances of the host *entity type* (4.2.21)

4.2.33
non-key attribute
attribute (4.2.4) that is not the unique identifier or an element of a composite *unique identifier* (4.2.44) of an *entity type* (4.2.21), or whose value is fully independent of all *relationships* (4.2.37) or other *attributes* (4.2.4)

4.2.34
object
anything perceivable or conceivable

Note 1 to entry: Adapted from ISO 1087-1:2000, 3.1.1.

4.2.35
object class
description (4.2.13) of a set of *objects* (4.2.34) that share the same *attributes* (4.2.4), operations, methods, *associations* (4.2.2), and semantics

Note 1 to entry: Adapted from ISO/IEC 11179-3:2013, 3.1.5.

4.2.36
object class model
information model (4.2.24) based on *object classes* (4.2.35) and their *attributes* (4.2.4) and *associations* (4.2.2)

4.2.37
relationship
set of characteristics common to a *collection* (4.2.8) of connections between instances of two or more *entity types* (4.2.21), or between instances of one entity type and other instances of the same entity type

Note 1 to entry: Can also be known as an *association* (4.2.2) when the *information model* (4.2.24) is based upon *object classes* (4.2.35).

4.2.38
relationship end
part of the definition of a *relationship* (4.2.37) as seen from a given *entity type* (4.2.21) which is known as the host

Note 1 to entry: Can also be known as an association end when the *information model* (4.2.24) is based upon *object classes* (4.2.35).

4.2.39
relationship end group
statement that links one or more *relationship ends* (4.2.37) to their host *entity type* (4.2.21) such that they are mutually exclusive

Note 1 to entry: The most common case is where the 'group' comprises of just one relationship end.

4.2.40
relationship end unique identifier element
unique identifier element (4.2.45) that is a statement that a particular *relationship end* (4.2.38) is a part of a particular *unique identifier* (4.2.44)

4.2.41
row
sequence of values in a *table* (4.2.43), one for each *column* (4.2.9) of the table

Note 1 to entry: Within this part, this definition applies only to the registration of database structures that conform to the SQL Core specification as described in ISO/IEC 9075-2:2011.

4.2.42
schema
persistent, named collection of descriptors for *objects* (4.2.34) in a database

Note 1 to entry: Within this part, this definition applies only to the registration of database structures that conform to the SQL Core specification as described in ISO/IEC 9075-2:2011.

Note 2 to entry: Adapted from ISO/IEC 9075-1:2011.

4.2.43
table
basic construct used to represent data in the SQL database language

Note 1 to entry: Within this part, this definition applies only to the registration of database structures that conform to the SQL Core specification as described in ISO/IEC 9075-2:2011.

4.2.44
unique identifier
statement that the values of a specified set of *attributes* (4.2.4) and/or *relationship ends* (4.2.38) are sufficient to uniquely identify an instance of an *entity type* (4.2.21)

Note 1 to entry: Can also be known as a key or a unique key.

4.2.45
unique identifier element
statement that a particular *attribute* (4.2.4) or a particular *relationship end* (4.2.38) is a part of a particular *unique identifier* (4.2.44)

4.2.46
valid value
one of the explicit set of permitted values that comprise an *enumerated domain* (4.2.22)

4.2.47
validation rule
statement of the validation that may be applied to a *described domain* (4.2.12)

Note 1 to entry: This can be a reference to a *data type* (4.2.11) to be applied to *attributes* (4.2.4), a range of values or a 'format mask', or any other expression that constrains the *domain* (4.2.15).

4.3 Abbreviated terms

IRI

Internationalized Resource Identifier

MFI Core and mapping

ISO/IEC 19763-10, *Information technology — Metamodel framework for interoperability (MFI) — Part 10: MFI Core model and basic mapping*

MFI Information model registration

ISO/IEC 19763-12, *Information technology — Metamodel framework for interoperability — Part 12: Metamodel for information model registration*

MDR Metamodel

ISO/IEC 11179-3:2013, *Information technology — Metadata registries (MDR) — Part 3: Registry metamodel and basic attributes*

5 Structure of MFI Information model registration

5.1 Overview of MFI Information model registration

Figure 1 shows the metamodel for the registration of information models developed using the common diagramming techniques and notations listed in Clause 1 above. This metamodel can also be used for registering database structure specifications that conform to the SQL Core specification.

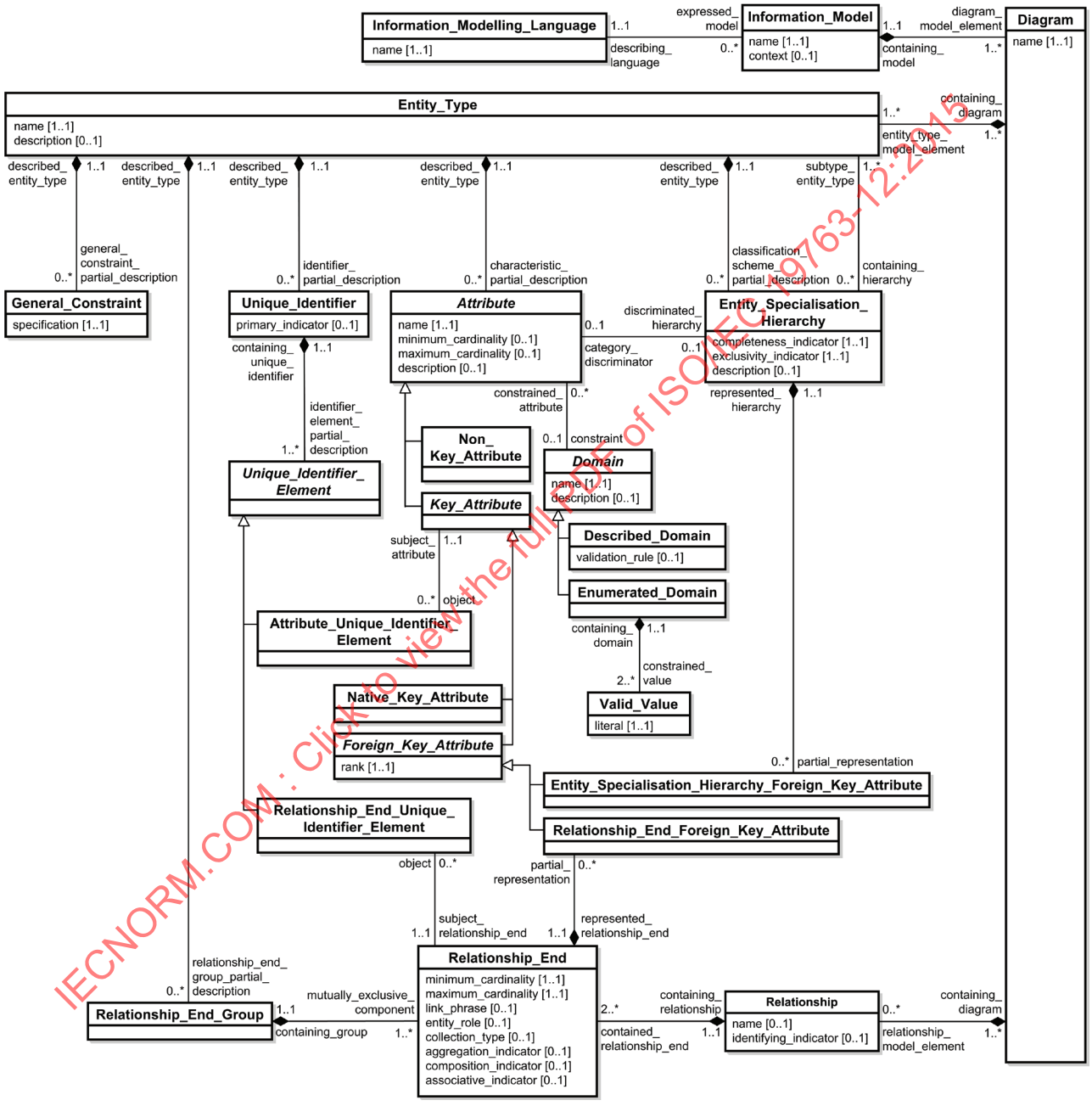

Figure 1 — Metamodel of MFI Information model registration

The metamodel for information model registration comprises the following metaclasses:

— **Attribute**

— **Attribute_Unique_Identifier_Element**, a subclass of **Unique_Identifier_Element**

— **Described_Domain**, a subclass of **Domain**

— **Diagram**

— **Domain**

— **Entity_Specialisation_Hierarchy**

— **Entity_Specialisation_Hierarchy_Foreign_Key_Attribute**, a subclass of **Foreign_Key_Attribute**

— **Entity_Type**

— **Enumerated_Domain**, a subclass of **Domain**

— **Foreign_Key_Attribute**, a subclass of **Key_Attribute**

— **General_Constraint**

— **Information_Model**

— **Information_Modelling_Language**

— **Key_Attribute**, a subclass of **Attribute**

— **Native_Key_Attribute**, a subclass of **Key_Attribute**

— **Non_Key_Attribute**, a subclass of **Attribute**

— **Relationship**

— **Relationship_End**

— **Relationship_End_Foreign_Key_Attribute**, a subclass of **Foreign_Key_Attribute**

— **Relationship_End_Group**

— **Relationship_End_Unique_Identifier_Element**, a subclass of **Unique_Identifier_Element**

— **Unique_Identifier**

— **Unique_Identifier_Element**

— **Valid_Value**

The metamodel is described in detail in Annex A (informative). Detailed specifications of the metaclasses are provided in Clause 5.3 below.

5.2 Association between MFI Information model registration and MFI Core and mapping

The associations between the metaclasses in MFI Information model registration and the metaclasses in MFI Core and mapping are shown in Figure 2.

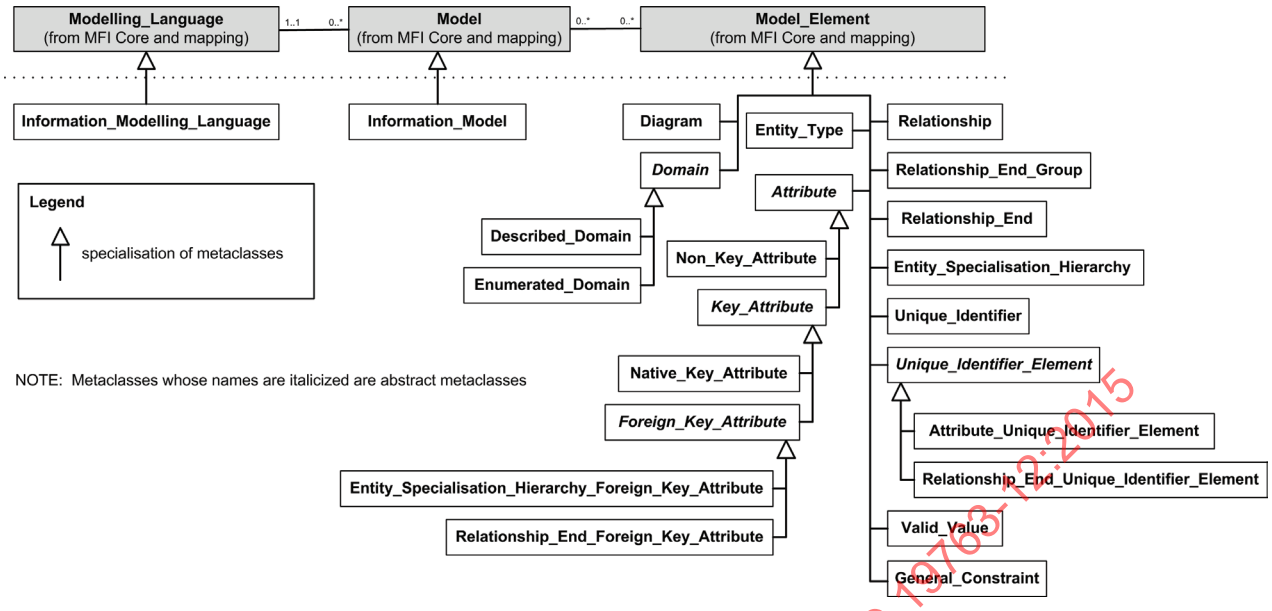

Figure 2 — The associations between MFI Information model registration and MFI Core and mapping

Information_Modelling_Language in MFI Information model registration is a specialisation of (or subclass of) Modelling_Language in MFI Core and mapping.

Information_Model in MFI Information model registration is a specialisation of Model in MFI Core and mapping.

All the remaining metaclasses are specialisations of Model_Element in MFI Core and mapping.

The association between Information_Model and Information_Modelling_ Language in MFI Information model registration is a specialisation of the association between Model and Modelling_Language in MFI Core and mapping.

The association between Information_Model and Diagram in MFI Information model registration is a specialization of the association between Model and Model_Element in MFI Core and mapping.

5.3 Metaclasses in MFI Information Model Registration

5.3.1 Attribute

Attribute is an abstract metaclass each instance of which represents a representation of a particular attribute. The Attribute metaclass may be used to register information about a column in a database structure that conforms to the SQL Core specification as described in ISO/IEC 9075-2:2011.

Superclass

Model_Element (from MFI Core and mapping)

Attribute	DataType	Multiplicity	Description
name	String	1..1	A name for this attribute.
minimum_ cardinality	String	1..1	A statement of the minimum number of occurrences of values of this attribute for any particular instance of the associated entity type. In most circumstances this will be '0' (indicating that the attribute is optional) or '1' (indicating that the attribute is mandatory).
maximum_ cardinality	String	0..1	A statement of the maximum number of occurrences of values of this attribute for any particular instance of the associated entity type. In most information modelling methods this is not specified.
description	String	0..1	A statement that explains the significance of this attribute to the business and or organisation that is the subject of this information model.

Reference	Class	Multiplicity	Description	Inverse	Precedence
described_ entity_type	Entity_Type	1..1	The entity type instances which are qualified, identified, classified, quantified or whose state is otherwise expressed by this attribute.	characteristic_ partial_ description	No
constraint	Domain	0..1	The domain which acts as a constraint on the values taken by this attribute.	constrained_ attribute	No
discriminated_ hierarchy	Entity_ Specialisation_ Hierarchy	0..1	The entity specialisation hierarchy for which this attribute is the category discriminator.	category_ discriminator	No

5.3.2 Attribute_Unique_Identifier_Element

Attribute_Unique_Identifier_Element is a metaclass each instance of which represents a particular attribute unique identifier element.

Superclass

Unique_Identifier_Element

Attribute	DataType	Multiplicity	Description
[None]			

Reference	Class	Multiplicity	Description	Inverse	Precedence
subject_ attribute	Key_Attribute	1..1	The native key attribute that is used as this unique identifier element.	object	No

5.3.3 Described_Domain

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

Superclass

Domain

Attribute	DataType	Multiplicity	Description
validation_rule	String	0..1	A statement of the validation that may be applied to this domain. At its simplest it may just be a statement of the data type that may be applied to attributes. It might show upper and lower bounds of a range of values. It might be a 'format mask'. Or, it may be any combination of these.

Reference	Class	Multiplicity	Description	Inverse	Precedence
[None]					

5.3.4 Diagram

Diagram is a metaclass each instance of which represents a representation of a particular diagram. The Diagram metaclass may be used to register information about a schema in a database structure that conforms to the SQL Core specification as described in ISO/IEC 9075-2:2011.

Superclass

Model_Element (from MFI Core and mapping)

Attribute	DataType	Multiplicity	Description
name	String	1..1	A name by which this diagram is known.

Reference	Class	Multiplicity	Description	Inverse	Precedence
containing_ model	Information_ Model	1..1	The information model of which this diagram is a part	diagram_ model_ element	No
entity_type_ model_element	Entity_Type	1..*	The set of entity types which comprise this information model.	containing_ model	Yes
relationship_ model_element	Relationship	0..*	The set of relationships which comprise this information model.	containing_ model	Yes

5.3.5 Domain

Domain is an abstract metaclass each instance of which represents a representation of a particular domain. The Domain metaclass may be used to register information about a data type in a database structure that conforms to the SQL Core specification as described in ISO/IEC 9075-2:2011.

Superclass

Model_Element (from MFI Core and mapping)

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

Reference	Class	Multiplicity	Description	Inverse	Precedence
constrained_ attribute	Attribute	0..*	The set of attributes whose values are constrained by this domain.	constraint	Yes

5.3.6 Entity_Specialisation_Hierarchy

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

Superclass

Model_Element (from MFI Core and mapping)

Attribute	DataType	Multiplicity	Description
completeness_indicator	Boolean	1..1	An indicator that specifies whether the instances of the associated entity subtypes that form this particular entity specialisation hierarchy are the complete set of the instances of the entity type that is the supertype or not.
exclusivity_indicator	Boolean	1..1	An indicator that specifies whether the instances of the associated entity subtypes that form this particular entity specialisation hierarchy are mutually exclusive or not.
description	String	0..1	A statement that describes the purpose or the classification of this particular entity specialisation hierarchy.

Reference	Class	Multiplicity	Description	Inverse	Precedence
described_entity_type	Entity_Type	1..1	The entity type whose instances are classified or specialised by this entity specialisation hierarchy.	classification_scheme_partial_description	No
subtype_entity_type	Entity_Type	1..*	The set of entity types that are the entity subtypes that comprise this entity specialisation hierarchy.	containing_hierarchy	Yes
category_discriminator	Attribute	0..1	The attribute that is the category discriminator for this entity specialisation hierarchy.	discriminated_hierarchy	Yes
partial_representation	Entity_Specialisation_Hierarchy_Foreign_Key_Attribute	0..*	The set of foreign key attributes that in sequence represent this entity specialisation hierarchy.	represented_hierarchy	Yes

5.3.7 Entity_Specialisation_Hierarchy_Foreign_Key_Attribute

Entity_Specialisation_Hierarchy_Foreign_Key_Attribute is a metaclass each instance of which represents a particular entity specialisation hierarchy foreign key attribute.

Superclass

Foreign_Key_Attribute

Attribute	DataType	Multiplicity	Description
[None]			

Reference	Class	Multiplicity	Description	Inverse	Precedence
represented_ hierarchy	Entity_ Specialisation_ Hierarchy	1..1	The entity specialisation hierarchy which has the entity supertype which is represented by this foreign key attribute along with others in sequence.	partial_ representation	No

5.3.8 Entity_Type

Entity_Type is a metaclass each instance of which represents a representation of a particular entity type. The Entity_Type metaclass may be used to register information about a table in a database structure that conforms to the SQL Core specification as described in ISO/IEC 9075-2:2011.

Superclass

Model_Element (from MFI Core and mapping)

Attribute	DataType	Multiplicity	Description
name	String	1..1	A name by which this entity type is known.
description	String	0..1	A statement that explains the significance of this entity type to the business and or organisation that is the subject of this Information Model.

Reference	Class	Multiplicity	Description	Inverse	Precedence
containing_ diagram	Diagram	1..1	The diagram which includes this entity type.	entity_type_ model_ element	No
classification_ scheme_ partial_ description	Entity_ Specialisation_ Hierarchy	0..*	The set of entity specialisation hierarchies that are used to classify instances of this entity type.	described_ entity_type	Yes
containing_ hierarchy	Entity_ Specialisation_ Hierarchy	0..*	The set of entity specialisation hierarchies which include this entity type as a subtype.	subtype_ entity_type	No

characteristic_partial_description	Attribute	0..*	The set of attributes that are used to qualify, identify, classify, quantify or express the state of any instance of this entity type.	described_entity_type	Yes
relationship_end_group_partial_description	Relationship_End_Group	0..*	The set of relationship end groups each of which has this entity type as their host entity type.	described_entity_type	Yes
identifier_partial_description	Unique_Identifier	0..*	The set of unique identifiers that uniquely identify an instance of this entity type	described_entity_type	Yes
general_constraint_partial_description	General_Constraint	0..*	The set of general constraints that apply to instances of this entity type.	described_entity_type	Yes

5.3.9 Enumerated_Domain

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

Superclass

Domain

Attribute	DataType	Multiplicity	Description
[None]			

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

5.3.10 Foreign_Key_Attribute

Foreign_Key_Attribute is an abstract metaclass each instance of which represents a representation of a particular foreign key attribute.

Superclass

Key_Attribute

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

Reference	Class	Multiplicity	Description	Inverse	Precedence
[None]					

5.3.11 General_Constraint

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

Superclass

Model_Element (from MFI Core and mapping)

Attribute	DataType	Multiplicity	Description
specification	String	1..1	A statement of that formally specifies this constraint.

Reference	Class	Multiplicity	Description	Inverse	Precedence
described_entity_type	Entity_Type	1..1	The entity type that is constrained by this constraint.	general_constraint_partial_description	No

5.3.12 Information_Model

Information_Model is a metaclass each instance of which represents a representation of a particular information model. The Information_Model metaclass may be used to register information about a catalog in a database structure that conforms to the SQL Core specification as described in ISO/IEC 9075-2:2011.

Superclass

Model (from MFI Core and mapping)

Attribute	DataType	Multiplicity	Description
name	String	1..1	A name by which this information model is known.
context	String	0..1	A description of the universe of discourse covered by this information model.

Reference	Class	Multiplicity	Description	Inverse	Precedence
describing_language	Information_Modelling_Language	1..1	The information modelling language in which this model is expressed.	expressed_model	No
diagram_model_element	Diagram	1..*	The set of diagrams which comprise this information model.	containing_model	Yes

5.3.13 Information_Modelling_Language

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

Superclass

Modelling_Language (from MFI Core and mapping)

Attribute	DataType	Multiplicity	Description
name	String	1..1	A name by which this information modelling language is known.

Reference	Class	Multiplicity	Description	Inverse	Precedence
expressed_model	Information_Model	0..*	The set of information models that are expressed in this language.	describing_language	Yes

5.3.14 Key_Attribute

Key_Attribute is an abstract metaclass each instance of which represents a representation of a particular key attribute.

Superclass

Attribute

Attribute	DataType	Multiplicity	Description
[None]			

Reference	Class	Multiplicity	Description	Inverse	Precedence
object	Attribute_ Unique_ Identifier_ Element	0..*	The set of attribute unique identifier elements for which this key attribute acts as such a unique identifier element.	subject_ attribute	Yes

5.3.15 Native_Key_Attribute

Native_Key_Attribute is a metaclass each instance of which represents a representation of a particular native key attribute.

Superclass

Key_Attribute

Attribute	DataType	Multiplicity	Description
[None]			

Reference	Class	Multiplicity	Description	Inverse	Precedence
[None]					

5.3.16 Non_Key_Attribute

Non_Key_Attribute is a metaclass each instance of which represents a representation of a particular non-key attribute.

Superclass

Attribute

Attribute	DataType	Multiplicity	Description
[None]			

Reference	Class	Multiplicity	Description	Inverse	Precedence
[None]					

5.3.17 Relationship

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

Superclass

Model_Element (from MFI Core and mapping)

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

identifying_ indicator	Boolean	0..1	If this relationship is a binary 'one-to-many' relationship, an indicator that specifies whether this relationship provides part (or all) of the primary unique identifier for the entity type that is at the 'many' end of the relationship or not. Not all information modelling methods recognise this concept.

Reference	Class	Multiplicity	Description	Inverse	Precedence
containing_ diagram	Diagram	1..1	The diagram which includes this relationship.	relationship_model_ element	No
contained_ relationship_ end	Relationship_ End	2..*	The set of relationship ends that comprise this relationship.	containing_ relationship	Yes

5.3.18 Relationship_End

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

Superclass

Model_Element (from MFI Core and mapping)

Attribute	DataType	Multiplicity	Description
minimum_ cardinality	String	1..1	A statement of the minimum number of instances of the associated entity type (through the associated relationship end group) that must participate in the relationship of which this relationship end is a part. In most circumstances this will be '0' (indicating that the entity type has optional participation) or '1' (indicating that the entity type has mandatory participation).
maximum_ cardinality	String	1..1	A statement of the maximum number of instances of the associated entity type (through the associated relationship end group) that may participate in the relationship of which this relationship end is a part. In most circumstances this will be '1' (indicating that one and only one entity type may participate) or '*' (indicating that an unspecified number of entity types may participate).
link_phrase	String	0..1	A statement that explains the nature of the relationship of which this relationship end is a part from the perspective of the associated entity type (through the associated relationship end group). This is normally expressed in business terms. Not all information modelling methods recognise this concept.
entity_role	String	0..1	A statement that explains the role that the associated entity type (through the associated relationship end group) is playing in the associated relationship. Not all information modelling methods recognise this concept.
collection_ type	String	0..1	A statement as to whether the instances of the associated entity type (through the associated relationship end group) are considered to be a 'set', a 'bag' (or 'multiset'), a 'list' or an 'array'. Most information modelling methods do not recognise this concept.

aggregation_ indicator	Boolean	0..1	An indicator that specifies whether the instance of the associated entity type (through the associated relationship end group) is considered to be an aggregation of the instances of the other entity type participating in the relationship (identified through the associated relationship and relationship end group) or not. Most information modelling methods do not recognise this concept.
composition_ indicator	Boolean	0..1	An indicator that specifies whether the instance of the associated entity type (through the associated relationship end group) is considered to be a composition of the instances of the other entity type participating in the relationship (identified through the associated relationship and relationship end group) or not. Most information modelling methods do not recognise this concept.
associative_ indicator	Boolean	0..1	An indicator that specifies whether the instance of the associated entity type (through the associated relationship end group) is an associative entity type or not. Most information modelling methods do not recognise this concept.

Reference	Class	Multiplicity	Description	Inverse	Precedence
containing_ relationship	Relationship	1..1	The relationship of which this relationship end is a part.	contained_ relationship_ end	No
containing_ group	Relationship_ End_ Group	1..1	The relationship end group of which this relationship is one of the relationship ends each of which is mutually exclusive with others in the group.	mutually_ exclusive_ component	No
partial_ representation	Relationship_ End_Foreign_ Key_Attribute	0..*	The set of foreign key attributes that in sequence represent this relationship end.	represented_ relationship_ end	Yes
object	Relationship_ End_Unique_ Identifier_ Element	0..*	The set of relationship end unique identifier elements for which this relationship end acts as such a unique identifier element.	subject_ relationship_ end	Yes

5.3.19 Relationship_End_Foreign_Key_Attribute

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

Superclass

Foreign_Key_Attribute

Attribute	DataType	Multiplicity	Description

[None]

Reference	Class	Multiplicity	Description	Inverse	Precedence
represented_ relationship_ end	Relationship_End	1..1	The relationship end which is, along with others in sequence, represented by this foreign key attribute.	partial_ representation	Yes

5.3.20 Relationship_End_Group

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

Superclass

Model_Element (from MFI Core and mapping)

Attribute	DataType	Multiplicity	Description
[None]			

Reference	Class	Multiplicity	Description	Inverse	Precedence
described_ entity_type	Entity_Type	1..1	The entity type (the host entity type) that is related to other entity types through this relationship end group.	relationship_ end_group_ partial_description	No
mutually_ exclusive_ component	Relationship_ End	1..*	The set of relationship ends that form part of this group and are mutually exclusive with each other.	containing_group	Yes

5.3.21 Relationship_End_Unique_Identifier_Element

Relationship_End_Unique_Identifier_Element is a metaclass each instance of which represents a particular relationship end unique identifier element.

Superclass

Unique_Identifier_Element

Attribute	DataType	Multiplicity	Description
[None]			

Reference	Class	Multiplicity	Description	Inverse	Precedence
subject_ relationship_ end	Relationship_End	1..1	The relationship end that is used as this unique identifier element.	object	Yes

5.3.22 Unique_Identifier

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

Superclass

Model_Element (from MFI Core and mapping)

Attribute	DataType	Multiplicity	Description
primary_ indicator	Boolean	0..1	An indicator that specifies whether this unique identifier is the primary unique identifier of the associated entity or not. Some information modelling methods do not recognise this concept.

Reference	Class	Multiplicity	Description	Inverse	Precedence
described_ entity_type	Entity_Type	1..1	The entity type whose instances can be uniquely identified by this unique identifier.	identifier_ partial_ description	No
identifier_ element_ partial_ description	Unique_ Identifier_ Element	1..*	The set of unique identifier elements that comprise this unique identifier.	containing_ unique_ identifier	Yes

5.3.23 Unique_Identifier_Element

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

Superclass

Model_Element (from MFI Core and mapping)

Attribute	DataType	Multiplicity	Description
[None]			

Reference	Class	Multiplicity	Description	Inverse	Precedence
containing_ unique_ identifier	Unique_Identifier	1..1	The unique identifier of which this unique identifier element is a part.	identifier_ element_ partial_ description	No

5.3.24 Valid_Value

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

Superclass

Model_Element (from MFI Core and mapping)

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

Reference	Class	Multiplicity	Description	Inverse	Precedence
containing_ domain	Enumerated_ Domain	1..1	The enumerated domain of which this valid value is one of the permitted values.	constrained_ value	No

Annex A
(informative)

Description of the metamodel

Figure A.1 repeats the metamodel for the registration of information models, which is then described in detail.

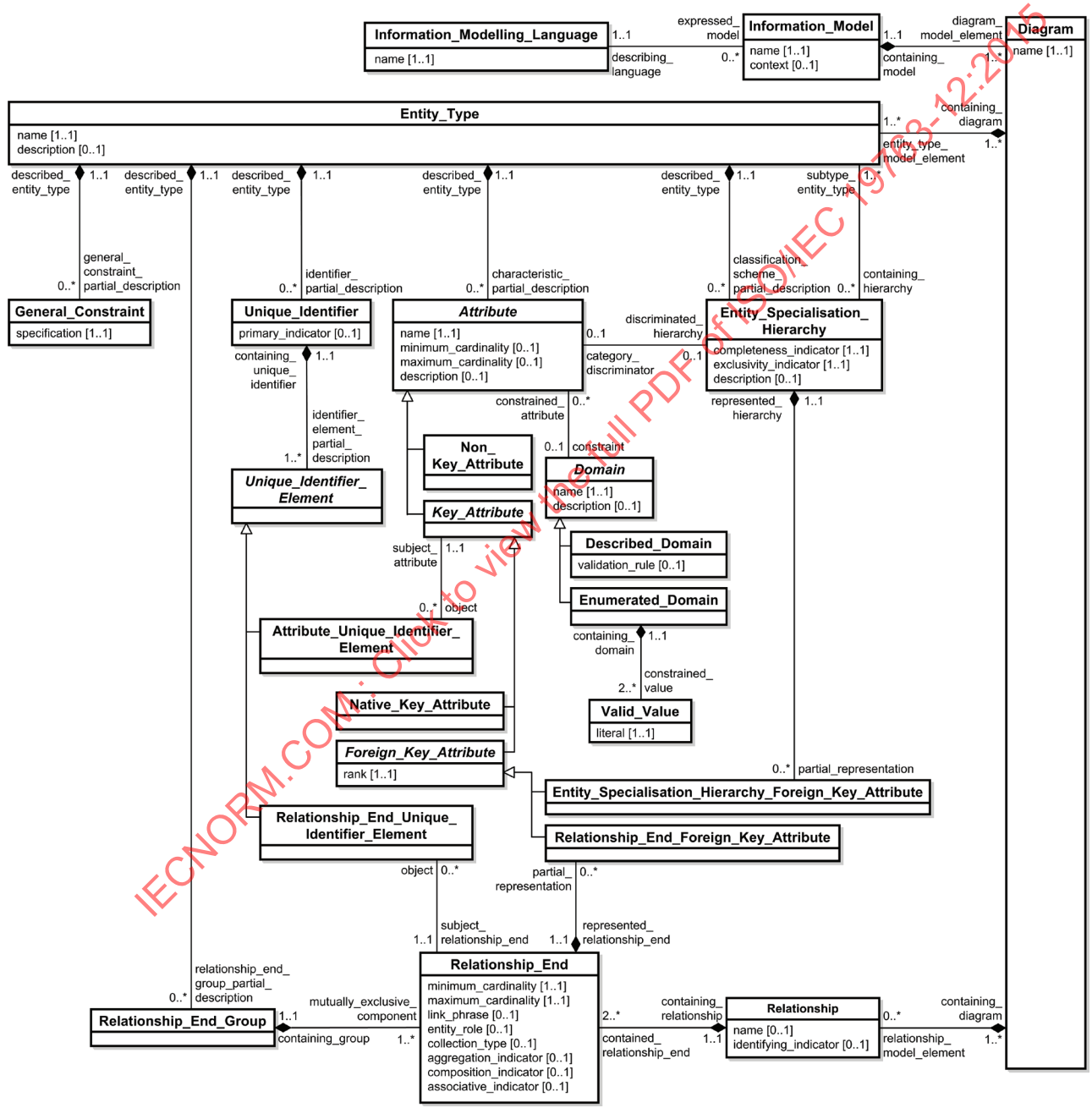

Figure A.1 — Metamodel of MFI Information model registration (repeated)

Each information modelling language is a language or notation that is used to model information requirements in an information model. In addition, each information modelling language:

— may be the language used to express zero, one or more information models (each of which are expressed models); not every information modelling language has to be used to express an information model.

— must have a name, which is a unique name by which this information modelling language is known.

Each information model is a graphical and textual representation of entities and the relationships between them. In addition, each information model:

— must be expressed in one and only one information modelling language (the describing language).

— must be comprised of one or more diagrams (each of which are diagram model elements).

— must have a name, which is a unique name by which this information model is known.

— may have a context, which is a description of the universe of discourse covered by the model; not every information model has to have a context.

Each diagram is a technical document showing part of an information model using graphical symbols. In addition, each diagram:

— must be part of one and only one information model (the containing model).

— must be comprised of one or more entity types (each of which are entity type model elements).

— may be comprised of zero, one or more relationships (each of which are relationship model elements); not every diagram has to be comprised of relationships.

— must have a name, which is a unique name by which this diagram is known.

Each entity type is a set of characteristics common to a collection of entities that are instances of the type. In addition, each entity type:

— must be part of one and only one diagram (the containing diagram).

— may be further defined with zero, one or more entity specialisation hierarchies (each of which is a classification scheme partial description); not every entity type has to be further defined with an entity specialisation hierarchy.

— may be used as a subtype in zero, one or more entity specialisation hierarchies; not every entity type has to be used as a subtype in an entity specialisation hierarchy.

— may be with instances described by values assigned to each of zero, one or more attributes (each of which is a characteristic partial description); not every entity type has to be defined with attributes.

— may be related to others through zero, one or more relationship end groups (each of which is a relationship end group partial description); not every entity type has to be related to others through a relationship end group.

— may be with instances identified by zero, one or more unique identifiers (each of which is an identifier partial description); not every entity type has to be defined with a unique identifier.

— may be further constrained by zero, one or more general constraints (each of which is a general constraint partial description); not every entity type has to be further constrained by a general constraint.

— must have a name, which is a unique name by which this entity type is known.

— may have a description, which is a statement that explains the significance of this entity type to the business and or organisation that is the subject of this information model; not every entity type has to have a description.

Each entity specialisation hierarchy is a means by which instances of an entity type may be classified or specialised. In addition, each entity specialisation hierarchy:

— must be a further description of one and only one entity type (the described entity type).

— must be comprised of one or more entity types (each of which is a subtype within the classification scheme).

— may be categorised by zero or one attribute (the category discriminator); most information modelling methods do not recognise the concept of a categorising attribute so not every entity specialisation hierarchy has to be categorised by an attribute.

— may be partially represented by zero, one or more entity specialisation hierarchy foreign key attributes; most information modelling methods do not recognise the concept of a representing entity specialisation hierarchies using foreign keys so not every entity specialisation hierarchy has to be represented by an entity specialisation hierarchy foreign key attribute.

— must have a completeness indicator, which is an indicator that specifies whether the instances of the associated entity subtypes that form this particular entity specialisation hierarchy are the complete set of the instances of the entity type that is the supertype or not.

— must have an exclusivity indicator, which is an indicator that specifies whether the instances of the associated entity subtypes that form this particular entity specialisation hierarchy are mutually exclusive or not.

— may have a description, which is a statement that describes the purpose or the classification of this particular entity specialisation hierarchy; some information modelling methods do not recognise the concept of describing an entity specialisation hierarchy so not every entity specialisation hierarchy has to have a description.

Each relationship is a set of characteristics common to a collection of connections between instances of two or more entity types, or between instances of one entity type and other instances of the same entity type. In addition, each relationship:

— must be part of one and only one diagram (the containing diagram).

— must be comprised of two or more relationship ends (each of which is a contained relationship end).

— may have a name, which is a name, which may not be unique within the information model, by which this relationship is known; some information modelling methods do not provide such a name so not every relationship has to have a name.

— may have an identifying indicator, which, if this relationship is a binary 'one-to-many' relationship, is an indicator that specifies whether this relationship provides part (or all) of the primary unique identifier for the entity type that is at the 'many' end of the relationship or not; not all information modelling methods recognise this concept so not every relationship has to have an identifying indicator.

Each relationship end is the part of the definition of a relationship as seen from a given entity type (the host). In addition, each relationship end:

— must be part of one and only one relationship (the containing relationship).

— must be mutually exclusive with other relationship ends within one and only one relationship end group (the containing group).

— may be represented by a sequence of zero, one or more relationship end foreign key attributes (each of which is a partial representation); some information modelling methods do not represent relationship ends by foreign key attributes so not every relationship end has be represented by a sequence of relationship end foreign key attributes.

— may be used as zero, one or more relationship end unique identifier elements (each of which is an object); some information modelling methods do not use relationship ends as unique identifier elements so not every relationship end has be used as a relationship end unique identifier element.

— must have a minimum cardinality, which is a statement of the minimum number of instances of the associated entity type (through the associated relationship end group) that must participate in the relationship of which this relationship end is a part; in most circumstances this will be '0' (indicating that the entity type has optional participation) or '1' (indicating that the entity type has mandatory participation).

— must have a maximum cardinality, which is a statement of the maximum number of instances of the associated entity type (through the associated relationship end group) that may participate in the relationship of which this relationship end is a part; in most circumstances this will be '1' (indicating that one and only one entity type may participate) or '*' (indicating that an unspecified number of entity types may participate).

— may have a link phrase, which is a statement, normally expressed in business terms, that explains the nature of the relationship of which this relationship end is a part from the perspective of the associated entity type (through the associated relationship end group); some information modelling methods do not use link phrases so not every relationship end has to have a link phrase.

— may have an entity role, which is a statement that explains the role that the associated entity type (through the associated relationship end group) is playing in the associated relationship; some information modelling methods do not use entity role so not every relationship end has to have an entity role.

— may have a collection type, which is a statement as to whether the instances of the associated entity type (through the associated relationship end group) are considered to be a 'set', a 'bag' (or 'multiset'), a 'list' or an 'array'; most information modelling methods do not recognise this concept so not every relationship end has to have a collection type.

— may have an aggregation indicator, which is an indicator that specifies whether the instance of the associated entity type (through the associated relationship end group) is considered to be an aggregation of the instances of the other entity type participating in the relationship (identified through the associated relationship and relationship end group) or not; most information modelling methods do not recognise this concept so not every relationship end has to have an aggregation indicator.

— may have a composition indicator, which is an indicator that specifies whether the instance of the associated entity type (through the associated relationship end group) is considered to be a composition of the instances of the other entity type participating in the relationship (identified through the associated relationship and relationship end group) or not; most information modelling methods do not recognise this concept so not every relationship end has to have a composition indicator.

— may have an associative indicator, which is an indicator that specifies whether the instance of the associated entity type (through the associated relationship end group) is an associative entity type or not; most information modelling methods do not recognise this concept so not every relationship end has to have an associative indicator.

Each relationship end group is a statement that links one or more relationship ends to their host entity type such that they are mutually exclusive. The most common case is where the 'group' comprises just one relationship end. In addition, each relationship end group:

— must be viewed from one and only one entity type (the described entity type).

— must be defined to include one or more relationship ends (each of which is a mutually exclusive component).

Each attribute is a named characteristic of an entity type whose values serve to qualify, identify, classify, quantify or express the state of an instance of an entity type. In addition, each attribute:

— must be either a key attribute or a non-key attribute, but not both.

— must be defined to be part of the description of one and only one entity type (the described entity type).

— may be assigned with values from zero or one domain (the constraint); some information modelling methods do not recognise the concept of assigning a domain to an attribute so not every attribute has to be assigned with values from a domain.

— may be the category discriminator for zero or one entity specialisation hierarchy (the discriminated hierarchy); most information modelling methods do not recognise the concept of a discriminating or categorising attribute so not every attribute has to be a category discriminator for an entity specialisation hierarchy.

— must have a name, which is a unique name for this attribute; in some information modelling methods this will be unique within the entity type (in which case the entity type name must be concatenated with the attribute name to gain model uniqueness) whilst in other information modelling methods this will be unique within the information model.

— may have a minimum cardinality, which is a statement of the minimum number of occurrences of values of this attribute for any particular instance of the associated entity type; in most circumstances this will be '0' (indicating that the attribute is optional) or '1' (indicating that the attribute is mandatory); some information modelling methods do not recognise the concept of cardinality of attributes so not every attribute has to have a minimum cardinality.

— may have a maximum cardinality, which is a statement of the maximum number of occurrences of values of this attribute for any particular instance of the associated entity type; in most information modelling methods this is not specified because it is assumed that this the maximum cardinality is '1'; some information modelling methods do not recognise the concept of cardinality of attributes so not every attribute has to have a maximum cardinality.

— may have a description, which is a statement that explains the significance of this attribute to the business and or organisation that is the subject of this information model; some information modelling methods do not recognise the concept of descriptions for attributes so not every attribute has to have a description.

Each non-key attribute is an attribute that is not the unique identifier or an element of a composite unique identifier of an entity type or whose value is fully independent of all relationships or other attributes.

Each key attribute is an attribute whose value contributes in some way to the identification of individual instances of the host entity type or of some related entity type. In addition each key attribute:

— must be either a native key attribute or a foreign key attribute, but not both.

— may be used as zero, one or more attribute unique identifier elements; some attributes may not be used as elements of a unique identifier so not every key attribute has to be used as zero, one or more attribute unique identifier elements.

Each native key attribute is a key attribute whose value contributes in some way to the identification of individual instances of the host entity type.

Each foreign key attribute is a key attribute whose value contributes in some way to the identification of the one related instance of that entity type involved in the associated relationship end. In addition, each foreign key attribute:

— must be either a relationship end foreign key attribute or an entity specialisation hierarchy foreign key attribute, but not both.

— must have a rank, which is a statement of the position of this foreign key attribute in the sequence of the foreign key attributes that make up the foreign key of which this foreign key attribute is a part.

Each relationship end foreign key attribute is a foreign key attribute whose referenced attribute is in an entity type that is related to the entity type for which this foreign key attribute is defined through a relationship. In addition, each relationship end foreign key attribute must be part of a sequence forming a representation of one and only one relationship end (the represented relationship end).

Each entity specialisation hierarchy foreign key attribute is a foreign key attribute whose referenced attribute in the entity supertype of the related entity specialisation hierarchy. In addition, each entity specialisation hierarchy foreign key attribute must be part of a sequence forming a representation of one and only one entity specialisation hierarchy (the represented hierarchy).

Each domain is a collection of values from which an instance of an attribute must take its value. A domain provides a set of business validation rules, format constraints and other properties for one or more attributes. In addition, each domain:

— must be either a described domain or an enumerated domain, but not both.

— may be a constraint on zero, one or more attributes (the constrained attribute).

— must have a name, which is a unique name by which this domain is known.

— may have a description, which is a statement that explains the significance of this domain to the business and or organisation that is the subject of this information model; some information modelling methods do not recognise the concept of descriptions for domains so not every domain has to have a significance statement.

Each described domain is a domain whose values are not drawn from an explicit list of valid values. In addition, each described domain may have a validation rule, which is a statement of the validation that may be applied to this domain. At its simplest this validation rule may just be a statement of the data type that may be applied to attributes. It might show upper and lower bounds of a range of values. It might be a 'format mask'. Or, it may be any combination of these. Some information modelling methods do not recognise the concept of validation rules for domains so not every domain has to have a validation rule.

Each enumerated domain is a domain whose permitted values consist of an explicit list of valid values. In addition, each enumerated domain must be constrained to two or more valid values (each of which is a constrained value).

Each valid value is one of the explicit set of permitted values that comprise an enumerated domain. In addition, each valid value:

— must be for one and only one enumerated domain (the constraining domain).

— must have a literal, which is the actual permitted value; in a platform independent model it will probably be the concept, for example, 'Male' whilst in a platform specific model it will probably be the code, for example, 'M' or '0'.

Each unique identifier is a statement that the values of a specified set of attributes and/or relationship ends are sufficient to uniquely identify an instance of an entity type. In addition, each unique identifier:

— must be defined for one and only one entity type (the described entity type).

— must be comprised of one or more unique identifier elements (each of which is an identifier element partial description).

— may have a primary indicator, which is an indicator that specifies whether this unique identifier instance is the primary unique identifier of the associated entity or not; some information modelling methods do not recognise this concept so not every unique identifier has to have a primary indicator.

Each unique identifier element is a statement that a particular attribute or a particular relationship end is a part of a particular unique identifier. In addition, each unique identifier element:

— must be either an attribute unique identifier element or a relationship end unique identifier element, but not both.

— must be part of one and only one unique identifier (the containing unique identifier).

Each attribute unique identifier element is a unique identifier element that is a statement that a particular attribute is a part of a particular unique identifier. In addition, each attribute unique identifier element must be a role played by one and only one key attribute (the subject attribute).

Each relationship end unique identifier element is a unique identifier element that is a statement that a particular relationship end is a part of a particular unique identifier. In addition, each relationship end unique identifier element must be a role played by one and only one relationship end (the subject relationship end).

Each general constraint is a constraint on the instances of an entity type that cannot be expressed using any other constructs in the metamodel. In addition, each general constraint:

— must be defined to be part of the description of one and only one entity type (the described entity type).

— must have a specification, which is a statement that formally specifies this constraint.

Annex B
(informative)

Relationship of metaclasses to the MDR Metamodel

As explained in ISO/IEC 19763-10, instances of the metaclasses defined in this part of ISO/IEC 19763 can be extended by types such as defined in the MDR Metamodel as follows:

— Instances of **Information_Modelling_Language** may be extended as an **Identified_Item** and as a **Designatable_Item**.

— Instances of **Information_Model** may be extended as an **Administered_Item** and as a **Designatable_Item**.

— Instances of **Entity_Type** may be extended as an **Attached_Item** and as a **Designatable_Item**.

— Instances of **Entity_Specialisation_Hierarchy** may be extended as an **Attached_Item** and as a **Designatable_Item**.

— Instances of **Relationship** may be extended as an **Attached_Item** and as a **Designatable_Item**.

— Instances of **Relationship_End** may be extended as an **Attached_Item** and as a **Designatable_Item**.

— Instances of **Relationship_End_Group** may be extended as an **Attached_Item**.

— Instances of **Attribute** (and its subclasses) may be extended as an **Attached_Item** and as a **Designatable_Item**.

— Instances of **Domain** (and its subclasses) may be extended as an **Attached_Item** and as a **Designatable_Item**.

— Instances of **Valid_Value** may be extended as an **Attached_Item** and as a **Designatable_Item**.

— Instances of **Unique_Identifier** may be extended as an **Attached_Item**.

— Instances of **Unique_Identifier_Element** (and its subclasses) may be extended as an **Attached_Item**.

— Instances of **General Constraint** may be extended as an **Attached_Item**.

Annex C
(informative)

Applicability of information modelling concepts to techniques

Table C.1 shows the applicability of major information modelling concepts to the techniques and notations used in the development of this part of ISO/IEC 19763.

Table C.1 — Use of concepts within techniques

Concept	Applicability						
	Ellis-Barker	IDEF1X	Information Engineering	Chen	Express-G	UML Class Diagrams	SQL DDL
Entity types are named	Yes	Yes	Yes	Yes	Yes	Yes	Yes (as Tables)
Entity types have descriptions	Optional	Optional	Optional	Optional	Optional	Optional	No
Entity specialization hierarchy allowed	Yes	Yes	Yes	No	Yes	Yes	No
Only one entity specialization hierarchy allowed	Yes	Yes	Yes	NA	No	No	NA
Entity specialization hierarchies must be complete	Yes	No	Yes	NA	No	No	NA
Entity subtypes in an entity specialization hierarchy must be mutually exclusive	Yes	Yes	Yes	NA	No	No	NA
Relationships have one name	No	Yes	Yes	Yes	Yes	Optional	NA
Relationships have two names (link phrases), one for each relationship end	Yes	No	No	No	No	No	NA
Relationships are recognised as identifying or non-identifying relationships	Optional	Yes	No	No	No	No	NA
Entity types are given role names to signify their role in a relationship	No	No	No	No	No	Optional	NA
Entity types at a relationship end can be recognized as being in a collection	No	No	No	No	Yes	No	NA
Relationships can be annotated to recognise that the entity types at the other end of the relationship form an 'aggregation'	No	No	No	No	No	Yes	NA
Relationships can be annotated to recognise that the entity types at the other end of the relationship form an 'composition'	No	No	No	No	No	Yes	NA
All relationships are binary relationships	Yes	Yes	Yes	No	Yes	Yes	NA
n-ary relationships allowed	No	No	No	Yes	No	No	NA
Attributes are named	Yes	Yes	Yes	Yes	Yes	Yes	Yes (as Columns)
Attributes have descriptions	Optional	Optional	Optional	Optional	Optional	Optional	No

Table C.1 *(continued)*

Concept	Applicability						
	Ellis-Barker	IDEF1X	Information Engineering	Chen	Express-G	UML Class Diagrams	SQL DDL
Domains are named	Yes	Yes (as Data Types)	No	No	Yes (as Data Types)	Yes (as Data Types)	Yes (as Data Types)
Domains have descriptions	Optional	Optional	No	No	No	No	No
Foreign keys are documented	No	Yes	Optional	No	No	No	Yes
Unique identifiers using native key attributes and foreign key attributes are documented	No	Yes	Optional	No	No	No	Yes (as Primary Keys)
Unique identifiers using native key attributes and relationships are documented	Optional	No	No	No	No	No	No

Annex D
(informative)

Examples of information model registration

D.1 General

This annex illustrates the registration of information models using the metamodel specified in MFI Information model registration. The examples are not exhaustive.

The examples all follow the same format. First the example models are presented and this is then followed by a set of illustrative instances. Each instance is identified with the name of the metaclass in angle brackets.

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

D.2 EXPRESS-G example

This example is based on a car sales scenario which is drawn in EXPRESS-G notation (see Figure D.1). Figure D.2 provides the object instances to illustrate the registration of this model.

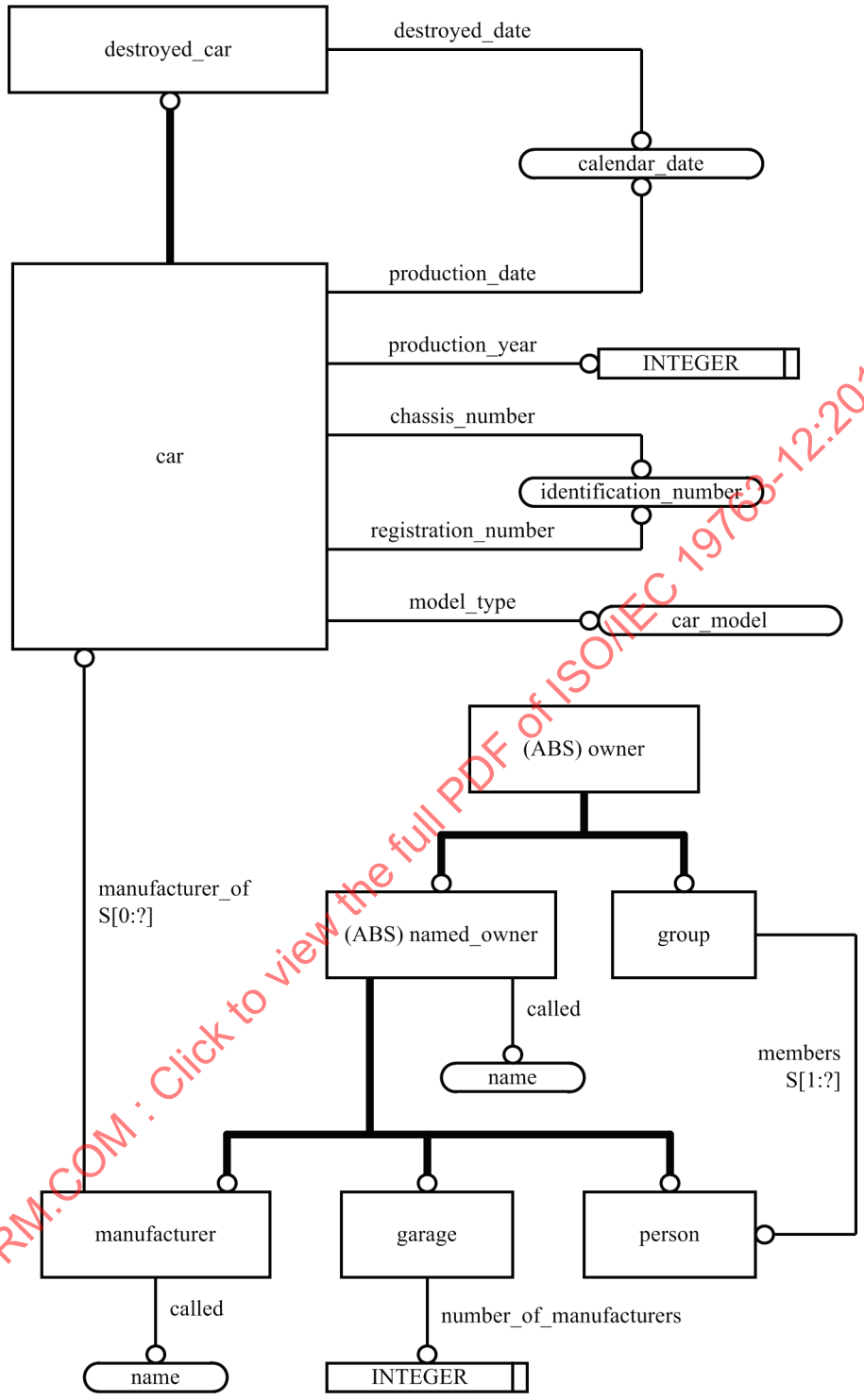

Figure D.1 — Example information model drawn in EXPRESS-G

<Information_Modelling_Language>
Object101

Attribute/Reference	Literal/Instance
name	"EXPRESS-G"
expressed_model	Object102

<Information_Model>
Object102

Attribute/Reference	Literal/Instance
name	"Car Sales Version 2.7"
describing_language	Object101
diagram_model_element	Object103

<Diagram>
Object103

Attribute/Reference	Literal/Instance
name	"Car Sales Version 2.7 Diagram 1"
containing_model	Object102
entity_type_model_element	Object104, Object105, Object106, Object107, Object108, Object109, Object110, Object111
relationship_model_element	Object129, Object130

<Entity_Type>
Object104

Attribute/Reference	Literal/Instance
name	"car"
containing_diagram	Object103
classification_scheme_partial_description	Object112
characteristic_partial_description	Object115, Object116, Object117, Object118, Object119
relationship_end_group_partial_description	Object136

<Entity_Type>
Object105

Attribute/Reference	Literal/Instance
name	"destroyed_car"
containing_diagram	Object103
containing_hierarchy	Object112
characteristic_partial_description	Object120

<Entity_Type>
Object106

Attribute/Reference	Literal/Instance
name	"owner"
containing_diagram	Object103
classification_scheme_partial_description	Object113

<Entity_Type>
Object107

Attribute/Reference	Literal/Instance
name	"named_owner"
containing_diagram	Object103
classification_scheme_partial_description	Object114
containing_hierarchy	Object113
characteristic_partial_description	Object121

<Entity_Type>
Object108

Attribute/Reference	Literal/Instance
name	"group"
containing_diagram	Object103
containing_hierarchy	Object113
relationship_end_group_partial_description	Object137

<Entity_Type>
Object109

Attribute/Reference	Literal/Instance
name	"manufacturer"
containing_diagram	Object103
containing_hierarchy	Object114
characteristic_partial_description	Object122
relationship_end_group_partial_description	Object135

<Entity_Type>
Object110

Attribute/Reference	Literal/Instance
name	"garage"
containing_diagram	Object103
containing_hierarchy	Object114
characteristic_partial_description	Object123

<Entity_Type>
Object111

Attribute/Reference	Literal/Instance
name	"person"
containing_diagram	Object103
containing_hierarchy	Object114
relationship_end_group_partial_description	Object138

<Entity_Specialisation_Hierarchy>
Object112

Attribute/Reference	Literal/Instance
completeness_indicator	False
exclusivity_indicator	True
described_entity_type	Object104
subtype_entity_type	Object105

<Entity_Specialisation_Hierarchy>
Object113

Attribute/Reference	Literal/Instance
completeness_indicator	True
exclusivity_indicator	True
described_entity_type	Object106
subtype_entity_type	Object107, Object108

<Entity_Specialisation_Hierarchy>
Object114

Attribute/Reference	Literal/Instance
completeness_indicator	True
exclusivity_indicator	True
described_entity_type	Object107
subtype_entity_type	Object109, Object110, Object111

<Non_Key_Attribute>
Object115

Attribute/Reference	Literal/Instance
name	"production_date"
described_entity_type	Object104
constraint	Object125

<Non_Key_Attribute>
Object116

Attribute/Reference	Literal/Instance
name	"production_year"
described_entity_type	Object104
constraint	Object124

<Non_Key_Attribute>
Object117

Attribute/Reference	Literal/Instance
name	"chassis_number"
described_entity_type	Object104
constraint	Object126

Figure D.2 — Registration of the EXPRESS-G example (Part 1 of 2)

<Non_Key_Attribute>
Object118

Attribute/Reference	Literal/Instance
name	"registration_number"
described_entity_type	Object104
constraint	Object126

<Non_Key_Attribute>
Object119

Attribute/Reference	Literal/Instance
name	"model_type"
described_entity_type	Object104
constraint	Object127

<Non_Key_Attribute>
Object120

Attribute/Reference	Literal/Instance
name	"destroyed_date"
described_entity_type	Object105
constraint	Object125

<Non_Key_Attribute>
Object121

Attribute/Reference	Literal/Instance
name	"called"
described_entity_type	Object107
constraint	Object128

<Non_Key_Attribute>
Object122

Attribute/Reference	Literal/Instance
name	"called"
described_entity_type	Object109
constraint	Object128

<Non_Key_Attribute>
Object123

Attribute/Reference	Literal/Instance
name	"number_of_manufacturers"
described_entity_type	Object110
constraint	Object124

<Described_Domain>
Object124

Attribute/Reference	Literal/Instance
name	"INTEGER"
constrained_attribute	Object116, Object123

<Described_Domain>
Object125

Attribute/Reference	Literal/Instance
name	"calendar_date"
constrained_attribute	Object115, Object120

<Described_Domain>
Object126

Attribute/Reference	Literal/Instance
name	"identification_number"
constrained_attribute	Object117, Object118

<Described_Domain>
Object127

Attribute/Reference	Literal/Instance
name	"car_model"
constrained_attribute	Object119

<Described_Domain>
Object128

Attribute/Reference	Literal/Instance
name	"name"
constrained_attribute	Object121, Object122

<Relationship>
Object129

Attribute/Reference	Literal/Instance
name	"manufacturer_of"
containing_diagram	Object103
contained_relationship_end	Object131, Object132

<Relationship>
Object130

Attribute/Reference	Literal/Instance
name	"members"
containing_diagram	Object103
contained_relationship_end	Object133, Object134

<Relationship_End>
Object131

Attribute/Reference	Literal/Instance
minimum_cardinality	"One"
maximum_cardinality	"One"
containing_relationship	Object129
containing_group	Object135

<Relationship_End>
Object132

Attribute/Reference	Literal/Instance
minimum_cardinality	"One"
maximum_cardinality	"Many"
collection_type	"Set"
containing_relationship	Object129
containing_group	Object136

<Relationship_End>
Object133

Attribute/Reference	Literal/Instance
minimum_cardinality	"One"
maximum_cardinality	"One"
containing_relationship	Object130
containing_group	Object137

<Relationship_End>
Object134

Attribute/Reference	Literal/Instance
minimum_cardinality	"One"
maximum_cardinality	"Many"
collection_type	"Set"
containing_relationship	Object130
containing_group	Object138

<Relationship_End_Group>
Object135

Attribute/Reference	Literal/Instance
described_entity_type	Object109
mutually_exclusive_component	Object131

<Relationship_End_Group>
Object136

Attribute/Reference	Literal/Instance
described_entity_type	Object104
mutually_exclusive_component	Object132

<Relationship_End_Group>
Object137

Attribute/Reference	Literal/Instance
described_entity_type	Object108
mutually_exclusive_component	Object133

<Relationship_End_Group>
Object138

Attribute/Reference	Literal/Instance
described_entity_type	Object111
mutually_exclusive_component	Object134

Figure D.2 — Registration of the EXPRESS-G example (Part 2 of 2)

D.3 IDEF1X example

This example is based on an employee relationships scenario which is drawn in IDEF1X notation (see Figure D.3). Figure D.4 provides the object instances to illustrate the registration of this model.

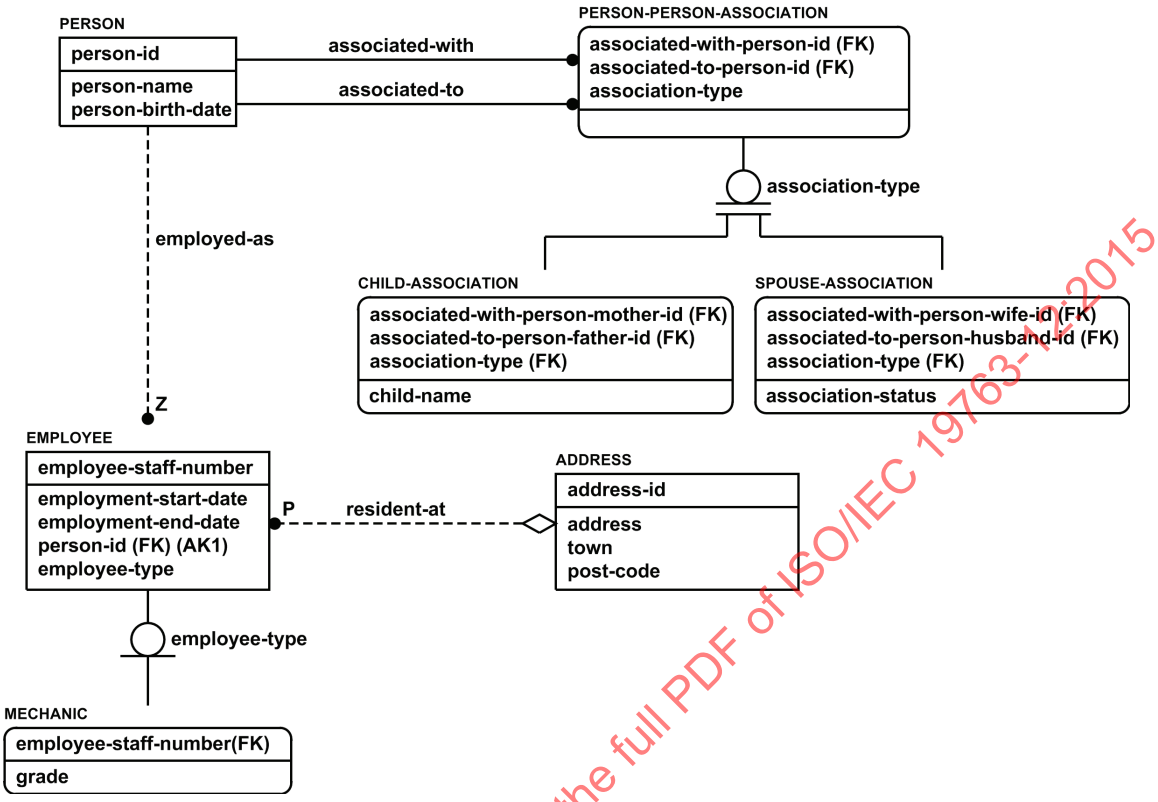

Figure D.3 — Example information model drawn in IDEF1X

<Information_Modelling_Language>
Object201

Attribute/Reference	Literal/Instance
name	"IDEF1X"
expressed_model	Object202

<Information_Model>
Object202

Attribute/Reference	Literal/Instance
name	"Employee Relationships V0.4"
describing_language	Object201
diagram_model_element	Object203

<Diagram>
Object203

Attribute/Reference	Literal/Instance
name	"Employee Relationships V0.4 Diagram 1"
containing_model	Object202
entity_type_model_element	Object204, Object205, Object206, Object207, Object208, Object209, Object210
relationship_model_element	Object213, Object214, Object215, Object216

<Entity_Type>
Object204

Attribute/Reference	Literal/Instance
name	"PERSON"
containing_diagram	Object203
characteristic_partial_description	Object234, Object235, Object236
relationship_end_group_partial_description	Object225, Object227, Object229
identifier_partial_description	Object277

<Entity_Type>
Object205

Attribute/Reference	Literal/Instance
name	"EMPLOYEE"
containing_diagram	Object203
classification_scheme_partial_description	Object211
characteristic_partial_description	Object237, Object238, Object239, Object240, Object241
relationship_end_group_partial_description	Object230, Object231
identifier_partial_description	Object278, Object 279

Figure D.4 — Registration of the IDEF1X example (Part 1 of 7)

<Entity_Type>
Object206

Attribute/Reference	Literal/Instance
name	"MECHANIC"
containing_diagram	Object203
containing_hierarchy	Object211
characteristic_partial_description	Object242, Object243
identifier_partial_description	Object280

<Entity_Type>
Object207

Attribute/Reference	Literal/Instance
name	"ADDRESS"
containing_diagram	Object203
characteristic_partial_description	Object244, Object245, Object246, Object247
relationship_end_group_partial_description	Object232
identifier_partial_description	Object281

<Entity_Type>
Object208

Attribute/Reference	Literal/Instance
name	"PERSON-PERSON-ASSOCIATION"
containing_diagram	Object203
classification_scheme_partial_description	Object212
characteristic_partial_description	Object248, Object249, Object250
relationship_end_group_partial_description	Object226, Object228
identifier_partial_description	Object282

<Entity_Type>
Object209

Attribute/Reference	Literal/Instance
name	"CHILD-ASSOCIATION"
containing_diagram	Object203
containing_hierarchy	Object212
characteristic_partial_description	Object251, Object252, Object253, Object254
identifier_partial_description	Object283

<Entity_Type>
Object210

Attribute/Reference	Literal/Instance
name	"SPOUSE-ASSOCIATION"
containing_diagram	Object203
containing_hierarchy	Object212
characteristic_partial_description	Object255, Object256, Object257, Object 258
identifier_partial_description	Object284

<Entity_Specialisation_Hierarchy>
Object211

Attribute/Reference	Literal/Instance
completeness_indicator	False
exclusivity_indicator	True
described_entity_type	Object205
subtype_entity_type	Object206
category_discriminator	Object241
partial_representation	Object242

<Entity_Specialisation_Hierarchy>
Object212

Attribute/Reference	Literal/Instance
completeness_indicator	True
exclusivity_indicator	True
described_entity_type	Object208
subtype_entity_type	Object209, Object210
category_discriminator	Object250
partial_representation	Object251, Object252, Object253, Object255, Object256, Object257

<Relationship>
Object213

Attribute/Reference	Literal/Instance
name	"associated-with"
containing_diagram	Object203
identifying_indicator	True
contained_relationship_end	Object217, Object218

<Relationship>
Object214

Attribute/Reference	Literal/Instance
name	"associated-to"
containing_diagram	Object203
identifying_indicator	True
contained_relationship_end	Object219, Object220

<Relationship>
Object215

Attribute/Reference	Literal/Instance
name	"employed_as"
containing_diagram	Object203
identifying_indicator	False
contained_relationship_end	Object221, Object222

<Relationship>
Object216

Attribute/Reference	Literal/Instance
name	"resident-at"
containing_diagram	Object203
identifying_indicator	False
contained_relationship_end	Object223, Object224

<Relationship_End>
Object217

Attribute/Reference	Literal/Instance
minimum_cardinality	"One"
maximum_cardinality	"One"
containing_relationship	Object213
containing_group	Object225

<Relationship_End>
Object218

Attribute/Reference	Literal/Instance
minimum_cardinality	"Zero"
maximum_cardinality	"Many"
containing_relationship	Object213
containing_group	Object226
partial_representation	Object248

<Relationship_End>
Object219

Attribute/Reference	Literal/Instance
minimum_cardinality	"One"
maximum_cardinality	"One"
containing_relationship	Object214
containing_group	Object227

<Relationship_End>
Object220

Attribute/Reference	Literal/Instance
minimum_cardinality	"Zero"
maximum_cardinality	"Many"
containing_relationship	Object214
containing_group	Object228
partial_representation	Object249

<Relationship_End>
Object221

Attribute/Reference	Literal/Instance
minimum_cardinality	"One"
maximum_cardinality	"One"
containing_relationship	Object215
containing_group	Object229

Figure D.4 — Registration of the IDEF1X example (Part 2 of 7)

<Relationship_End>
Object222

Attribute/Reference	Literal/Instance
minimum_cardinality	"Zero"
maximum_cardinality	"Many"
containing_relationship	Object215
containing_group	Object230
partial_representation	Object240

<Relationship_End>
Object223

Attribute/Reference	Literal/Instance
minimum_cardinality	"One"
maximum_cardinality	"Many"
containing_relationship	Object216
containing_group	Object231

<Relationship_End>
Object224

Attribute/Reference	Literal/Instance
minimum_cardinality	"Zero"
maximum_cardinality	"One"
containing_relationship	Object216
containing_group	Object232

<Relationship_End_Group>
Object225

Attribute/Reference	Literal/Instance
described_entity_type	Object204
mutually_exclusive_component	Object217

<Relationship_End_Group>
Object226

Attribute/Reference	Literal/Instance
described_entity_type	Object208
mutually_exclusive_component	Object218

<Relationship_End_Group>
Object227

Attribute/Reference	Literal/Instance
described_entity_type	Object204
mutually_exclusive_component	Object219

<Relationship_End_Group>
Object228

Attribute/Reference	Literal/Instance
described_entity_type	Object208
mutually_exclusive_component	Object220

<Relationship_End_Group>
Object229

Attribute/Reference	Literal/Instance
described_entity_type	Object204
mutually_exclusive_component	Object221

<Relationship_End_Group>
Object230

Attribute/Reference	Literal/Instance
described_entity_type	Object205
mutually_exclusive_component	Object222

<Relationship_End_Group>
Object231

Attribute/Reference	Literal/Instance
described_entity_type	Object205
mutually_exclusive_component	Object223

<Relationship_End_Group>
Object232

Attribute/Reference	Literal/Instance
described_entity_type	Object207
mutually_exclusive_component	Object224

<Native_Key_Attribute>
Object234

Attribute/Reference	Literal/Instance
name	"person-id"
minimum_cardinality	"One"
maximum_cardinality	"One"
described_entity_type	Object204
object	Object285

<Non_Key_Attribute>
Object235

Attribute/Reference	Literal/Instance
name	"person-name"
minimum_cardinality	"One"
maximum_cardinality	"One"
described_entity_type	Object204

<Non_Key_Attribute>
Object236

Attribute/Reference	Literal/Instance
name	"person-birth-date"
minimum_cardinality	"Zero"
maximum_cardinality	"One"
described_entity_type	Object204

<Native_Key_Attribute>
Object237

Attribute/Reference	Literal/Instance
name	"employee-staff-number"
minimum_cardinality	"One"
maximum_cardinality	"One"
described_entity_type	Object205
object	Object286

<Non_Key_Attribute>
Object236

Attribute/Reference	Literal/Instance
name	"employment-start-date"
minimum_cardinality	"One"
maximum_cardinality	"One"
described_entity_type	Object205

<Non_Key_Attribute>
Object239

Attribute/Reference	Literal/Instance
name	"employment-end-date"
minimum_cardinality	"Zero"
maximum_cardinality	"One"
described_entity_type	Object205

<Relationship_End_Foreign_Key_Attribute>
Object240

Attribute/Reference	Literal/Instance
name	"person-id"
minimum_cardinality	"One"
maximum_cardinality	"One"
rank	1
described_entity_type	Object205
object	Object287
represented_relationship_end	Object222

Figure D.4 — Registration of the IDEF1X example (Part 3 of 7)

<Non_Key_Attribute>
Object241

Attribute/Reference	Literal/Instance
name	"employee_type"
minimum_cardinality	"One"
maximum_cardinality	"One"
described_entity_type	Object205
constraint	Object261
discriminated_hierarchy	Object211

<Entity_Specialisation_Hierarchy_Foreign_Key_Attribute>
Object242

Attribute/Reference	Literal/Instance
name	"employee-staff-number"
minimum_cardinality	"One"
maximum_cardinality	"One"
rank	1
described_entity_type	Object206
object	Object288
represented_hierarchy	Object211

<Non_Key_Attribute>
Object243

Attribute/Reference	Literal/Instance
name	"grade"
minimum_cardinality	"One"
maximum_cardinality	"One"
described_entity_type	Object206
constraint	Object262

<Native_Key_Attribute>
Object244

Attribute/Reference	Literal/Instance
name	"address-id"
minimum_cardinality	"One"
maximum_cardinality	"One"
described_entity_type	Object207
object	Object289

<Non_Key_Attribute>
Object245

Attribute/Reference	Literal/Instance
name	"address"
minimum_cardinality	"One"
maximum_cardinality	"One"
described_entity_type	Object207

<Non_Key_Attribute>
Object246

Attribute/Reference	Literal/Instance
name	"town"
minimum_cardinality	"One"
maximum_cardinality	"One"
described_entity_type	Object207

<Non_Key_Attribute>
Object247

Attribute/Reference	Literal/Instance
name	"post-code"
minimum_cardinality	"One"
maximum_cardinality	"One"
described_entity_type	Object207

<Relationship_End_Foreign_Key_Attribute>
Object248

Attribute/Reference	Literal/Instance
name	"associated-with-person-id"
minimum_cardinality	"One"
maximum_cardinality	"One"
rank	1
described_entity_type	Object208
object	Object290
represented_relationship_end	Object218

<Relationship_End_Foreign_Key_Attribute>
Object249

Attribute/Reference	Literal/Instance
name	"associated-to-person-id"
minimum_cardinality	"One"
maximum_cardinality	"One"
rank	1
described_entity_type	Object208
object	Object291
represented_relationship_end	Object220

<Native_Key_Attribute>
Object250

Attribute/Reference	Literal/Instance
name	"association-type"
minimum_cardinality	"One"
maximum_cardinality	"One"
described_entity_type	Object208
constraint	Object259
object	Object292
discriminated_hierarchy	Object212

<Entity_Specialisation_Hierarchy_Foreign_Key_Attribute>
Object251

Attribute/Reference	Literal/Instance
name	"associated-with-person-mother-id"
minimum_cardinality	"One"
maximum_cardinality	"One"
rank	1
described_entity_type	Object209
object	Object293
represented_hierarchy	Object212

<Entity_Specialisation_Hierarchy_Foreign_Key_Attribute>
Object252

Attribute/Reference	Literal/Instance
name	"associated-to-person-father-id"
minimum_cardinality	"One"
maximum_cardinality	"One"
rank	2
described_entity_type	Object209
object	Object294
represented_hierarchy	Object212

<Entity_Specialisation_Hierarchy_Foreign_Key_Attribute>
Object253

Attribute/Reference	Literal/Instance
name	"association-type"
minimum_cardinality	"One"
maximum_cardinality	"One"
rank	3
described_entity_type	Object209
constraint	Object259
object	Object295
represented_hierarchy	Object212

Figure D.4 — Registration of the IDEF1X example (Part 4 of 7)

<Non_Key_Attribute>
Object254

Attribute/Reference	Literal/Instance
name	"child-name"
minimum_cardinality	"One"
maximum_cardinality	"One"
described_entity_type	Object209

<Entity_Specialisation_Hierarchy_Foreign_Key_Attribute>
Object255

Attribute/Reference	Literal/Instance
name	"associated-with-person-wife-id"
minimum_cardinality	"One"
maximum_cardinality	"One"
rank	1
described_entity_type	Object210
object	Object296
represented_hierarchy	Object212

<Entity_Specialisation_Hierarchy_Foreign_Key_Attribute>
Object256

Attribute/Reference	Literal/Instance
name	"associated-to-person-husband-id"
minimum_cardinality	"One"
maximum_cardinality	"One"
rank	2
described_entity_type	Object210
object	Object297
represented_hierarchy	Object212

<Entity_Specialisation_Hierarchy_Foreign_Key_Attribute>
Object257

Attribute/Reference	Literal/Instance
name	"association-type"
minimum_cardinality	"One"
maximum_cardinality	"One"
rank	3
described_entity_type	Object210
constraint	Object259
object	Object298
represented_hierarchy	Object212

<Non_Key_Attribute>
Object258

Attribute/Reference	Literal/Instance
name	"association-status"
minimum_cardinality	"One"
maximum_cardinality	"One"
described_entity_type	Object210
constraint	Object260

<Enumerated_Domain>
Object259

Attribute/Reference	Literal/Instance
name	"association-type"
constrained_attribute	Object250, Object253, Object257
constrained_value	Object263, Object264

<Enumerated_Domain>
Object260

Attribute/Reference	Literal/Instance
name	"association-status"
constrained_attribute	Object258
constrained_value	Object265, Object266, Object267 Object268, Object269, Object270 Object271

<Enumerated_Domain>
Object261

Attribute/Reference	Literal/Instance
name	"employee-type"
constrained_attribute	Object241
constrained_value	Object272, Object273

<Enumerated_Domain>
Object262

Attribute/Reference	Literal/Instance
name	"grade"
constrained_attribute	Object243
constrained_value	Object274, Object275, Object276

<Valid_Value>
Object263

Attribute/Reference	Literal/Instance
literal	"child-association"
containing_domain	Object259

<Valid_Value>
Object264

Attribute/Reference	Literal/Instance
literal	"spouse-association"
containing_domain	Object259

<Valid_Value>
Object265

Attribute/Reference	Literal/Instance
literal	"current-marriage"
containing_domain	Object260

<Valid_Value>
Object266

Attribute/Reference	Literal/Instance
literal	"current-civil-partnership"
containing_domain	Object260

<Valid_Value>
Object267

Attribute/Reference	Literal/Instance
literal	"current-cohabitation"
containing_domain	Object260

<Valid_Value>
Object268

Attribute/Reference	Literal/Instance
literal	"separated"
containing_domain	Object260

<Valid_Value>
Object269

Attribute/Reference	Literal/Instance
literal	"divorced"
containing_domain	Object260

<Valid_Value>
Object270

Attribute/Reference	Literal/Instance
literal	"wife-deceased"
containing_domain	Object260

<Valid_Value>
Object271

Attribute/Reference	Literal/Instance
literal	"husband-deceased"
containing_domain	Object260

Figure D.4 — Registration of the IDEF1X example (Part 5 of 7)

<Valid_Value>
Object272

Attribute/Reference	Literal/Instance
literal	"mechanic"
containing_domain	Object261

<Valid_Value>
Object273

Attribute/Reference	Literal/Instance
literal	"other-employee"
containing_domain	Object261

<Valid_Value>
Object274

Attribute/Reference	Literal/Instance
literal	"trainee-mechanic"
containing_domain	Object262

<Valid_Value>
Object275

Attribute/Reference	Literal/Instance
literal	"trained-mechanic"
containing_domain	Object262

<Valid_Value>
Object276

Attribute/Reference	Literal/Instance
literal	"lead-mechanic"
containing_domain	Object262

<Unique_ Identifier>
Object277

Attribute/Reference	Literal/Instance
primary_indicator	True
described_entity_type	Object204
identifier_element_partial_ description	Object285

<Unique_ Identifier>
Object278

Attribute/Reference	Literal/Instance
primary_indicator	True
described_entity_type	Object205
identifier_element_partial_ description	Object286

<Unique_ Identifier>
Object279

Attribute/Reference	Literal/Instance
primary_indicator	False
described_entity_type	Object205
identifier_element_partial_ description	Object287

<Unique_ Identifier>
Object280

Attribute/Reference	Literal/Instance
primary_indicator	True
described_entity_type	Object206
identifier_element_partial_ description	Object288

<Unique_ Identifier>
Object281

Attribute/Reference	Literal/Instance
primary_indicator	True
described_entity_type	Object207
identifier_element_partial_ description	Object289

<Unique_ Identifier>
Object282

Attribute/Reference	Literal/Instance
primary_indicator	True
described_entity_type	Object208
identifier_element_partial_ description	Object290, Object291, Object292

<Unique_ Identifier>
Object283

Attribute/Reference	Literal/Instance
primary_indicator	True
described_entity_type	Object209
identifier_element_partial_ description	Object293, Object294, Object295

<Unique_ Identifier>
Object284

Attribute/Reference	Literal/Instance
primary_indicator	True
described_entity_type	Object210
identifier_element_partial_ description	Object296, Object297, Object298

<Attribute_Unique_ Identifier_Element>
Object285

Attribute/Reference	Literal/Instance
containing_unique_identifier	Object277
subject_attribute	Object234

<Attribute_Unique_ Identifier_Element>
Object286

Attribute/Reference	Literal/Instance
containing_unique_identifier	Object278
subject_attribute	Object237

<Attribute_Unique_ Identifier_Element>
Object287

Attribute/Reference	Literal/Instance
containing_unique_identifier	Object279
subject_attribute	Object240

<Attribute_Unique_ Identifier_Element>
Object288

Attribute/Reference	Literal/Instance
containing_unique_identifier	Object280
subject_attribute	Object242

<Attribute_Unique_ Identifier_Element>
Object289

Attribute/Reference	Literal/Instance
containing_unique_identifier	Object281
subject_attribute	Object244

Figure D.4 — Registration of the IDEF1X example (Part 6 of 7)

<Attribute_Unique_ Identifier_Element>
Object290

Attribute/Reference	Literal/Instance
containing_unique_identifier	Object282
subject_attribute	Object248

<Attribute_Unique_ Identifier_Element>
Object291

Attribute/Reference	Literal/Instance
containing_unique_identifier	Object282
subject_attribute	Object249

<Attribute_Unique_ Identifier_Element>
Object292

Attribute/Reference	Literal/Instance
containing_unique_identifier	Object282
subject_attribute	Object250

<Attribute_Unique_ Identifier_Element>
Object293

Attribute/Reference	Literal/Instance
containing_unique_identifier	Object283
subject_attribute	Object251

<Attribute_Unique_ Identifier_Element>
Object294

Attribute/Reference	Literal/Instance
containing_unique_identifier	Object283
subject_attribute	Object252

<Attribute_Unique_ Identifier_Element>
Object295

Attribute/Reference	Literal/Instance
containing_unique_identifier	Object283
subject_attribute	Object253

<Attribute_Unique_ Identifier_Element>
Object296

Attribute/Reference	Literal/Instance
containing_unique_identifier	Object284
subject_attribute	Object255

<Attribute_Unique_ Identifier_Element>
Object297

Attribute/Reference	Literal/Instance
containing_unique_identifier	Object284
subject_attribute	Object256

<Attribute_Unique_ Identifier_Element>
Object298

Attribute/Reference	Literal/Instance
containing_unique_identifier	Object284
subject_attribute	Object257

Figure D.4 — Registration of the IDEF1X example (Part 7 of 7)

D.4 Ellis-Barker example

This example is based on a vehicle servicing scenario which is drawn in Ellis-Barker notation (see Figure D.5). Figure D.6 provides the object instances to illustrate the registration of this model.

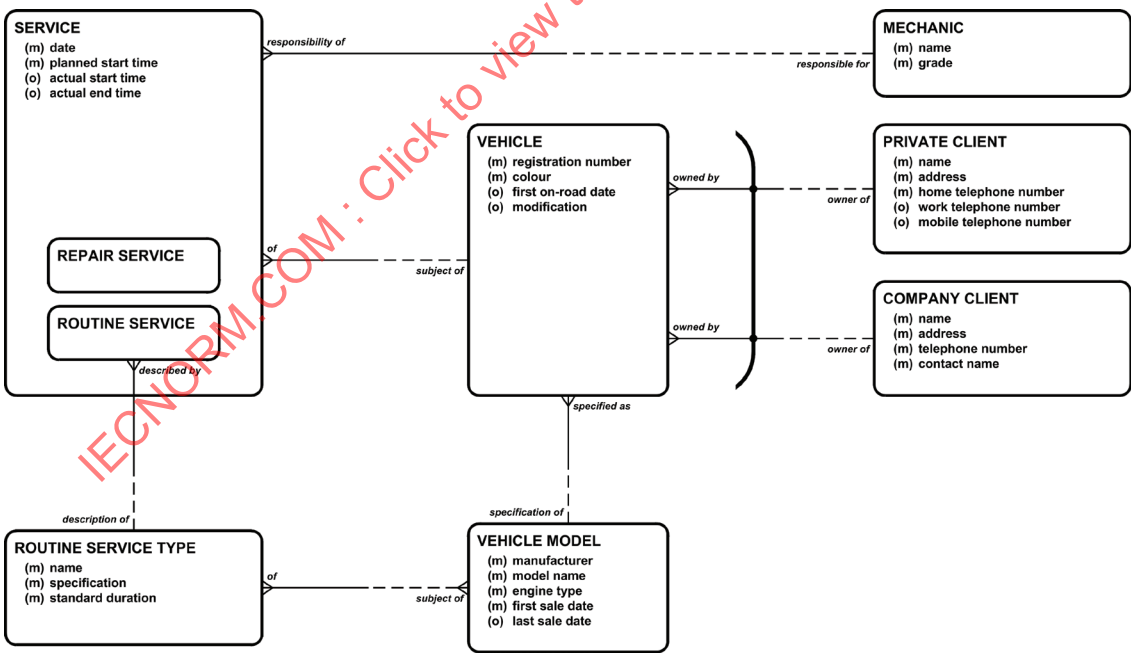

Figure D.5 — Example information model drawn in 'Ellis-Barker' notation

<Information_Modelling_Language>
Object301

Attribute/Reference	Literal/Instance
name	"E-R (Ellis-Barker)"
expressed_model	Object302

<Information_Model>
Object302

Attribute/Reference	Literal/Instance
name	"Robinson Motors Vehicle Service Recording Ver 1.0"
describing_language	Object301
diagram_model_element	Object303

<Diagram>
Object303

Attribute/Reference	Literal/Instance
name	"Robinson Motors Vehicle Service Recording Ver 1.0 Diagram 1"
containing_model	Object302
entity_type_model_element	Object304, Object310, Object315, Object318, Object324, Object329, Object333, Object339, Object341
relationship_model_element	Object365, Object369, Object374, Object379, Object384, Object389, Object394

<Entity_Type>
Object304

Attribute/Reference	Literal/Instance
name	"PRIVATE CLIENT"
description	"has significance as a client of Robinson Motors who is the personal owner of a vehicle serviced by Robinson Motors"
containing_diagram	Object303
characteristic_partial_description	Object305, Object306, Object307, Object308, Object309
relationship_end_group_partial_description	Object363

<Non_Key_Attribute>
Object305

Attribute/Reference	Literal/Instance
name	"name"
minimum_cardinality	"One"
maximum_cardinality	"One"
described_entity_type	Object304
constraint	Object342

<Non_Key_Attribute>
Object306

Attribute/Reference	Literal/Instance
name	"address"
minimum_cardinality	"One"
maximum_cardinality	"One"
described_entity_type	Object304
constraint	Object343

<Non_Key_Attribute>
Object307

Attribute/Reference	Literal/Instance
name	"home telephone number"
minimum_cardinality	"One"
maximum_cardinality	"One"
described_entity_type	Object304
constraint	Object344

<Non_Key_Attribute>
Object308

Attribute/Reference	Literal/Instance
name	"work telephone number"
minimum_cardinality	"Zero"
maximum_cardinality	"One"
described_entity_type	Object304
constraint	Object344

<Non_Key_Attribute>
Object309

Attribute/Reference	Literal/Instance
name	"mobile telephone number"
minimum_cardinality	"Zero"
maximum_cardinality	"One"
described_entity_type	Object304
constraint	Object344

<Entity_Type>
Object310

Attribute/Reference	Literal/Instance
name	"COMPANY CLIENT"
description	"has significance as a client of Robinson Motors who is the company owner of a vehicle serviced by Robinson Motors"
containing_diagram	Object303
characteristic_partial_description	Object311, Object312, Object313, Object314
relationship_end_group_partial_description	Object371

<Non_Key_Attribute>
Object311

Attribute/Reference	Literal/Instance
name	"name"
minimum_cardinality	"One"
maximum_cardinality	"One"
described_entity_type	Object310
constraint	Object345

<Non_Key_Attribute>
Object312

Attribute/Reference	Literal/Instance
name	"address"
minimum_cardinality	"One"
maximum_cardinality	"One"
described_entity_type	Object310
constraint	Object343

<Non_Key_Attribute>
Object313

Attribute/Reference	Literal/Instance
name	"telephone number"
minimum_cardinality	"One"
maximum_cardinality	"One"
described_entity_type	Object310
constraint	Object344

<Non_Key_Attribute>
Object314

Attribute/Reference	Literal/Instance
name	"contact name"
minimum_cardinality	"One"
maximum_cardinality	"One"
described_entity_type	Object310
constraint	Object342

Figure D.6 — Registration of the Ellis-Barker example (Part 1 of 6)

<Entity_Type>
Object315

Attribute/Reference	Literal/Instance
name	"MECHANIC"
description	"has significance as an employee of Robinson Motors who is qualified to repair vehicles"
containing_diagram	Object303
characteristic_partial_description	Object316, Object317
relationship_end_group_partial_description	Object396

<Non_Key_Attribute>
Object316

Attribute/Reference	Literal/Instance
name	"name"
minimum_cardinality	"One"
maximum_cardinality	"One"
described_entity_type	Object315
constraint	Object342

<Non_Key_Attribute>
Object317

Attribute/Reference	Literal/Instance
name	"grade"
minimum_cardinality	"One"
maximum_cardinality	"One"
described_entity_type	Object315
constraint	Object346

<Entity_Type>
Object318

Attribute/Reference	Literal/Instance
name	"VEHICLE MODEL"
description	"has significance as the specification of one or more of the vehicles serviced by Robinson Motors"
containing_diagram	Object303
characteristic_partial_description	Object319, Object320, Object321, Object322, Object323
relationship_end_group_partial_description	Object376, Object377

<Non_Key_Attribute>
Object319

Attribute/Reference	Literal/Instance
name	"manufacturer"
minimum_cardinality	"One"
maximum_cardinality	"One"
described_entity_type	Object318
constraint	Object345

<Non_Key_Attribute>
Object320

Attribute/Reference	Literal/Instance
name	"model name"
minimum_cardinality	"One"
maximum_cardinality	"One"
described_entity_type	Object318
constraint	Object350

<Non_Key_Attribute>
Object321

Attribute/Reference	Literal/Instance
name	"engine type"
minimum_cardinality	"One"
maximum_cardinality	"One"
described_entity_type	Object318
constraint	Object351

<Non_Key_Attribute>
Object322

Attribute/Reference	Literal/Instance
name	"first sale date"
minimum_cardinality	"One"
maximum_cardinality	"One"
described_entity_type	Object318
constraint	Object354

<Non_Key_Attribute>
Object323

Attribute/Reference	Literal/Instance
name	"last sale date"
minimum_cardinality	"Zero"
maximum_cardinality	"One"
described_entity_type	Object318
constraint	Object354

<Entity_Type>
Object324

Attribute/Reference	Literal/Instance
name	"VEHICLE"
description	"has significance as a vehicle serviced by Robinson Motors"
containing_diagram	Object303
characteristic_partial_description	Object325, Object326, Object327, Object328
relationship_end_group_partial_description	Object367, Object372, Object391

<Non_Key_Attribute>
Object325

Attribute/Reference	Literal/Instance
name	"registration number"
minimum_cardinality	"One"
maximum_cardinality	"One"
described_entity_type	Object324
constraint	Object355

<Non_Key_Attribute>
Object326

Attribute/Reference	Literal/Instance
name	"colour"
minimum_cardinality	"One"
maximum_cardinality	"One"
described_entity_type	Object324
constraint	Object356

<Non_Key_Attribute>
Object327

Attribute/Reference	Literal/Instance
name	"first on-road date"
minimum_cardinality	"Zero"
maximum_cardinality	"One"
described_entity_type	Object324
constraint	Object354

<Non_Key_Attribute>
Object328

Attribute/Reference	Literal/Instance
name	"modification"
minimum_cardinality	"Zero"
maximum_cardinality	"One"
described_entity_type	Object324
constraint	Object357

Figure D.6 — Registration of the Ellis-Barker example (Part 2 of 6)