

**ISO/IEC JTC 1/SC 32/WG 2 N1579**

Date: 2011-09-21

**ISO/IEC WD 19763-12**

ISO/IEC JTC 1/SC 32/WG 2

Secretariat: ANSI

**Information technology – Metamodel framework for interoperability (MFI)  
– Part 12: Metamodel for information model registration**

**Warning**

This document is not an ISO International Standard. It is distributed for review and comment. It is subject to change without notice and may not be referred to as an International Standard.

Recipients of this draft are invited to submit, with their comments, notification of any relevant patent rights of which they are aware and to provide supporting documentation.

### Copyright notice

This ISO document is a working draft or committee draft and is copyright-protected by ISO. While the reproduction of working drafts or committee drafts in any form for use by participants in the ISO standards development process is permitted without prior permission from ISO, neither this document nor any extract from it may be reproduced, stored or transmitted in any form for any other purpose without prior written permission from ISO.

Requests for permission to reproduce this document for the purpose of selling it should be addressed as shown below or to ISO's member body in the country of the requester:

ISO copyright office  
Case postale 56 • CH-1211 Geneva 20  
Tel. + 41 22 749 01 11  
Fax + 41 22 749 09 47 E-mail [copyright@iso.ch](mailto:copyright@iso.ch)  
Web [www.iso.ch](http://www.iso.ch)

Reproduction for sales purposes may be subject to royalty payments or a licensing agreement.

Violators may be prosecuted.

# Contents

Page

Foreword .....	v
Introduction.....	vi
<b>1 Scope .....</b>	<b>1</b>
<b>2 Conformance .....</b>	<b>1</b>
<b>2.1 General .....</b>	<b>1</b>
<b>2.2 Degree of conformance .....</b>	<b>1</b>
<b>2.2.1 General .....</b>	<b>1</b>
<b>2.2.2 Strictly conforming implementation .....</b>	<b>2</b>
<b>2.2.3 Conforming implementation.....</b>	<b>2</b>
<b>2.3 Implementation Conformance Statement (ICS).....</b>	<b>2</b>
<b>3 Normative references .....</b>	<b>2</b>
<b>4 Terms, definitions and abbreviated terms .....</b>	<b>3</b>
<b>4.1 Terms and definitions .....</b>	<b>3</b>
<b>4.2 Terms for concepts used in this part of 17963.....</b>	<b>3</b>
<b>4.3 Abbreviated terms .....</b>	<b>6</b>
<b>5 Structure of MFI Information model registration .....</b>	<b>6</b>
<b>5.1 Overview of MFI Information model registration .....</b>	<b>6</b>
<b>5.2 Metaclasses in MFI Information Model Registration .....</b>	<b>8</b>
<b>5.2.1 InformationModellingLanguage .....</b>	<b>8</b>
<b>5.2.2 InformationModel .....</b>	<b>9</b>
<b>5.2.3 EntityType .....</b>	<b>9</b>
<b>5.2.4 EntityTypeAlias.....</b>	<b>10</b>
<b>5.2.5 EntitySpecialisationHierarchy.....</b>	<b>10</b>
<b>5.2.6 EntitySubtype .....</b>	<b>11</b>
<b>5.2.7 Relationship .....</b>	<b>12</b>
<b>5.2.8 RelationshipEnd .....</b>	<b>12</b>
<b>5.2.9 RelationshipEndGroup .....</b>	<b>14</b>
<b>5.2.10 Attribute.....</b>	<b>14</b>
<b>5.2.11 NonKeyAttribute .....</b>	<b>15</b>
<b>5.2.12 KeyAttribute .....</b>	<b>15</b>
<b>5.2.13 NativeKeyAttribute .....</b>	<b>16</b>
<b>5.2.14 ForeignKeyAttribute .....</b>	<b>16</b>
<b>5.2.15 RelationshipEndForeignKeyAttribute .....</b>	<b>16</b>
<b>5.2.16 EntitySpecialisationHierarchyForeignKeyAttribute.....</b>	<b>17</b>
<b>5.2.17 Domain.....</b>	<b>17</b>
<b>5.2.18 ContinuousDomain .....</b>	<b>18</b>
<b>5.2.19 DiscreteDomain .....</b>	<b>18</b>
<b>5.2.20 ValidValue .....</b>	<b>18</b>
<b>5.2.21 UniquelIdentifier .....</b>	<b>19</b>
<b>5.2.22 UniquelIdentifierElement .....</b>	<b>19</b>
<b>5.2.23 AttributeUniquelIdentifierElement .....</b>	<b>20</b>
<b>5.2.24 RelationshipEndUniquelIdentifierElement.....</b>	<b>20</b>
<b>Annex A (informative) Description of the metamodel .....</b>	<b>21</b>
<b>Annex B (informative) Use of information modelling concepts within techniques .....</b>	<b>27</b>
<b>Annex C (informative) Examples of information model registration .....</b>	<b>28</b>
<b>Annex D (informative) Bibliography .....</b>	<b>37</b>

**Figures**

Figure 1 - Metamodel of MFI Information model registration..... 7

Figure A.1 - Metamodel of MFI Information model registration (repeated) ..... 21

Figure C.1 – Example information model drawn in EXPRESS-G ..... 28

Figure C.2 – Example information model drawn in IDEF1X ..... 29

Figure C.3 – Example information model drawn in 'Ellis-Barker' notation ..... 29

Figure C.4 – Example information model drawn as a UML Class Diagram ..... 30

**Tables**

Table B.1 – Use of concepts within techniques ..... 27

Table C.1 – Example 'InformationModellingLanguage' table ..... 30

Table C.2 – Example 'InformationModel' table ..... 30

Table C.3 – Example 'EntityType' table ..... 30

Table C.4 – Example 'EntityTypeAlias' table ..... 31

Table C.5 – Example 'EntitySpecialisationHierarchy' table ..... 31

Table C.6 – Example 'EntitySubtype' table ..... 31

Table C.7 – Example 'Relationship' table ..... 32

Table C.8 – Example 'RelationshipEnd' table ..... 32

Table C.9 – Example 'RelationshipEndGroup' table ..... 33

Table C.10 – Example 'Attribute' table ..... 33

Table C.11 – Example 'Domain' table ..... 35

Table C.12 – Example 'ValidValue' table ..... 35

Table C.13 – Example 'UniquelIdentifier' table ..... 36

Table C.14 – Example 'Unique Identifier Element' table ..... 36

## Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.

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

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

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

ISO/IEC 19763-12 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information Technology*, Subcommittee 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: Reference model*

*Part 2: Core model and model mapping*

*Part 3: Metamodel for ontology registration*

*Part 5: Metamodel for process model registration*

*Part 6: Registry Summary*

*Part 7: Metamodel for service registration*

*Part 8: Metamodel for role and goal registration*

*Part 9: On Demand Model Selection (ODMS) [Technical Report]*

*Part 11: Advanced Mapping [Technical Report]*

*Part 12: Metamodel for information model 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.

This part of ISO/IEC 19763 intends to provide a generic framework for registering these information models so that the mappings between models can be recognised and registered using the facilities specified in ISO/IEC 19763 Part 2.

# 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 information models developed using a number of different industry standard diagramming techniques and notations.

These techniques were:

- Express-G, an ISO standard.
- IDEF1X, a US Federal standard.
- The 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).
- UML Class Diagrams.
- Entity-Relationship Modelling, as described by Peter Chen.
- Information Engineering, as described by James Martin and Clive Finkelstein.

An information model can be used to represent the information requirements that are met by a system. 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 Part 2 of ISO/IEC 19763. An interface between the two systems can then be designed, enabling the two systems to interoperate.

## 2 Conformance

### 2.1 General

An implementation claiming conformance with this part of ISO/IEC 19763 shall support the metamodel specified in 5.3, depending on a 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 needs of users, vendors, institutions and industries, but are not specified by this part of ISO/IEC 19763.

A strictly conforming implementation may be limited in usefulness but is maximally interoperable with respect to this part of ISO/IEC 19763. A conforming implementation may be more useful, but may 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 5.1;
- b) shall not support any extensions to the metamodel specified in 5.1.

### 2.2.3 Conforming implementation

A conforming implementation

- a) shall support the metamodel specified in 5.1;
- b) may support extensions to the metamodel specified in 5.1 that are consistent with the metamodel specified in 5.1.

## 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 or a conforming implementation (2.2);
- b) what extensions are supported if it is a conforming implementation.

## 3 Normative references

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

ISO/IEC 19763-1, Information technology – Metamodel framework for interoperability (MFI) – Part 1: Reference model

ISO/IEC 19763-2, Information technology – Metamodel framework for interoperability (MFI) – Part 2: Core model and model mapping

ISO/IEC 11179-3:[Ed 3 - date to follow], Information technology – Metadata registries (MDR) – Part 3: Registry metamodel and basic attributes

[Others to follow?]

## 4 Terms, definitions and abbreviated terms

### 4.1 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO/IEC 19763-1, ISO/IEC 11179-3:[Ed 3 - date to follow] and the following apply.

### 4.2 Terms for concepts used in this part of 19763

#### 4.2.1

##### **attribute**

detail that serves to qualify, identify, classify, quantify or express the state of an instance of an entity type

#### 4.2.2

##### **attribute unique identifier element**

unique identifier element that is a statement that a particular attribute is a part of a particular unique identifier

#### 4.2.3

##### **cardinality**

measure of the number of elements of a set

#### 4.2.4

##### **collection**

grouping of elements of the same type

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

#### 4.2.5

##### **continuous domain**

domain whose values are not drawn from an explicit list of valid values

NOTE may also be known as a *non-enumerated domain*

#### 4.2.6

##### **discrete domain**

domain whose permitted values consist of an explicit list of valid values

NOTE may also be known as an *enumerated domain*

#### 4.2.7

##### **domain**

pool of values from which an attribute (or attributes) must take its (or their) value

NOTE a domain provides a set of business validation rules, format constraints and other properties for one or more attributes

#### 4.2.8

##### **entity role**

role that an entity type is playing in a relationship

#### 4.2.9

##### **entity specialisation hierarchy**

means by which instances of an entity type may be classified or specialised.

NOTE may also be known as an *entity generalisation hierarchy*, an *entity subtype hierarchy*, an *entity type classification* or an *entity classification*

**4.2.10**

**entity subtype**

subset of the instances of an entity type (the supertype) that share common attributes and/or relationships distinct from other subsets.

NOTE may also be known as an *subtype*, an *object subclass* or a *subclass*

**4.2.11**

**entity**

concept or thing of significance about which information may need to be held in support of business operations

NOTE this is often known, incorrectly, as an *entity instance*

**4.2.11**

**entity type**

class of entities which have some characteristics in common

NOTE this is often known, incorrectly, as an *entity*

NOTE may also be known as an *object class*

**4.2.12**

**entity type alias**

one of a set of alternative names for an entity type

NOTE may also be known as an *entity type synonym*

**4.2.13**

**foreign key attribute**

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

NOTE a foreign key attribute provides or contributes to an alternative representation of the relationship concerned. Its value must be drawn from the domain of the corresponding key attribute of the related entity type

**4.2.14**

**information model**

set of definitions for things of significance to an organisation, about which information needs to be held, and the relationships between them

NOTE may also be known as a *data model*, a *conceptual data model*, a *logical data model*, an *entity relationship model* or an *object class diagram*

**4.2.15**

**information modelling language**

language or notation that is used to model information requirements

**4.2.16**

**key attribute**

attribute whose value contributes in some way to the identification of individual instances of the host entity type or of some related entity type

**4.2.17**

**link phrase**

statement that explains the nature, expressed in business terms, of a relationship from the perspective of one of the associated entity types

**4.2.18****native key attribute**

key attribute whose value contributes in some way to the identification of individual instances of the host entity type

**4.2.19****non key attribute**

attribute whose value is fully independent of all relationships or other attributes

**4.2.20****relationship**

association between two or more entity types, or between one entity type and itself. Each relationship may have a name and may have an identifying indicator

NOTE may also be known as an *association*

**4.2.21****relationship end**

part of the definition of a relationship as seen from a given entity type (the host)

NOTE may also be known as an *association end*

**4.2.22****relationship end group**

statement that links one or more relationship ends to their host entity type such that they are mutually exclusive

NOTE the most common case is where the 'group' comprises just one relationship end

**4.2.23****relationship end unique identifier element**

unique identifier element that is a statement that a particular relationship end is a part of a particular unique identifier

**4.2.24****significance statement**

statement that explains the significance of this model element to the business and or organisation that is the subject of this Information Model

NOTE may also be known as a description

**4.2.25****unique identifier**

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

NOTE may also be known as a *key* or a *unique key*

**4.2.26****unique identifier element**

statement that a particular attribute or a particular relationship end is a part of a particular unique identifier

**4.2.27****valid value**

one of the explicit set of permitted values that comprise a discrete domain

**4.2.28****validation rule**

statement of the validation that may be applied to a continuous domain

NOTE this may be a reference to a data type to be applied to attributes, a range of values, or a 'format mask', or any expression that constrains the domain

### 4.3 Abbreviated terms

#### **MFI Core and model mapping**

ISO/IEC 19763-2, Information technology – Metamodel Framework for Interoperability – Part-2: Core model and model mapping

#### **MDR Metamodel**

ISO/IEC 11179-3:[Ed 3 - date to follow], Information technology – Metadata registries (MDR) – Part 3: Registry metamodel and basic attributes

[Others to follow?]

## 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 industry standard diagramming techniques and notations listed in Clause 1 above.

The metamodel for information model registration comprises the following metaclasses:

- **InformationModellingLanguage**
- **InformationModel**
- **EntityType**
- **EntityTypeAlias**
- **EntitySpecialisationHierarchy**
- **EntitySubtype**
- **Relationship**
- **RelationshipEnd**
- **RelationshipEndGroup**
- **Attribute**
- **NonKeyAttribute**, a subclass of **Attribute**
- **KeyAttribute**, a subclass of **Attribute**
- **NativeKeyAttribute**, a subclass of **KeyAttribute**
- **ForeignKeyAttribute**, a subclass of **KeyAttribute**
- **RelationshipEndForeignKeyAttribute**, a subclass of **ForeignKeyAttribute**
- **EntitySpecialisationHierarchyForeignKeyAttribute**, a subclass of **ForeignKeyAttribute**



The metamodel is described in detail in Annex A (informative). Detailed specifications of the metaclasses are provided in 5.2 below.

As explained in ISO/IEC 19763 Part 2, instances of the metaclasses defined in this part of ISO/IEC 19763 are to be extended by the types defined in the MDR Metamodel as follows:

- **InformationModellingLanguage** is to be extended as an **Identified\_Item** and as a **Designatable\_Item**.
- **InformationModel** is to be extended as an **Administered\_Item** and as a **Designatable\_Item**.
- **EntityType** is to be extended as an **Attached\_Item** and as a **Designatable\_Item**.
- **EntityTypeAlias** is to be extended as an **Attached\_Item** and as a **Designatable\_Item**.
- **EntitySpecialisationHierarchy** is to be extended as an **Attached\_Item** and as a **Designatable\_Item**.
- **EntitySubtype** is to be extended as an **Attached\_Item**.
- **Relationship** is to be extended as an **Attached\_Item** and as a **Designatable\_Item**.
- **RelationshipEnd** is to be extended as an **Attached\_Item**.
- **RelationshipEndGroup** is to be extended as an **Attached\_Item**.
- **Attribute** (and its subclasses) is to be extended as an **Attached\_Item** and as a **Designatable\_Item**.
- **Domain** (and its subclasses) is to be extended as an **Attached\_Item** and as a **Designatable\_Item**.
- **ValidValue** is to be extended as an **Attached\_Item** and as a **Designatable\_Item**.
- **UniquelIdentifier** is to be extended as an **Attached\_Item**.
- **UniquelIdentifierElement** (and its subclasses) is to be extended as an **Attached\_Item**.

For completeness of explanation, all attributes are included. In some cases these attributes will be provided through the extension of the metaclasses by the types defined in the MDR Metamodel.

## 5.2 Metaclasses in MFI Information Model Registration

### 5.2.1 InformationModellingLanguage

InformationModellingLanguage is a metaclass each instance of which represents a language or notation that is used to model information requirements.

#### Superclass

ModellingLanguage (defined in MFI Core and model mapping)

Attribute	DataType	Multiplicity	Description
name	String	1..1	A unique name by which this information modelling language is known.
Reference	Class	Multiplicity	Description
expresses	InformationModel	0..*	The set of information models that are expressed in this language.

### 5.2.2 InformationModel

InformationModel is a metaclass each instance of which represents a set of definitions for things of significance to an organisation, about which information needs to be held, and the relationships between them

#### Superclass

Model (defined in MFI Core and model mapping)

#### Aliases

Data Model, Conceptual Data Model, Logical Data Model, Entity Relationship Model, Object Class Diagram

Attribute	DataType	Multiplicity	Description
name	String	1..1	A unique name by which this information model is known.
Reference	Class	Multiplicity	Description
languageFor	InformationModelling Language	1..1	The information modelling language in which this model is expressed.
entityTypesWithin	EntityType	1..*	The set of entity types which comprise this information model.
relationshipsWithin	Relationship	1..*	The set of relationships which comprise this information model.

### 5.2.3 EntityType

EntityType is a metaclass each instance of which represents a concept or thing of significance about which information may need to be held in support of business operations.

#### Superclass

ModelElement (defined in MFI Core and model mapping)

#### Aliases

Entity, Object Class

Attribute	DataType	Multiplicity	Description
name	String	1..1	A unique name by which this entity type is primarily known.
significance Statement	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. This may also be known as a description.

Reference	Class	Multiplicity	Description
within	InformationModel	1..1	The information model of which includes this entity type.
alsoKnownAs	EntityTypeAlias	0..*	The set of entity type aliases that provides aliases or synonyms for this entity type.
specialised Through	EntitySpecialisation Hierarchy	0..*	The set of entity specialisation hierarchies that are used to classify instances of this entity type.
usedAs	EntitySubtype	0..*	The set of entity subtypes within entity specialisation hierarchies for other entity types which are roles played by this entity type.
describedBy	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.
identifiedBy	UniquelIdentifier	0..*	The set of unique identifiers that uniquely identify an instance of this entity type
relatedThrough	RelationshipEndGroup	0..*	The set of relationship end groups each of which has this entity type as their host entity type.

### 5.2.4 EntityTypeAlias

EntityTypeAlias is a metaclass each instance of which represents one of a set of alternative names for an entity type.

#### Superclass

ModelElement (defined in MFI Core and model mapping)

#### Aliases

Entity Alias, Entity Type Synonym, Entity Synonym, Object Class Alias, Object Class Synonym

Attribute	DataType	Multiplicity	Description
name	String	1..1	A name by which the associated entity type is alternatively known

Reference	Class	Multiplicity	Description
synonymFor	EntityType	1..1	The entity type for which this entity type alias is an alternative name.

### 5.2.5 EntitySpecialisationHierarchy

EntitySpecialisationHierarchy is a metaclass each instance of which represents a means by which instances of an entity type may be classified or specialised.

#### Superclass

ModelElement (defined in MFI Core and model mapping)

#### Aliases

Entity Generalisation Hierarchy, Entity Subtype Hierarchy, Entity Type Classification, Entity Classification

<b>Attribute</b>	<b>DataType</b>	<b>Multiplicity</b>	<b>Description</b>
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.
<b>Reference</b>	<b>Class</b>	<b>Multiplicity</b>	<b>Description</b>
further DescriptionOf	EntityType	1..1	The entity type whose instances are classified or specialised by this entity specialisation hierarchy.
comprisedOf	EntitySubtype	1..*	The set of entity subtypes that comprise this entity specialisation hierarchy.
categorisedBy	Attribute	0..1	The attribute that is the category discriminator for this entity specialisation hierarchy.
representedBy	EntitySpecialisation HierarchyForeignKey Attribute	0..*	The set of foreign key attributes that in sequence represent this entity specialisation hierarchy.

### 5.2.6 EntitySubtype

EntitySubtype is a metaclass each instance of which represents a subset of the instances of an entity type (the supertype) that share common attributes and/or relationships distinct from other subsets.

#### Superclass

ModelElement (defined in MFI Core and model mapping)

#### Aliases

Subtype, Object Subclass, Subclass

<b>Attribute</b>	<b>DataType</b>	<b>Multiplicity</b>	<b>Description</b>
[None]			
<b>Reference</b>	<b>Class</b>	<b>Multiplicity</b>	<b>Description</b>
partOf	EntitySpecialisation Hierarchy	1..1	The entity specialisation hierarchy of which this entity subtype is a part.
useOf	EntityType	1..1	The entity type that is used as this entity subtype.

### 5.2.7 Relationship

Relationship is a metaclass each instance of which represents an association between two or more entity types, or between one entity type and itself.

**Superclass**

ModelElement (defined in MFI Core and model mapping)

**Aliases**

Association

Attribute	DataType	Multiplicity	Description
name	String	0..1	A unique name by which this relationship is known. Some 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 methods recognise this concept.

Reference	Class	Multiplicity	Description
within	InformationModel	1..1	The information model of which includes this relationship.
comprisedOf	RelationshipEnd	2..*	The set of relationship ends that comprise this relationship.

### 5.2.8 RelationshipEnd

RelationshipEnd is a metaclass each instance of which represents the part of the definition of a relationship as seen from a given entity type (the host).

**Superclass**

ModelElement (defined in MFI Core and model mapping)

**Aliases**

Association End

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).
linkPhrase	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 methods recognise this concept.
entityRole	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 methods recognise this concept.
collectionType	String	0..1	A statement as 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 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 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 methods do not recognise this concept.

Reference	Class	Multiplicity	Description
partOf	Relationship	1..1	The relationship of which this relationship end is a part.
within	RelationshipEndGroup	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.
representedBy	RelationshipEnd ForeignKeyAttribute	0..*	The set of foreign key attributes that in sequence represent this relationship end.
usedAs	RelationshipEnd UniqueIdentifier Element	0..*	The set of relationship end unique identifier elements for which this relationship end acts as such a unique identifier element.

### 5.2.9 RelationshipEndGroup

RelationshipEndGroup is a metaclass each instance of which represents 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.

#### Superclass

ModelElement (defined in MFI Core and model mapping)

#### Aliases

Association End Group

Attribute	Data Type	Multiplicity	Description
[None]			
Reference	Class	Multiplicity	Description
viewedFrom	EntityType	1..1	The entity type (the host entity type) that is related to other entity types through this relationship end group.
includes	RelationshipEnd	1..*	The set of relationship ends that form part of this group and are mutually exclusive with each other.

### 5.2.10 Attribute

Attribute is an abstract metaclass each instance of which represents a detail that serves to qualify, identify, classify, quantify or express the state of an instance of an entity type.

#### Superclass

ModelElement (defined in MFI Core and model mapping)

#### Subclasses

KeyAttribute, NonKeyAttribute

Attribute	DataType	Multiplicity	Description
name	String	1..1	A unique name for this attribute. In some 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 methods this will be unique within the information model.
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 methods this is not specified.
significance Statement	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. May also be known as a description.
Reference	Class	Multiplicity	Description
of	EntityType	1..1	The entity type instances of which are qualified, identified, classified, quantified or whose state is otherwise expressed by this attribute.
valuesAssigned From	Domain	0..1	The domain which acts as a constraint on the values taken by this attribute.
category DiscriminatorFor	EntitySpecialisation Hierarchy	0..1	The entity specialisation hierarchy for which this attribute is the category discriminator.

### 5.2.11 NonKeyAttribute

NonKeyAttribute is a metaclass each instance of which represents an attribute whose value is fully independent of all relationships or other attributes.

#### Superclass

Attribute

Attribute	DataType	Multiplicity	Description
[None]			
Reference	Class	Multiplicity	Description
[None]			

### 5.2.12 KeyAttribute

KeyAttribute is an abstract metaclass each instance of which represents 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.

#### Superclass

Attribute

**Subclasses**

NativeKeyAttribute, ForeignKeyAttribute

Attribute	DataType	Multiplicity	Description
[None]			
Reference	Class	Multiplicity	Description
[None]			

**5.2.13 NativeKeyAttribute**

NativeKeyAttribute is a metaclass each instance of which represents a key attribute whose value contributes in some way to the identification of individual instances of the host entity type.

**Superclass**

KeyAttribute

Attribute	DataType	Multiplicity	Description
[None]			
Reference	Class	Multiplicity	Description
usedAs	AttributeUnique IdentifierElement	0..*	The set of attribute unique identifier elements for which this native key attribute acts as such a unique identifier element.

**5.2.14 ForeignKeyAttribute**

ForeignKeyAttribute is an abstract metaclass each instance of which represents a key attribute whose value is a duplication of the value of the key attribute (the referenced attribute) in another entity type.

**Superclass**

KeyAttribute

**Subclasses**

RelationshipEndForeignKeyAttribute, EntitySpecialisationHierarchyForeignKeyAttribute

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
[None]			

**5.2.15 RelationshipEndForeignKeyAttribute**

RelationshipEndForeignKeyAttribute is a metaclass each instance of which represents 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.

**Superclass**

ForeignKeyAttribute

Attribute	Data Type	Multiplicity	Description
[None]			
Reference	Class	Multiplicity	Description
representationOf	RelationshipEnd	1..1	The relationship end which is represented by this foreign key attribute along with others in sequence.

### 5.2.16 EntitySpecialisationHierarchyForeignKeyAttribute

EntitySpecialisationHierarchyForeignKeyAttribute is a metaclass each instance of which represents a foreign key attribute whose referenced attribute is in the entity supertype of the related entity specialisation hierarchy.

#### Superclass

ForeignKeyAttribute

Attribute	Data Type	Multiplicity	Description
[None]			
Reference	Class	Multiplicity	Description
representationOf	EntitySpecialisationHierarchy	1..1	The entity specialisation which has the entity supertype which is represented by this foreign key attribute along with others in sequence.

### 5.2.17 Domain

Domain is an abstract metaclass each instance of which represents a pool of values from which an attribute or attributes must take its or their value. A domain provides a set of business validation rules, format constraints and other properties for one or more attributes.

#### Superclass

ModelElement (defined in MFI Core and model mapping)

#### Subclasses

ContinuousDomain, DiscreteDomain

Attribute	Data Type	Multiplicity	Description
name	String	1..1	A unique name by which this domain is known.
significanceStatement	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. May also be known as a description.
Reference	Class	Multiplicity	Description
constraintOn	Attribute	0..*	The set of attributes whose values are constrained by this domain.

**5.2.18 ContinuousDomain**

ContinuousDomain is a metaclass each instance of which represents a domain whose values are not drawn from an explicit list of valid values.

**Superclass**

Domain

**Aliases**

Non-enumerated Domain

Attribute	Data Type	Multiplicity	Description
validationRule	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
[None]			

**5.2.19 DiscreteDomain**

DiscreteDomain is a metaclass each instance of which represents a domain whose permitted values consist of an explicit list of valid values.

**Superclass**

Domain

**Aliases**

Enumerated Domain

Attribute	Data Type	Multiplicity	Description
[None]			

Reference	Class	Multiplicity	Description
constrainedTo	ValidValue	2..*	The set of valid values that comprise this discrete domain.

**5.2.20 ValidValue**

ValidValue is a metaclass each instance of which represents one of the explicit set of permitted values that comprise a discrete domain.

**Superclass**

ModelElement (defined in MFI Core and model mapping)

**Aliases**

Permitted Value

Attribute	DataType	Multiplicity	Description
literal	String	1..1	The actual permitted value. In a conceptual (or platform independent) model it will probably be the concept, for example, 'Male'. In a logical (or platform specific) model it will probably be the code, for example, 'M' or '0'.
Reference	Class	Multiplicity	Description
for	DiscreteDomain	1..1	The discrete domain of which this valid value is one of the permitted values.

### 5.2.21 UniqueIdentifier

UniqueIdentifier is a metaclass each instance of which represents 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.

#### Superclass

ModelElement (defined in MFI Core and model mapping)

#### Aliases

Key, Unique Key

Attribute	DataType	Multiplicity	Description
[None]			
Reference	Class	Multiplicity	Description
of	EntityType	1..1	The entity type whose instances can be uniquely identified by this unique identifier.
comprisedOf	UniqueIdentifier Element	1..*	The set of unique identifier elements that comprise this unique identifier.

### 5.2.22 UniqueIdentifierElement

UniqueIdentifierElement is an abstract metaclass each instance of which represents a statement that a particular attribute or a particular relationship end is a part of a particular unique identifier.

#### Superclass

ModelElement (defined in MFI Core and model mapping)

#### Subclasses

AttributeUniqueIdentifierElement, RelationshipEndUniqueIdentifierElement

Attribute	DataType	Multiplicity	Description
[None]			
Reference	Class	Multiplicity	Description
partOf	UniqueIdentifier	1..1	The unique identifier of which this unique identifier element is a part.

### 5.2.23 AttributeUniqueIdentifierElement

AttributeUniqueIdentifierElement is a metaclass each instance of which represents a unique identifier element that is a statement that a particular attribute is a part of a particular unique identifier.

#### Superclass

UniqueIdentifierElement

Attribute	DataType	Multiplicity	Description
[None]			
Reference	Class	Multiplicity	Description
roleOf	NativeKeyAttribute	1..1	The native key attribute that acts as an element of the unique identifier of which this attribute unique identifier element is a part.

### 5.2.24 RelationshipEndUniqueIdentifierElement

RelationshipEndUniqueIdentifierElement is a metaclass each instance of which represents a unique identifier element that is a statement that a particular relationship end is a part of a particular unique identifier.

#### Superclass

UniqueIdentifierElement

Attribute	DataType	Multiplicity	Description
[None]			
Reference	Class	Multiplicity	Description
roleOf	RelationshipEnd	1..1	The relationship end that acts as an element of the unique identifier of which this relationship end unique identifier element is a part.



Each information model is a set of definitions for things of significance to an organisation, about which information needs to be held, and the relationships between them. In addition, each information model:

- must have a name, which is a unique name by which this information model is known.
- must be expressed in one and only one information modelling language.
- must be comprised of one or more entity types.
- must be comprised of one or more relationships.

Each entity type is a class of concepts or things of significance about which information may need to be held in support of business operations. In addition, each entity type:

- must have a name, which is a unique name by which this entity type is primarily known.
- may have a significance statement, 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.
- must be part of one and only one information model.
- may be known alternatively by one or more entity type aliases.
- may be further defined with one or more entity specialisation hierarchies.
- may be used as one or more entity subtype.
- may be with instances described by values assigned to each of one or more attributes.
- may be related to others through one or more relationship end groups.
- may be with instances identified by one or more unique identifiers.

Each entity type alias is one of a set of alternative names for an entity type. In addition, each entity type alias:

- must have a name, which is a name by which the associated entity type is alternatively known.
- must be a synonym for one and only one entity type.

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 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.
- must be a further description of one and only one entity type.
- must be comprised of one or more entity subtypes.
- may be categorised by one and only one attribute.

Each entity subtype is a subset of the instances of an entity type (the supertype) that share common attributes and/or relationships distinct from other subsets. In addition, each entity subtype:

- must be part of one and only one entity specialisation hierarchy.
- must be a use of one and only one entity type.

Each relationship is an association between two or more entity types, or between one entity type and itself. In addition, each relationship:

- may have a name, which is a unique name by which this relationship is known; some methods do not provide such 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 methods recognise this concept.
- must be part of one and only one information model.
- must be comprised of two or more relationship ends.

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 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); not all methods recognise this concept.
- 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; not all methods recognise this concept.
- 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 methods do not recognise this concept.
- 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 methods do not recognise this concept.
- 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 methods do not recognise this concept.

- must be mutually exclusive with other relationship ends within one and only one relationship end group.
- must be part of one and only one relationship.
- may be used as one or more relationship end unique identifier elements.
- may be represented by a sequence of one or more foreign key attributes.

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.
- must be defined to include one or more relationship ends.

Each attribute is a detail that serves 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 have a name, which is a unique name for this attribute; in some 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 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).
- 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 methods this is not specified because it is assumed that this the maximum cardinality is '1'.
- may have a significance statement, 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.
- must be defined to be part of the description of one and only one entity type.
- may be the category discriminator for one and only one entity specialisation hierarchy.
- may be assigned with values from one and only one domain.

Each non key attribute is an attribute 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.

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. In addition, each native key attribute may be used as one or more attribute unique identifier elements.

Each foreign key attribute is a key attribute whose value whose value is a duplication of the value of the key attribute (the referenced attribute) in another entity type. 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.

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.

Each domain is a pool of values from which an attribute (or attributes) must take its (or their) 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 continuous domain or a discrete domain, but not both.
- must have a name, which is a unique name by which this domain is known.
- may have a significance statement, 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.
- must be a constraint on one or more attributes.

Each continuous domain is a domain whose values are not drawn from an explicit list of valid values. In addition, each continuous 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.

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

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

- must have a literal, which is the actual permitted value; in a conceptual (or platform independent) model it will probably be the concept, for example, 'Male' whilst in a logical (or platform specific) model it will probably be the code, for example, 'M' or '0'.
- must be for one and only one discrete domain.

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.
- must be comprised of one or more unique identifier elements.

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.

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 native key 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.

## Annex B (informative) Use of information modelling concepts within techniques

Table B.1 shows the applicability of major information modelling concepts within the techniques and notations used to develop this part of ISO/IEC 19763.

**Table B.1 – Use of concepts within techniques**

Concept	Applicability					
	Ellis – Barker	IDEF1X	Information Engineering	Chen	Express-G	UML Class Diagrams
Entity types are named	Yes	Yes	Yes	Yes	Yes	Yes
Entity type have significance statements	Optional	Optional	Optional	Optional	Optional	Optional
Entity specialisation hierarchies allowed	Yes	Yes	Yes	No	Yes	Yes
Only one entity specialisation hierarchy allowed	Yes	No	No	NA	No	No
Entity specialisation hierarchies must be complete	Yes	No	Yes	NA	No	No
Entity subtypes in an entity specialisation hierarchy must be mutually exclusive	Yes	Yes	Yes	NA	Yes	No
Relationships have one name	No	Yes	Yes	Yes	Yes	Optional
Relationships have two names (link phrases), one for each relationship end	Yes	No	No	No	No	No
Relationships are recognised as identifying or non-identifying relationships	Optional	Yes	No	No	No	No
Entity types are given role names to signify their role in a relationship	No	No	No	No	No	Optional
Entity types at a relationship end can be recognised as being in a collection	No	No	No	No	Yes	No
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
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
All relationships are binary relationships	Yes	Yes	Yes	No	Yes	Yes
<i>n</i> -ary relationships are allowed	No	No	No	Yes	No	No
Attributes are named	Yes	Yes	Yes	Yes	Yes	Yes
Attributes have significance statements	Optional	Optional	Optional	Optional	Optional	Optional
Domains are named	Yes	Yes	No	No	Yes (as data type)	Yes (as data type)
Domains have significance statements	Optional	Optional	No	No	No	No
Foreign keys are documented	No	Yes	Optional	No	No	No
Unique identifiers (using native key attributes and foreign key attributes) are documented	No	Yes	Optional	No	No	No
Unique identifiers (using native key attributes and relationships) are documented	Optional	No	No	No	No	No

## Annex C (informative) Examples of information model registration

### C.1 Introduction

These examples are based on the registration of the following information models:

- An example based on a car sales scenario which is drawn in EXPRESS-G notation (see Figure C.1).
- An example based on an employee relationships scenario which is drawn in IDEF1X notation (see Figure C.2).
- An example based on a vehicle servicing scenario which is drawn in Ellis-Barker notation (see Figure C.3).
- An example based on a sales order processing scenario which is drawn as a UML Class Diagram (see Figure C.3).

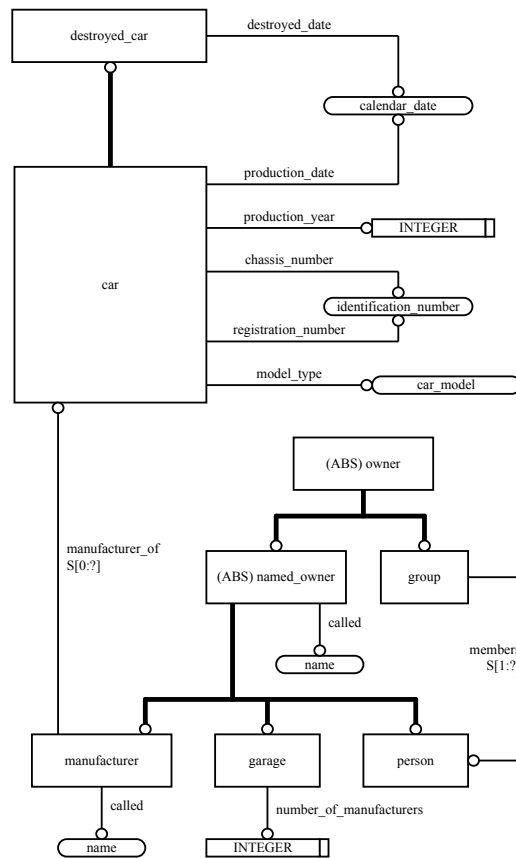


Figure C.1 – Example information model drawn in EXPRESS-G

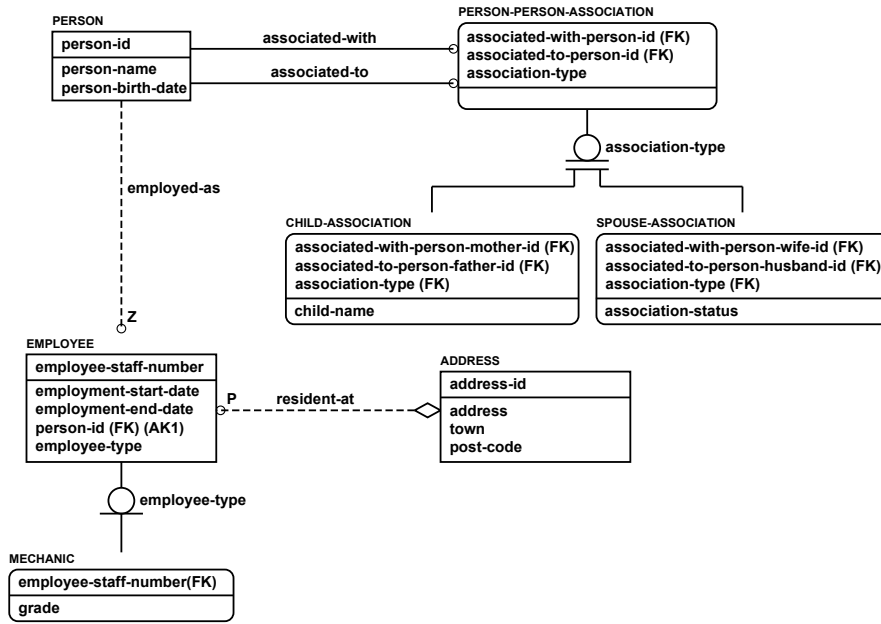


Figure C.2 – Example information model drawn in IDEF1X

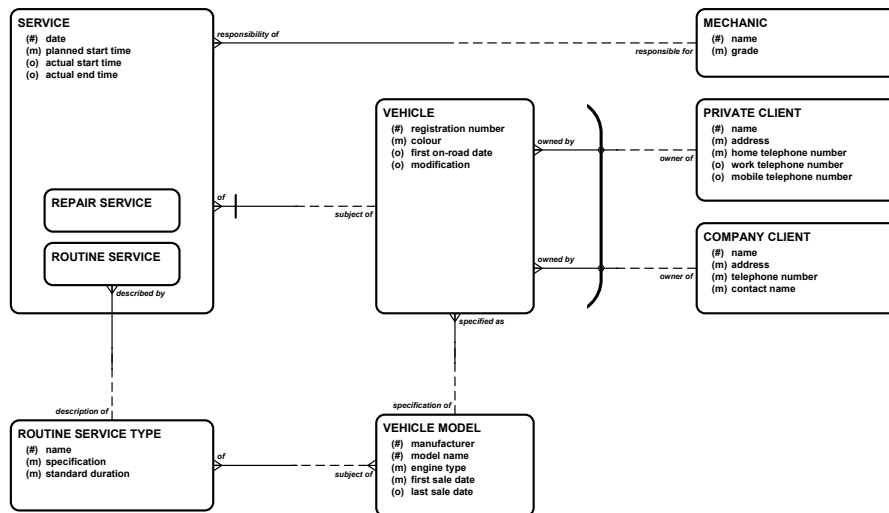


Figure C.3 – Example information model drawn in 'Ellis-Barker' notation

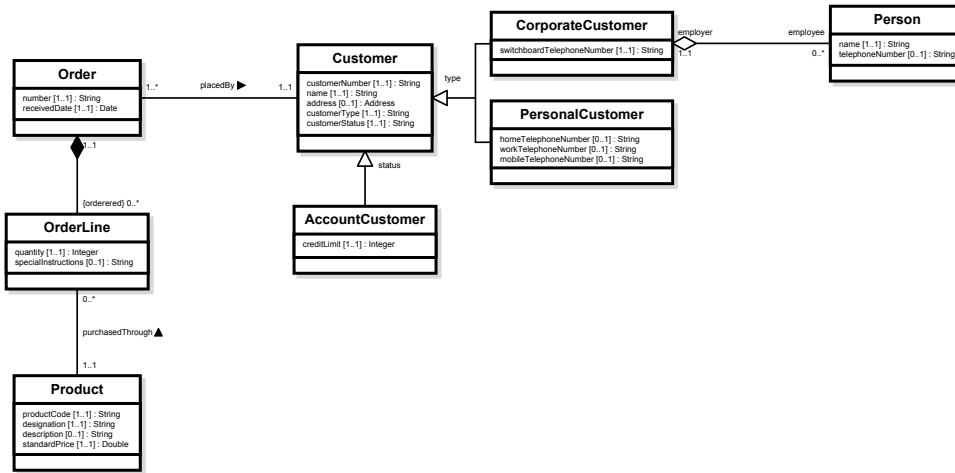


Figure C.4 – Example information model drawn as a UML Class Diagram

## C.2 Example tables

The following tables are populated with information extracted from the models in C.1 and, in some cases, representative information that would be expected in supporting documentation.

The tables represent an implementation in a Relational DBMS. Surrogate keys are used for each table. Supertype-subtype hierarchies are implemented with a single table at the supertype level.

An entry of '-' represents a NULL where an attribute is optional.

An entry of '\*' represents a NULL where an attribute is not appropriate for this subtype.

Table C.1 – Example 'InformationModellingLanguage' table

id	name
im1	EXPRESS-G
im2	IDEF1X
im3	Ellis-Barker
im4	UML Class Diagram

Table C.2 – Example 'InformationModel' table

id	languageFor	name
im1	im1	Car Sales Version 2.7
im2	im2	Employee Relationships V0.4
im3	im3	Robinson Motors Vehicle Service Recording Ver 1.0
im4	im4	Johnson Brothers Ltd sales order processing system (implemented)

Table C.3 – Example 'EntityType' table

id	withinModel	name	significanceStatement
et1	im1	car	--
et2	im1	destroyed_car	--
et3	im1	owner	--
et4	im1	named_owner	--
et5	im1	group	--

id	withinModel	name	significanceStatement
et6	im1	manufacturer	--
et7	im1	garage	--
et8	im1	person	--
et9	im2	PERSON	--
et10	im2	EMPLOYEE	--
et11	im2	MECHANIC	--
et12	im2	ADDRESS	--
et13	im2	PERSON-PERSON-ASSOCIATION	--
et14	im2	CHILD-ASSOCIATION	--
et15	im2	SPOUSE-ASSOCIATION	--
et16	im3	PRIVATE CLIENT	has significance as a client of Robinson Motors who is the personal owner of a vehicle serviced by Robinson Motors
et17	im3	COMPANY CLIENT	has significance as a client of Robinson Motors who is the company owner of a vehicle serviced by Robinson Motors
et18	im3	MECHANIC	has significance as an employee of Robinson Motors who is qualified to repair vehicles
et19	im3	VEHICLE MODEL	has significance as the specification of one or more of the vehicles serviced by Robinson Motors
et20	im3	VEHICLE	has significance as a vehicle serviced by Robinson Motors
et21	im3	ROUTINE SERVICE TYPE	has significance as the specification of a routine service that is specified for a particular vehicle model
et22	im3	SERVICE	has significance as a job of work carried out on a vehicle by Robinson Motors; it may be a routine service or a repair service
et23	im3	REPAIR SERVICE	has significance as a type of service which is carried out as a result of a malfunction of the vehicle or an accident
et24	im3	ROUTINE SERVICE	has significance as a type of service which is carried out as part of the routine maintenance of the vehicle
et25	im4	Customer	--
et26	im4	CorporateCustomer	--
et27	im4	PersonalCustomer	--
et28	im4	AccountCustomer	--
et29	im4	Person	--
et30	im4	Order	--
et31	im4	OrderLine	--
et32	im4	Product	--

Table C.4 – Example 'EntityTypeAlias' table

id	synonymFor	name
eta1	et1	vehicle
eta3	et10	WORKER

Table C.5 – Example 'EntitySpecialisationHierarchy' table

id	furtherDescriptionOf	completenessIndicator	exclusivityIndicator	description	categorisedBy
esh1	et1	False	True	--	--
esh2	et3	True	True	--	--
esh3	et4	True	True	--	--
esh4	et10	False	True	--	a17
esh5	et13	True	True	--	a26
esh6	et22	True	True	Classifies services	--
esh7	et25	True	True	type	a65
esh8	et25	False	True	status	a66

Table C.6 – Example 'EntitySubtype' table

id	partOf	useOf
es1	esh1	et2
es2	esh2	et4
es3	esh2	et5
es4	esh3	et6
es5	esh3	et7
es6	esh3	et8
es7	esh4	et11
es8	esh5	et14

id	partOf	useOf
es9	esh5	et15
es10	esh6	et23
es11	esh6	et24
es12	esh7	et26
es13	esh7	et27
es14	esh8	et28

**Table C.7 – Example 'Relationship' table**

id	withinModel	name	identifyingIndicator
r1	im1	manufacturer_of	--
r2	im1	members	--
r3	im2	associated-with	True
r4	im2	associated-to	True
r5	im2	employed-as	False
r6	im2	resident-at	False
r7	im3	--	False
r8	im3	--	False
r9	im3	--	False
r10	im3	--	False
r11	im3	--	False
r12	im3	--	True
r13	im3	--	False
r14	im4	--	--
r15	im4	placedBy	--
r16	im4	--	--
r17	im4	purchasedThrough	--

**Table C.8 – Example 'RelationshipEnd' table**

id	part Of	within	minimum Cardinality	maximum Cardinality	link Phrase	entity Role	collection Type	aggregation Indicator	composition Indicator
re1	r2	reg1	One	One	--	--	--	--	--
re2	r2	reg2	One	Many	--	--	Set	--	--
re3	r1	reg3	One	One	--	--	--	--	--
re4	r1	reg4	Zero	Many	--	--	Set	--	--
re5	r3	reg5	One	One	--	--	--	--	--
re6	r3	reg6	Zero	Many	--	--	--	--	--
re7	r4	reg7	One	One	--	--	--	--	--
re8	r4	reg8	Zero	Many	--	--	--	--	--
re9	r5	reg9	One	One	--	--	--	--	--
re10	r5	reg10	Zero	One	--	--	--	--	--
re11	r6	reg11	One	Many	--	--	--	--	--
re12	r6	reg12	Zero	One	--	--	--	--	--
re13	r7	reg13	One	One	owner of	--	--	--	--
re14	r7	reg14	Zero	Many	owned by	--	--	--	--
re15	r8	reg14	One	One	owner of	--	--	--	--
re16	r8	reg15	Zero	Many	owned by	--	--	--	--
re17	r9	reg16	Zero	Many	specified as	--	--	--	--
re18	r9	reg17	One	One	specification of	--	--	--	--
re19	r10	reg18	One	Many	subject of	--	--	--	--
re20	r10	reg19	Zero	Many	of	--	--	--	--
re21	r11	reg20	One	One	description of	--	--	--	--
re22	r11	reg21	Zero	Many	described by	--	--	--	--
re23	r12	reg22	Zero	Many	of	--	--	--	--
re24	r12	reg23	One	One	subject of	--	--	--	--
re25	r13	reg24	Zero	Many	responsibility of	--	--	--	--
re26	r13	reg25	One	One	responsible for	--	--	--	--
re27	r14	reg26	Zero	Many	--	employee	--	True	False
re28	r14	reg27	One	One	--	employer	--	False	False
re29	r15	reg28	One	One	--	--	--	False	False
re30	r15	reg29	One	Many	--	--	--	False	True
re31	r16	reg30	One	One	--	--	--	False	False
re32	r16	reg31	Zero	Many	--	--	List	False	False
re33	r17	reg32	Zero	Many	--	--	--	False	False
re34	r17	reg33	One	One	--	--	--	False	False

Table C.9 – Example 'RelationshipEndGroup' table

id	viewedFrom
reg1	et5
reg2	et8
reg3	et6
reg4	et1
reg5	et9
reg6	et13
reg7	et9
reg8	et13
reg9	et9
reg10	et10
reg11	et11
reg12	et12
reg13	et16
reg14	et20
reg15	et17
reg16	et20
reg17	et19
reg18	et19
reg19	et21
reg20	et21
reg21	et24
reg22	et22
reg23	et20
reg24	et24
reg25	et18
reg26	et29
reg27	et26
reg28	et25
reg29	et30
reg30	et30
reg31	et31
reg32	et31
reg33	et32

Table C.10 – Example 'Attribute' table

id	of	name	minimum Cardinality	maximum Cardinality	significance Statement	assigned From	type	representation OfRelationship End	representation OfEntitySpecialisationHierarchy	rank
a1	et1	production_date	--	--	--	d2	non key	*	*	*
a2	et1	production_year	--	--	--	d1	non key	*	*	*
a3	et1	chassis_number	--	--	--	d3	non key	*	*	*
a4	et1	registration_number	--	--	--	d3	non key	*	*	*
a5	et1	model_type	--	--	--	d4	non key	*	*	*
a6	et2	destroyed_date	--	--	--	d2	non key	*	*	*
a7	et4	called	--	--	--	d5	non key	*	*	*
a8	et6	called	--	--	--	d5	non key	*	*	*
a9	et7	number_of_manufacturers	--	--	--	d1	non key	*	*	*
a10	et9	person-id	One	One	--	--	native key	*	*	*
a11	et9	person-name	One	One	--	--	non key	*	*	*
a12	et9	person-birth-date	Zero	One	--	--	non key	*	*	*
a13	et10	employee-staff-number	One	One	--	--	native key	*	*	*
a14	et10	employment-start-date	One	One	--	--	non key	*	*	*
a15	et10	employment-end-date	Zero	One	--	--	non key	*	*	*
a16	et10	person-id	One	One	--	--	foreign key	re10	*	1
a17	et10	employee-type	One	One	--	d8	non key	*	*	*
a18	et11	employee-staff-number	One	One	--	--	foreign key	*	esh4	1
a19	et11	grade	One	One	--	d9	non key	*	*	*
a20	et12	address-id	One	One	--	--	native key	*	*	*
a21	et12	address	One	One	--	--	non key	*	*	*

id	of	name	minimum Cardinality	maximum Cardinality	significance Statement	assigned From	type	representation OfRelationship End	representation OfEntitySpecialisationHierarchy	rank
a22	et12	town	One	One	--	--	non key	*	*	*
a23	et12	post-code	One	One	--	--	non key	*	*	*
a24	et13	associated-with-person-id	One	One	--	--	foreign key	re6	*	1
a25	et13	associated-to-person-id	One	One	--	--	foreign key	re8	*	1
a26	et13	association-type	One	One	--	d6	native key	*	*	*
a27	et14	associated-with-person-mother-id	One	One	--	--	foreign key	*	esh5	1
a28	et14	associated-to-person-father-id	One	One	--	--	foreign key	*	esh5	2
a29	et14	association-type	One	One	--	d6	foreign key	*	esh5	3
a30	et14	child-name	One	One	--	--	non key	*	*	*
a31	et15	associated-with-person-wife-id	One	One	--	--	foreign key	*	esh5	1
a32	et15	associated-to-person-husband-id	One	One	--	--	foreign key	*	esh5	2
a33	et15	association-type	One	One	--	d6	foreign key	*	esh5	3
a34	et15	association-status	One	One	--	d7	non key	*	*	*
a35	et16	name	One	One	has sig ...	d10	native key	*	*	*
a36	et16	address	One	One	has sig ...	d11	non key	*	*	*
a37	et16	home telephone number	One	One	has sig ...	d12	non key	*	*	*
a38	et16	work telephone number	Zero	One	has sig ...	d12	non key	*	*	*
a39	et16	mobile telephone number	Zero	One	has sig ...	d12	non key	*	*	*
a40	et17	name	One	One	has sig ...	d13	native key	*	*	*
a41	et17	address	One	One	has sig ...	d11	non key	*	*	*
a42	et17	telephone number	One	One	has sig ...	d12	non key	*	*	*
a43	et17	contact name	One	One	has sig ...	d10	non key	*	*	*
a44	et18	name	One	One	has sig ...	d10	native key	*	*	*
a45	et18	grade	One	One	has sig ...	d14	non key	*	*	*
a46	et19	manufacturer	One	One	has sig ...	d13	native key	*	*	*
a47	et19	model name	One	One	has sig ...	d15	native key	*	*	*
a48	et19	engine type	One	One	has sig ...	d16	non key	*	*	*
a49	et19	first sale date	One	One	has sig ...	d17	non key	*	*	*
a50	et19	last sale date	Zero	One	has sig ...	d17	non key	*	*	*
a51	et20	registration number	One	One	has sig ...	d18	native key	*	*	*
a52	et20	colour	One	One	has sig ...	d19	non key	*	*	*
a53	et20	first on-road date	Zero	One	has sig ...	d17	non key	*	*	*
a54	et20	modification	Zero	One	has sig ...	d20	non key	*	*	*
a55	et21	name	One	One	has sig ...	d21	native key	*	*	*
a56	et21	specification	One	One	has sig ...	d22	non key	*	*	*
a57	et21	standard duration	One	One	has sig ...	d23	non key	*	*	*
a58	et22	date	One	One	has sig ...	d17	native key	*	*	*
a59	et22	planned start time	One	One	has sig ...	d24	non key	*	*	*
a60	et22	actual start time	Zero	One	has sig ...	d24	non key	*	*	*
a61	et22	actual end time	Zero	One	has sig ...	d24	non key	*	*	*
a62	et25	customer Number	One	One	--	d27	non key	*	*	*
a63	et25	name	One	One	--	d27	non key	*	*	*
a64	et25	address	Zero	One	--	d31	non key	*	*	*
a65	et25	customerType	One	One	--	d25	non key	*	*	*
a66	et25	customerStatus	One	One	--	d26	non key	*	*	*
a67	et26	switchboardTelephoneNumber	One	One	--	d27	non key	*	*	*
a68	et27	homeTelephoneNumber	Zero	One	--	d27	non key	*	*	*
a69	et27	workTelephoneNumber	Zero	One	--	d27	non key	*	*	*
a70	et27	mobileTelephoneNumber	Zero	One	--	d27	non key	*	*	*
a71	et28	creditLimit	One	One	--	d29	non key	*	*	*
a72	et29	name	One	One	--	d27	non key	*	*	*
a73	et29	telephoneNumber	Zero	One	--	d27	non key	*	*	*
a74	et30	number	One	One	--	d27	non key	*	*	*
a75	et30	receivedDate	One	One	--	d28	non key	*	*	*
a76	et31	quantity	One	One	--	d29	non key	*	*	*
a77	et31	specialInstructions	Zero	One	--	d27	non key	*	*	*
a78	et32	product Code	One	One	--	d27	non key	*	*	*
a79	et32	designation	One	One	--	d27	non key	*	*	*
a80	et32	description	Zero	One	--	d27	non key	*	*	*
a81	et32	price	One	One	--	d30	non key	*	*	*

Table C.11 – Example 'Domain' table

id	name	significanceStatement	type	validationRule
d1	INTEGER	--	continuous	--
d2	calendar_date	--	continuous	--
d3	identification_number	--	continuous	--
d4	car_model	--	continuous	--
d5	name	--	continuous	--
d6	association-type	--	discrete	*
d7	association-status	--	discrete	*
d8	employee-type	--	discrete	*
d9	grade	--	discrete	*
d10	person names	has significance as ...	continuous	--
d11	addresses	has significance as ...	continuous	--
d12	telephone numbers	has significance as ...	continuous	--
d13	company names	has significance as ...	continuous	--
d14	mechanic grades	has significance as ...	discrete	*
d15	model names	has significance as ...	continuous	--
d16	engine types	has significance as ...	discrete	*
d17	calendar dates	has significance as ...	continuous	--
d18	vehicle registration numbers	has significance as ...	continuous	--
d19	vehicle colours	has significance as ...	continuous	--
d20	vehicle modifications	has significance as ...	continuous	--
d21	service names	has significance as ...	continuous	--
d22	service specifications	has significance as ...	continuous	--
d23	intervals	has significance as ...	continuous	--
d24	times	has significance as ...	continuous	--
d25	customerType	--	discrete	*
d26	customerStatus	--	discrete	*
d27	String	--	continuous	--
d28	Date	--	continuous	--
d29	Integer	--	continuous	--
d30	Double	--	continuous	--
d31	Address	--	continuous	--

Table C.12 – Example 'ValidValue' table

id	for	literal
vv1	d6	child-association
vv2	d6	spouse-association
vv3	d7	current-marriage
vv4	d7	current-civil-partnership
vv5	d7	current-cohabitation
vv6	d7	separated
vv7	d7	divorced
vv8	d7	wife-deceased
vv9	d7	husband-deceased
vv10	d8	mechanic
vv11	d8	other-employee
vv12	d9	trainee-mechanic
vv13	d9	trained-mechanic
vv14	d9	lead-mechanic
vv15	d14	Mechanic Class 1
vv16	d14	Mechanic Class 2
vv17	d14	Mechanic Trainee
vv18	d16	Petrol
vv19	d16	Diesel
vv20	d25	Corporate
vv21	d25	Personal
vv22	d26	Account
vv23	d26	Casual

Table C.13 – Example 'UniquelIdentifier' table

id	of
ui1	et9
ui2	et10
ui3	et10
ui4	et11
ui5	et12
ui6	et13
ui7	et14
ui8	et15
ui9	et16
ui10	et17
ui11	et18
ui12	et19
ui13	et20
ui14	et21
ui15	et22

Table C.14 – Example 'Unique Identifier Element' table

id	partOf	type	roleOfAttribute	roleOfRelationshipEnd
uie1	ui1	attribute	a10	*
uie2	ui2	attribute	a13	*
uie3	ui3	attribute	a16	*
uie4	ui4	attribute	a18	*
uie5	ui5	attribute	a20	*
uie6	ui6	attribute	a24	*
uie7	ui6	attribute	a25	*
uie8	ui6	attribute	a26	*
uie9	ui7	attribute	a27	*
uie10	ui7	attribute	a28	*
uie11	ui7	attribute	a29	*
uie13	ui8	attribute	a31	*
uie14	ui8	attribute	a32	*
uie15	ui8	attribute	a33	*
uie16	ui9	attribute	a35	*
uie17	ui10	attribute	a40	*
uie18	ui11	attribute	a44	*
uie19	ui12	attribute	a46	*
uie20	ui12	attribute	a47	*
uie21	ui13	attribute	a51	*
uie22	ui14	attribute	a55	*
uie23	ui15	relationship end	*	re23
uie24	ui15	attribute	a58	*

## Annex D (informative) Bibliography

- Arlow, J. and Neustadt, I. (2005) *UML 2 and the Unified Process: Practical Object-Oriented Analysis and Design, Second Edition*. Addison-Wesley
- Barker, R. (1989) *CASE\*METHOD: Entity Relationship Modelling*. Addison-Wesley
- Bruce, T.A. (1992) *Designing Quality Databases with IDEF1X Information Models*. Dorset House
- Chen, P. P-S. (1976) *The Entity-Relationship Model - Toward a Unified View of Data*. ACM Transactions on Database Systems, Vol 1, No 1, March 1976, pp 9-36
- EPISTLE (European Process Industry STEP Technical Liaison Executive) (2003) *Developing High Quality Data Models* [online],  
<http://www.matthew-west.org.uk/documents/princ03.pdf>
- ISO 10303-11:2004 *Industrial automation systems and integration -- Product data representation and exchange -- Part 11: Description methods: The EXPRESS language reference manual*
- Martin, J. (1990) *Information Engineering, Book II: Planning and Analysis*. Prentice Hall