

**MODELLING GUIDELINES FOR 19763 – REVISION 2**

- 1 This document lists the decisions made (or previous decisions endorsed) during the WG2 Interim Meeting held in Crete 24-28 October 2011 that affect the standards to be used for the metamodels in the various parts of ISO/IEC 19763. It was later amended to reflect decisions made at an ad-hoc meeting of the editors of Parts 1, 3, 5, 7, 8, 9, 10 and 12 of 19763 held in Wuhan 17-18 January 2012 and at the WG2 Meeting held in Berlin 31 May 2012 to 8 June 2012.
- 2 Text always takes precedence over diagrams.
- 3 The normative text of a standard alone should provide sufficient information for the standard to be implemented. It should be possible to remove the diagrams and still implement the standard. However, diagrams provide a useful method of visualising the text. (The editors accept that this is good principle but still believe that a diagram is essential to help with understanding.)
- 4 Name style:
  - 4.1 All natural language names are to have any spaces in the names replaced by underscores. This is sometimes known as 'snake\_case'.
  - 4.2 The names of packages and metaclasses (including enumeration classes used as datatypes) are to have initial capital letters for each word of the name, for example, Model\_Element.
  - 4.3 All other names (for example, attribute names, role names) are to be totally in lower case, for example, message\_type, described\_role.
  - 4.4 Names of abstract classes are to be italicised on diagrams.
- 5 Naming of associations on diagrams:
  - 5.1 Each metaclass participating in an association is to have a role name expressing the role that it plays in the association. This role name may be based on a noun, for example, described\_role, or based on a verb, for example, described\_by. This means that each association is to have two names.
- 6 Indicating multiplicities of associations on diagrams:
  - 6.1 A multiplicity statement is to be shown for each end of an association. This means that each association is to have two multiplicity statements.
  - 6.2 Each multiplicity statement is to include both a minimum and a maximum multiplicity, for example, use '0..\*' instead of '\*' and use '1..1' instead of '1'.

- 7 Indicating compositions and aggregations on diagrams:
  - 7.1 Diamonds are to be used where appropriate to indicate compositions and aggregations.
- 8 Representation of attributes on diagrams:
  - 8.1 Attributes should be shown on diagrams where it is feasible to do so. Where it is not feasible to do so the diagram should be annotated to indicate that attributes are not shown.
  - 8.2 In addition to the name of the attribute the multiplicity of the attribute is to be included. The multiplicity is to be of the form "[n..m]" showing the minimum and maximum multiplicity respectively.
  - 8.3 The datatype of the attribute is not to be shown on the diagram.
- 9 Other diagramming issues:
  - 9.1 Notations that represent implementation factors, such as "+" used to represent public attributes, etc, are not to be used.
- 10 Representing metaclasses in normative text:
  - 10.1 For each metaclass a definition that describes the role or significance of instances of the metaclass is to be provided. A suitable form of words for most situations is "XXX is a metaclass each instance of which represents data about ..." or "XXX is an abstract metaclass each instance of which represents data about ...".
  - 10.2 For each metaclass the name of its immediate supertype is to be provided under the heading "Supertype". If there is no immediate supertype the heading is to be omitted.
  - 10.3 For each metaclass, if appropriate, any alternative names (synonyms or aliases) for the metaclass are to be provided.
  - 10.4 For each metaclass a list of attributes is to be provided (see 11 below).
  - 10.5 For each metaclass a list of references is to be provided (see 12 below).
  - 10.6 Subtypes are not to be listed.
- 11 Representing attributes in normative text:
  - 11.1 For each attribute the name of the attribute is to be provided.
  - 11.2 For each attribute the name of the datatype for values of the attribute is to be provided.

(Note: Datatypes are restricted to "boolean", "integer", "date", "value", "sign", "postal\_address", "string", "natural\_range", "datetime", "text", "notation" and "phone\_number". In addition enumeration classes may be included in the model.)

11.3 For each attribute the multiplicity of the attribute is to be provided. The multiplicity is to be of the form "[n..m]" showing the minimum and maximum multiplicity respectively.

11.4 For each attribute a description that describes the role or significance of values of the attribute is to be provided. A suitable form of words is "A ..." where the maximum multiplicity is "1" or "A set of ..." where the maximum multiplicity is "\*" (or any other number greater than one).

## 12 Representing associations in normative text:

12.1 Associations are to be represented in text by providing a reference in each metaclass participating in the association.

12.2 For each reference the name of the reference is to be provided. This name will normally be the role name that describes the role played by the referenced metaclass with respect to the association.

12.3 For each reference the name of the referenced metaclass is to be provided.

12.4 For each reference the multiplicity of the reference is to be provided. The multiplicity is to be of the form "[n..m]" showing the minimum and maximum multiplicity respectively.

12.5 For each reference a description that describes the role or significance of the instance, or instances, of the referenced metaclass with respect to an instance of this metaclass is to be provided. A suitable form of words is "The ..." where the maximum multiplicity is "1" or "The set of ..." where the maximum multiplicity is "\*" (or any other number greater than one).

12.6 For each reference the name of the reference in the referenced metaclass that provides the inverse definition for this association is to be provided.

12.7 For each reference an indication as to whether this metaclass is responsible for the maintenance of the association, ie the precedence of the metaclass with respect to the association, is to be provided. This is to be "Yes" if the metaclass is responsible for the maintenance of the association or "No" metaclass is not responsible for the maintenance of the association.

## 13 Representing associations in informative text:

13.1 Each association is to be described using two sentences as follows:

"Each Model is to be expressed in one and only one Modelling\_Language."

"Each Modelling\_Language is to be used to describe zero, one or more Models."

## 14 Language and spelling:

14.1 All text, normative and informative, is to be expressed in British English. The following dictionaries are to be used, in order of precedence, to check spelling:

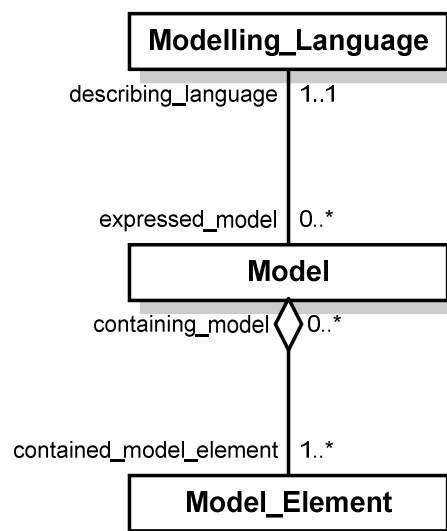
- The Shorter Oxford English Dictionary

- The Concise Oxford Dictionary
- The Collins Concise English Dictionary
- Webster's New World College Dictionary
- Chambers Concise Dictionary

[Note: this means that "modelling" should be used instead of "modeling"]

## 15 Relationship between parts and to MDR-3:

15.1 The metamodel for the core model (Part 2) consists of three metaclasses: Modelling\_Language, Model and Model\_Element. The associations between these metaclasses are shown below:



15.2 Each subordinate part is to include a metaclass that is an immediate subclass of Modelling\_Language.

15.3 Each subordinate part is to include a metaclass that is an immediate subclass of Model.

15.4 All other metaclasses in each subordinate part is to be an immediate subclass of Model\_Element or a subclass of a metaclass that is a subclass of ModelElement.

15.5 The association between the immediate subclass of Modelling\_Language and Model and the associations between the immediate subclasses of Model and Model\_Element in each subordinate part may enforce tighter constraints than those shown in the core model. A subordinate part may not allow looser constraints than those shown in the core model.

15.6 [This paragraph may need to be changed depending on the resolution of the input from the 11179-3 editor] Modelling\_Language, Model and Model\_Element are NOT subclasses of any metaclass specified in MDR-3. Instead the concepts of Clause 5.4 of Edition 3 of MDR-3, which explains the relationship between types of

data/metadata, and instances of these types and their associated values, comes into play.

15.7 [This paragraph may need to be changed depending on the resolution of the input from the 11179-3 editor] All instances of the metaclasses (metadata items) specified in 19763 may be extended by one or more of the types described in Edition 3 of MDR-3, as follows:

- Any metadata item that is to be retrieved directly (as opposed to indirectly through a related item), shall be an Identified\_Item (defined in clause 7.2.2.1 of Edition 3 of MDR-3), so the item can be referenced.
- Any metadata item that is to be designated (named) and/or defined shall be a Designatable\_Item (defined in clause 7.3.2.2 of Edition 3 of MDR-3).
- Any metadata item that is to be registered in a registry shall be a subclass of a Registered\_Item (defined in clause 8.1.2.1 of Edition 3 of MDR-3). Since Registered\_Item is an abstract class, each item must be instantiated as one of the two mutually exclusive and collectively exhaustive subclasses: Administered\_Item (defined in clause 8.1.2.2 of Edition 3 of MDR-3), or Attached\_Item (defined in clause 8.1.2.3 of Edition 3 of MDR-3).
- Any metadata item that is to be classified in a classification scheme shall be a Classifiable\_Item (defined in clause 9.2.2.1 of Edition 3 of MDR-3). [It is anticipated that this may only be required in a future edition of 19763-3.]

16 Use of colour:

16.1 All models and diagrams are to be in black and white only. Colour is not to be used.