BS EN 62656-1:2015

BSI Standards Publication

Standardized product ontology register and transfer by spreadsheets

Part 1: Logical structure for data parcels

bsi.

...making excellence a habit.™

National foreword

This British Standard is the UK implementation of EN 62656-1:2015. It is identical to IEC 62656-1:2014.

The UK participation in its preparation was entrusted to Technical Committee GEL/3, Documentation and graphical symbols.

A list of organizations represented on this committee can be obtained on request to its secretary.

This publication does not purport to include all the necessary provisions of a contract. Users are responsible for its correct application.

Published by BSI Standards Limited 2015

ISBN 978 0 580 69579 7

ICS 33.180.01; 33.180.99

Compliance with a British Standard cannot confer immunity from legal obligations.

This British Standard was published under the authority of the Standards Policy and Strategy Committee on 31 January 2015.

Amendments/corrigenda issued since publication

Date	Text affected

EUROPEAN STANDARD

NORME EUROPÉENNE

EUROPÄISCHE NORM

EN 62656-1

January 2015

ICS 33.180.01; 33.180.99

English Version

Standardized product ontology register and transfer by spreadsheets - Part 1: Logical structure for data parcels (IEC 62656-1:2014)

Enregistrement d'ontologie de produits normalisés et transfert par tableurs - Partie 1: Structure logique pour les paquets de données
(IECI 62656-1:2014)

Standardisierte Übertragung und Registrierung von Ontologien für Produkte mittels Tabellen - Teil 1: Logische Struktur der Datenpakete
(IEC 62656-1:2014)

European Committee for Electrotechnical Standardization
Comité Européen de Normalisation Electrotechnique
Europäisches Komitee für Elektrotechnische Normung

CEN-CENELEC Management Centre: Avenue Marnix 17, B-1000 Brussels

Ref. No. EN 62656-1:2015 E

Foreword

The text of document 3D/226/FDIS, future edition 1 of IEC 62656-1, prepared by IEC/SC 3D "Product properties and classes and their identification", of IEC/TC 3 "Information structures, documentation and graphical symbols", was submitted to the IEC-CENELEC parallel vote and was approved by CENELEC as EN 62656-1.

The following dates were fixed:

• latest date by which the document has to be implemented at national level by publication of an identical national standard or by endorsement	(dop)	2015-07-09
• latest date by which the national standards conflicting with the document have to be withdrawn	(dow)	2017-09-30

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

Endorsement notice

The text of the International Standard IEC 62656-1:2014 was approved by CENELEC as a European Standard without any modification.

In the official version, for Bibliography, the following notes have to be added for the standards indicated:

IEC 61360-4	NOTE	Harmonized as EN 61360-4.
IEC 61968 (Series)	NOTE	Harmonized as EN 61968 (Series).
IEC 61970 (Series)	NOTE	Harmonized as EN 61970 (Series).
ISO 10303 (Series)	NOTE	Harmonized as EN ISO 10303 (Series).
ISO 80000 (Series)	NOTE	Harmonized as EN ISO 80000 (Series).

Annex ZA
(normative)

Normative references to international publications with their corresponding European publications

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 1 When an International Publication has been modified by common modifications, indicated by (mod), the relevant EN/HD applies.

NOTE 2 Up-to-date information on the latest versions of the European Standards listed in this annex is available here: www.cenelec.eu.

Publication	Year	Title	EN/HD	Year
IEC 61360-1	2009	Standard data elements types with associated classification scheme for electric items -- Part 1: Definitions - Principles and methods	EN 61360-1	2010
IEC 61360-2	2012	Standard data element types with associated classification scheme for electric components -- Part 2: EXPRESS dictionary schema	EN 61360-2	2013
ISO 639-1	2002	Codes for the representation of names of languages – Part 1: Alpha-2 code	-	-
ISO 3166-1	2013	Codes for the representation of names of countries and their subdivisions - Part 1: Country codes	-	-
ISO 8601	2004	Data elements and interchange formats - Information interchange - Representation of dates and times	-	-
ISO 10303-11	2004	Industrial automation systems and integration - Product data representation and exchange – Part 11: Description methods: The EXPRESS language reference manual	-	-
ISO 10303-21	2002	Industrial automation systems and integration - Product data representation and exchange – Part 21: Implementation methods: Clear text encoding of the exchange structure	-	-
ISO 13584-24	2003	Industrial automation systems and integration - Parts library – Part 24: Logical resource: Logical model of supplier library	-	-
ISO 13584-25	2004	Industrial automation systems and integration - Parts library - Part 25: Logical resource: Logical model of supplier library with aggregate values and explicit content	-	-
ISO 13584-42	2010	Industrial automation systems and integration -- Parts library -- Part 42: Description methodology: Methodology for structuring parts families	-	-
IEC/TS 62720	2013	Identification of units of measurement for computer-based processing	-	-
ISO/TS 13584-35	2010	Industrial automation systems and integration - Parts library - Part 35: Implementation resources: Spreadsheet interface for parts library	-	-

ISO/TS 29002-5	2009	Industrial automation systems and integration - Exchange of characteristic data – Part 5: Identification scheme	-	-
ISO/IEC 6523-1	1998	Information technology -- Structure for the identification of organizations and organization parts -- Part 1: Identification of organization identification schemes	-	-
ISO/IEC 6523-2	1998	Information technology - Structure for the identification of organizations and organization parts - Part 2: Registration of organization identification schemes	-	-
ISO/IEC 8824-1	2008	Information technology - Abstract Syntax Notation One (ASN.1): Specification of basic notation	-	-
ISO/IEC 11179-3	2013	Information technology - Metadata registries (MDR) - Part 3: Registry metamodel and basic attributes	-	-
ISO/IEC 11179-5	2005	Information technology - Metadata registries (MDR) – Part 5: Naming and identification principles	-	-

CONTENTS

INTRODUCTION

IEC 62656 consists of the following parts, under the general title *Standardized product ontology register and transfer by spreadsheets*:

– Part 1: Logical structure for data parcels;
– Part 2: Application guide for use with IEC CDD;
– Part 3[1]: Interface for common information model.

[1] To be published.

STANDARDIZED PRODUCT ONTOLOGY REGISTER AND TRANSFER BY SPREADSHEETS –

Part 1: Logical structure for data parcels

1 Scope

This part of IEC 62656 specifies the logical structure for a set of spreadsheets, used as “data parcels”, to define, transfer and register product ontologies. Such ontology descriptions in other literatures or disciplines are sometimes called “reference dictionaries”. Thus the logical data structure described in this standard is named “Parcellized Ontology Model” or “POM” for short, and each vehicle of transport of the model is called a “parcel”, and may be used for definition, transfer, and registering of a reference dictionary as a collection of metadata, or for similar purposes for instances belonging to a certain class of the reference dictionary. Moreover, this ontology model allows for modelling or modifying an ontology model per se as a set of instance data, thus it enables an ontology model to evolve over time.

This part of IEC 62656 also includes a standard mapping between the meta-data of dictionary parcels in the spreadsheet format conforming to this standard and the meta-data represented in IEC 61360-2 compliant EXPRESS model for dictionary exchange.

It is assumed that a tool supporting this part of IEC 62656 may read and write a set of spreadsheet data whose semantics and syntax are defined in this part of standard, where the physical file structure of the spreadsheets may be based on the CSV (Comma Separated Values) format, typically used in a commercial spreadsheet application, or any other tabular formats including XML schema compatible or convertible to the CSV format.

The spreadsheet interface structure defined in this part of IEC 62656 contains the following:

– Definition and specification of the logical structure and layout of the spreadsheet interface for definition, transfer, and registering of a reference dictionary;
– Definition and specification of library instance data belonging to a class of a reference dictionary described by a set of spreadsheets conformant to this part of IEC 62656;
– Definition and specification of the meta dictionary that enables the definition and transfer of a reference dictionary as a set of instance data conforming to the meta dictionary;
– Definition and specification of the meta-model as data that enables the definition and transfer of a reference dictionary as a set of instance data conforming to the meta-meta-dictionary
– Specification of the mapping between the dictionary data expressed in the spreadsheet format and the EXPRESS model specified by IEC 61360-2/ISO 13584-42 (with some elements of ISO 13584-25);
– Description of the basic semantic mapping between the dictionary data expressed in the spreadsheet formats defined in this part of IEC 62656 and that of DIN 4002.

The following items are outside the scope of this part of IEC 62656:

– Explanation of the CSV format per se, used in spreadsheet applications;
– Presentation of the data parcels conformant to this part of IEC 62656, such as colouring and sizing of the spreadsheets;
– Specification of the dictionary EXPRESS model conformant to IEC 61360 or ISO 13584 series of standards;
– Normative definition of the mappings between an IEC 61360-ISO 13584 compliant dictionary and another that is based upon a standard other than IEC 61360-ISO 13584;

– Specification of the maintenance procedure of this part of IEC 62656.

This standard is closely related with ISO 13584-35, and developed as a superset or generalisation of the latter. A major difference with the ISO 13584-35 is that this IEC standard enables updates and evolutions in a meta dictionary consisting of meta classes, by which the changes and evolution of an ontology model is realised as an update and modification of the meta dictionary, just by updates and modifications of the instances of the meta-meta dictionary. With this capability, mapping and interfacing with other ontology standards are also facilitated.

2 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.

IEC 61360-1:2009, *Standard data elements types with associated classification scheme for electric items – Part 1: Definitions – Principles and methods*

IEC 61360-2:2012, *Standard data element types with associated classification scheme for electric components – Part 2: EXPRESS dictionary schema*

IEC/TS 62720:2013, *Identification of units of measurement for computer-based processing*

ISO 639-1:2002, *Codes for the representation of names of languages – Part 1: Alpha-2 code*

ISO 3166-1:2013, *Codes for the representation of names of countries and their subdivisions – Part 1: Country codes*

ISO 8601:2004, *Data elements and interchange formats – Information interchange – Representation of dates and times*

ISO 10303-11:2004, *Industrial automation systems and integration – Product data representation and exchange – Part 11: Description methods: The EXPRESS language reference manual*

ISO 10303-21:2002 *Industrial automation systems and integration – Product data representation and exchange – Part 21: Implementation methods: Clear text encoding of the exchange structure*

ISO 13584-24:2003, *Industrial automation systems and integration – Parts library – Part 24: Logical resource: Logical model of supplier library*

ISO 13584-25:2004, *Industrial automation systems and integration – Parts library – Part 25: Logical resource: Logical model of supplier library with aggregate values and explicit content*

ISO 13584-42:2010, *Industrial automation systems and integration – Parts library – Part 42: Description methodology: Methodology for structuring parts families*

ISO/TS 13584-35, 2010, *Industrial automation systems and integration – Parts library – Part 35: Implementation resources: Spreadsheet interface for parts library*

ISO 29002-5:2009, *Industrial automation systems and integration – Exchange of characteristic data – Part 5: Identification scheme*

ISO/IEC 6523-1:1998, *Information technology – Structure for the identification of organizations and organization parts – Part 1: Identification of organization identification schemes*

ISO/IEC 6523-2:1998, *Information technology – Structure for the identification of organizations and organization parts – Part 2: Registration of organization identification schemes*

ISO/IEC 8824-1:2008, *Information technology — Abstract Syntax Notation One (ASN.1): Specification of basic notation*

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

ISO/IEC 11179-5:2005, *Information technology – Metadata registries (MDR) – Part 5: Naming and identification principles*

3 Terms and definitions

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

3.1
applicable data element type
data element type defined for an item class and which applies to all items belonging to that class

[SOURCE: IEC 61360-1:2009, 2.23]

3.2
application
use of software or a standard in an industrial domain

3.3
Common Data Dictionary
CDD
IEC 61360 CDD
data dictionary (available as IEC 61360-4 database) to be shared among all electrotechnical domains, based on the data model specified by IEC 61360-2 and conforming to the semantic requirements specified by IEC 61360-1

Note 1 to entry: This note applies to the French language only.

3.4
common dictionary schema
information model for reference dictionary based on the EXPRESS data model defined in IEC 61360-2 and ISO 13584-42

3.5
composite property
collection of properties that can be referenced from a class or classes as a single entity

[SOURCE: ISO 13584-501:2007, 3.8]

3.6
conjunctive parcels
parcel sheets that are used together to define a library, reference dictionary, or meta-dictionary

3.7
data
quantities, characters, images, or symbols on which operations are performed by computers and other automatic equipment, and which may be stored or transmitted in the form of electric signals, records on magnetic or magneto-optic medium, or other types of recording medium including pieces of paper

3.8
data parcel
parcel
information structure in a form of a level-pair, comprising a set of properties and a set of tuples of values for the set of properties, with an aim to describe a domain data dictionary, a domain data library or an ontological modelling concept

Note 1 to entry: A data parcel is typically implemented and exchanged as a set of spreadsheets, but the medium of implementation or exchange is not limited to spreadsheets; it may be in any other form.

3.9
data type
representation, interpretation, and structure of values used in computer systems and other automatic equipment

3.10
dictionary
data dictionary
set of terms with respective identifiers formulated in a canonical syntax and with commonly accepted definitions designed to yield a lexical or taxonomical framework for knowledge representation in a computer interpretable form, which can be shared by different information systems and communities

3.11
dictionary data
dictionary represented as rows of data conforming to the schema collectively defined by a meta dictionary

3.12
dictionary parcel
set of spreadsheets that are used to define in parts a reference dictionary as the instance of the class that each schema header section of the parcel sheet represents

3.13
dictionary parcel client
dictionary parcel client system
parcel client that can read or write dictionary parcel, and may have an optional capability to send them to, or receive them from a server system

3.14
dictionary parcel server
dictionary server
parcel server that can provide dictionary parcels over Internet

3.15
entity
class of information defined by common properties

[SOURCE: ISO 10303-11:2004, 3.3.6]

3.16
feature
aspect of an item that can be captured by a class structure and set of properties and that cannot exist independently of the item

[SOURCE: ISO 13584-24:2003, 3.41]

3.17
globally unique identifier
identifier that may or may not be based on ISO/IEC 6523 for the global identification of the source of identifier information, and which provides a globally unique identification of a concept without recourse to the linguistic interpretation of the meaning of the letter sequence of the identifier

3.18
instance
unary or n-ary values being identified as a distinct member of a class and characterized by the same set of properties

3.19
international concept identifier
international concept ID
ICID
globally unique identifier that is used in the parcelling format for identifying each data concept

Note 1 to entry: This note applies to the French language only.

3.20
is-a relationship
inheritance relationship defined in the object oriented paradigm

Note 1 to entry: In ISO 13584 the is-a relationship holds between a family of parts and a generic family of parts to which the former family belongs.

[SOURCE: ISO 13584-24:2003, 3.61]

3.21
is-case-of relationship
relationship providing a formal expression of the fact that an object conforms to the partial specification defined by another object

Note 1 to entry: In ISO 13584, all the properties and data types visible or applicable for some family of parts may be imported by all the families of parts that declare to be case-of the former family. These properties and data types may then be used to describe the latter families.

[SOURCE: ISO 13584-24:2003, 3.62]

3.22
is-part-of
aggregation part/whole relationship

Note 1 to entry: In ISO 13584 the is-part-of relationship holds between a family of constituent parts and a family of assembled parts to which the constituent parts belongs.

[SOURCE: ISO 13584-24:2003, 3.63]

3.23
item
thing that can be captured by a structure of class or by a structure of property

3.24
library
set of value instances conforming to a class or a set of classes of a dictionary, and the whole or part of the schema definition that describes the set of instances

3.25
library data supplier
supplier
organization that delivers a library in the standard format defined in ISO 13584 and is responsible for its content

[SOURCE: ISO 13584-1:2001, 3.1.10]

3.26
library external file
file, referenced from a library delivery file, that contributes to the definition of a supplier library

Note 1 to entry: The structure and the format of a library external file is specified in the library delivery file that references it.

[SOURCE: ISO 13584-24:2003, 3.71]

3.27
library integrated information model
LIIM
EXPRESS schema that integrates resource constructs from different EXPRESS schemas for representing supplier libraries for the purpose of exchange and that is associated with conformance requirements

Note 1 to entry: Three library integrated information models are defined in ISO 13584-24 for representing different kinds of supplier libraries.

Note 2 to entry: This note applies to the French language only.

[SOURCE: ISO 13584-24:2003, 3.72]

3.28
library parcel
parcel sheets that are used to define instances of a class whose properties are collated in their header part

3.29
library parcel client
parcel client that can read or write library parcel, and may have an optional capability to send them to, or receive them from a server system

3.30
library parcel server
library parcel server system
parcel server that can provide library parcels over Internet

3.31
meta-class
class representing a category of concepts that is used to instantiate modelling constructs for a domain dictionary, such as (domain) classes, properties, enumerations, or predefined terms

3.32
meta-dictionary
set of meta-classes each of which is characterized by a different set of properties called “meta-properties”, and as an instance of which a reference dictionary may be defined and specified

3.33
meta-property
property that is used to characterize a meta-class and define the syntactic structure of the meta-class

3.34
MOF
Meta Object Facility
metadata management framework, and a set of metadata services for development and interoperability of model and metadata driven systems

Note 1 to entry: This note applies to the French language only.

3.35
Office Open XML
set of XML vocabularies standardized as ISO/IEC 29500, being based on W3C XML Schema, for representing word-processing documents, spreadsheets and presentations

Note 1 to entry: The abbreviation "XML" is derived from the full term "eXtended Markup Language".

3.36
ontological entity
artefact that is used to represent a category of being of things or relationship among them

3.37
parcel client
client system or application that can read or write parcelling sheets in general, and may have an optional capability to send them to or receive them from a server system

3.38
parcel editor
software that edits data parcels, which may have a capability to send or receive the parcels over Internet

3.39
parcelling
act of defining, exchanging, or transmitting information using parcelling sheets defined in this standard

3.40
parcelling tool
tool that can process parcel spreadsheets in general, including PCL-clients, PCL-editors, and PCL-servers

3.41
parcel server
server system or application that can provide parcel spreadsheets in general over Internet

3.42
parcel sheet
parcelling sheet
normalised use and implementation of a data parcel as a spreadsheet

3.43
part
material or functional element that is intended to constitute a component of different products

[SOURCE: ISO 13584-1:2001, 3.1.16]

3.44
PLIB
dictionary data model defined by ISO 13584-42 which is common with IEC 61360-2

Note 1 to entry: PLIB is an acronym of the ISO 13584 series of standards.

3.45
property
set of characteristic information that conceptually characterizes a class and the value of which may be actually supplied by a library supplier and used to characterize instances (parts) that belong to the class or its subclasses

3.46
property of parts library
PLIB-property
kind of property that is used strictly in the sense of the property defined in IEC 61360-2 and ISO 13584-42 standards

Note 1 to entry: This note applies to the French language only.

3.47
reference dictionary
dictionary that is based on ISO 13584-IEC 61360 data model

4 Parcel use cases and scenarios

4.1 Typical use cases

The spreadsheet interface structure defined in this standard may be used either for exchanging a dictionary (an aggregation of metadata) or a library (of instances) conforming to a dictionary (metadata) of objects of a given representation layer, including the metadata of the objects that can be mapped one-to-one to the entirety or a subset of IEC 61360-2/ISO 13584-42 dictionary model. Moreover this standard may be applied to transport or store part or whole of a meta-model, such as (but not limited to) IEC 61360-ISO 13584 dictionary model as a payload (content) of the format, by employing a higher abstraction layer (meta-meta data) to which the payload must conform. However, the most typical use case of the format defined in this standard is for the transport and register of the dictionary and library, which embodies the classification and specification of products by their properties and values. Thus, for ease of understanding, the following 4.2 to 4.5 are dedicated to the illustration of how the parcel format can be used in industry for the above use cases.

4.2 Spreadsheet representation of dictionary or library

The spreadsheet interface structure defined in this standard may be used either for expressing a dictionary or a library. Two interface formats for dictionary and library are in fact two different and specialized uses of the same format, whose underlying physical file structure, i.e., “spreadsheet” is widely recognized and processible by many software applications. For the ease of identification of the two uses, and for the distinction of the specialized spreadsheets from general purpose ones, one for dictionary representation of parts library shall be called “dictionary parcel format”, and the other for library representation of parts library shall be called “library parcel format”, in the following part of the standard. Among the parcel formats used either for dictionary or for library, several spreadsheets need to be processed together, in order to represent consistently a dictionary or a library. Hence, the word “parcelling” comes from the mere fact that both formats use a certain number of spreadsheets packed together, each of which is called “parcel” in this standard, and each one of the spreadsheets represents a semantically different group of EXPRESS entities from one another, although syntactically very similar in structure. The typical use scenario of the spreadsheet interface for dictionary and library is illustrated in Figure 1.

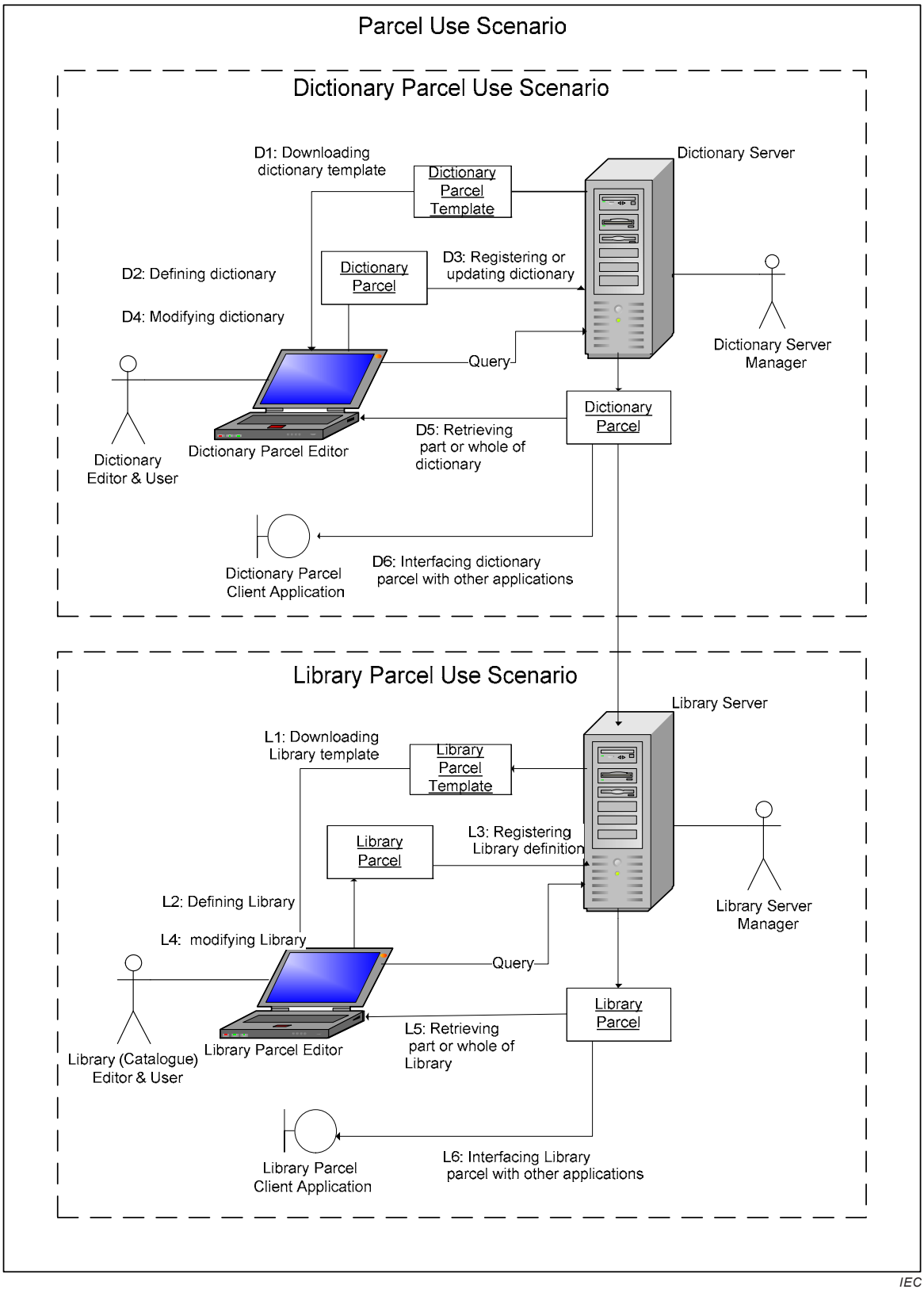


Figure 1 – Parcel use scenario

4.3 Use scenario of dictionary parcel format

The dictionary parcel format may be used in the following use-cases as typical ones:

– D1: Retrieval of a set of blank dictionary parcels from a dictionary server as a dictionary template for defining a new reference dictionary;
– D2: Definition of the content of a reference dictionary;
– D3: Registering of the content of a reference dictionary in a dictionary server;
– D4: Modification of the content of a reference dictionary;
– D5: Retrieving part or whole of a reference dictionary from a dictionary server;
– D6: Interfacing dictionary parcel with other software applications or engineering tools.

Among the above cases, the use cases D1 through D3 are concerned with defining a new reference dictionary with use of spreadsheet software, while the use cases D4 through D6 are for the reuse of information of a reference dictionary which has been registered beforehand. The benefit of putting a reference dictionary into a set of spreadsheets, i.e., a tabular or matrix form is to make it easy for human readers to understand the content, while from an information processing view point, a spreadsheet format is quite close to the data forms suitable for interfacing with relational databases. In addition, the advantage of assigning a global identifier to each of the attributes of dictionary elements is that item-referencing structure of the data model becomes independent from the change of names of the attributes, necessitated by an evolution of standard or merging of standards, and it helps to map between the attributes of two or more different dictionary standards. Furthermore, the parcel format defined in this standard provides a unique functionality called “alternate ID” that enables identification of an item by either identifier, used temporally or locally. Annex L to this part of IEC 62656 may well be used to establish a mapping between this part of IEC 62656 and the DIN 4002, with the help of alternate ID.

4.4 Use scenario of library parcel format

The library parcel format may be used in the following typical use-cases:

– L1: Retrieval of a set of blank library parcels from a library server as a library template for defining a new library;
– L2: Definition of the content of a library;
– L3: Registering of the content of a library in a library server;
– L4: Modification of the content of a library;
– L5: Retrieving part or whole of a library from a library server;
– L6: Interfacing library parcel with other software applications or engineering tools.

Among the above use-cases, as explained in L6 has the purposes of interfaceing with other software applications such as DTP (Desk-Top Publishing) applications, PDM (Product Data Management) applications or ERP (Enterprise Resource Planning) systems or other engineering applications that support spreadsheet formats for data input and output, which are especially important. In fact, many commercial DTP, PDM and analytic engineering tools have a built-in spreadsheet input/output interface, and once the data are read into a commercial spreadsheet application available on the market, conversion of data between one format (or use) and another on the spreadsheet is quite easy and can be almost intuitively done by a domain engineer, who has no special knowledge or prior training about information processing. This demarcates the parcel format from other conventional exchange formats of parts library. In other words, the parcel format may be used as an interface between a tool or application that reads and writes data in a spreadsheet format and another tool or application that processes conventional ISO 13584 dictionary or library data in ISO 10303-21 step physical file format. In this concern, it must be noted that any column which is not headed by a property ID of applicable property which is supposed to be in the class designated by the class ID of the parcel sheet shall be ignored by the system compliant to this standard. This feature is useful for a wide range of applications based on this part of IEC 62656, for information may be embedded in the parcel sheet for other processing, and the same spreadsheet may be used in multiple ways by different applications for their own specific purposes. Thus the parcel formats serve not just for data exchange between ISO 13584 compliant systems, but also as an interface between an ISO 13584 compliant system and

others that are compliant to other standards or protocols but may read and write spreadsheets for data input or output.

4.5 Use scenario of parcel format of higher layers

The parcel format may be used for purposes other than dictionary or library exchange, in particular for defining and exchanging an ontology model as an instance of the model of the immediately upper abstraction layer. The level to be instantiated is called “meta-ontology” abbreviated as “MO”, while the level that gives schema to the MO is called “axiomatic ontology” abbreviated as “AO” in POM. Thanks to this capability, one can extend or modify an ontology model and ultimately it enables schema evolution. For example, adding an extra attribute to a class or to a property is relatively easy in POM. In addition, the method of defining an MO is quite similar to defining a domain dictionary (ontology) with dictionary parcels. Use scenario for defining a meta-ontology may be similar to the dictionary use scenario, explained in a previous section. Meanwhile, the overall structure of a parcel is explained in Clause 5.

5 The Parcellized Ontology Model (POM)

5.1 Overview of the parcel structure

The generic parcel model architecture, or “POM (Parcellized Ontology Model)” for short, has four layers of abstraction, as shown in Figure 2, each of which comprises a level pair. Such a layer is embodied as a set of spreadsheets each sheet of which is a parcel comprising a pair of header and data sections as illustrated in Figure 4. If we explain this in a light of a hierarchical meta-modelling architecture, commonly used in UML (Unified Modelling Language) and MOF (Meta-Object Facility) related literature, a layer consisting of M1 and M0 as header and data section is used for modelling a Domain Library (DL), while a layer consisting of M2 and M1, is used for modelling a dictionary or Domain Ontology (DO). In addition, in order to articulate the modelling structure of the M2, we need a layer corresponding to meta-dictionary or meta-ontology (MO) consisting of the pair M3- M2, as the header section and data section, in a form of data matrix, often embodied as a spreadsheet.

Adding such a superstructure (M3-M2) to the parcel modelling architecture above the dictionary representation layer (M2-M1) makes an implementation of this standard immune to a premature obsolescence, often caused by a partial or minor modification of the data model. The key to achieving this is that updating or modifying a predefined set of meta-meta data as “data” in another layer sufficiently accommodates the gradual evolution of a data model over time. Conversely, constructs of a data modelling standard at M2 being reduced to a set of meta-metadata, can be kept as a set of database records, again in a form of rows and columns of a spreadsheet. As a consequence, another superstructure (M4-M3) at a higher layer may be added just as a logical extension to specify the meta-metadata, in order to explain the constructs of a meta-modelling standard (at M3 layer) in a homogeneous way. In short this layer forms an “axiomatic ontology”. However the Axiomatic Ontology layer (M4-M3) may be kept invariant or constant over time, for its functions are axiomatic and definitions are self-explanatory or axiomatic in nature – not any higher modelling layer is meaningful or significant for our purpose. Thus, the choice of top layer being implemented in a database or being kept in a paper document would not and should not largely influence the effectiveness of this standard. In other words the axiom shall be designed so that it would not unnecessarily restrict the capability of the layers below. Figure 3 illustrates a simplified view of the POM architecture, for ease of understanding.

Meanwhile, regardless of the deepness of the layers of meta-modelling, the underlying structure for accommodating the payload, i.e., the spreadsheet-like structure, and the basic functions to retrieve and manipulate the relevant pieces of data from the payload should remain the same. In all cases, it must be able to represent an ordering among the classes (is_a tree) on a plane of a given layer of abstraction, where the properties of an upper class may be inherited into its subclasses, and a class may be composed of other classes as its parts or components or “blocks” (composition or whole-part hierarchy). In addition, there will be some relational mechanism to relate a member or members of one class to another.

For the above purposes, if a set of dictionary metadata and its instance library data are represented and always stored in a self-similar structure, it is sufficient. The parcel structure being standardized here as a specialised use of spreadsheets is one of possible representation frameworks. Since it is in essence a multi-layer representation of a dictionary and library, it is sufficient to illustrate how a dictionary and its library are used at a particular layer, for example, M1 and M0. Hence, although 5.2 to 5.12.11 are dedicated to the representation of M1 and M0 objects as examples, the basic use scenarios are exactly the same for all other layers.

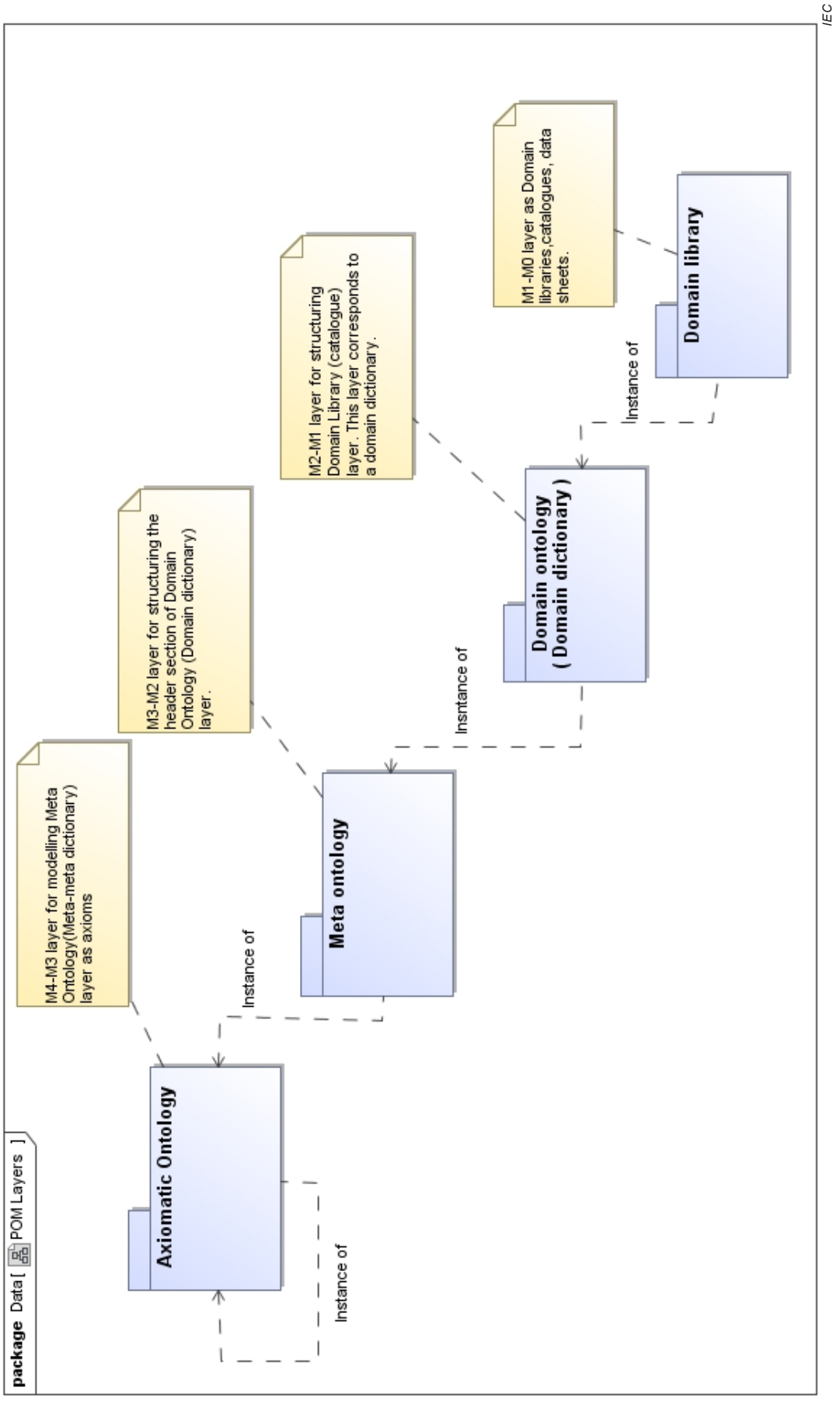


Figure 2 – Parcel architecture as four levels of spreadsheets

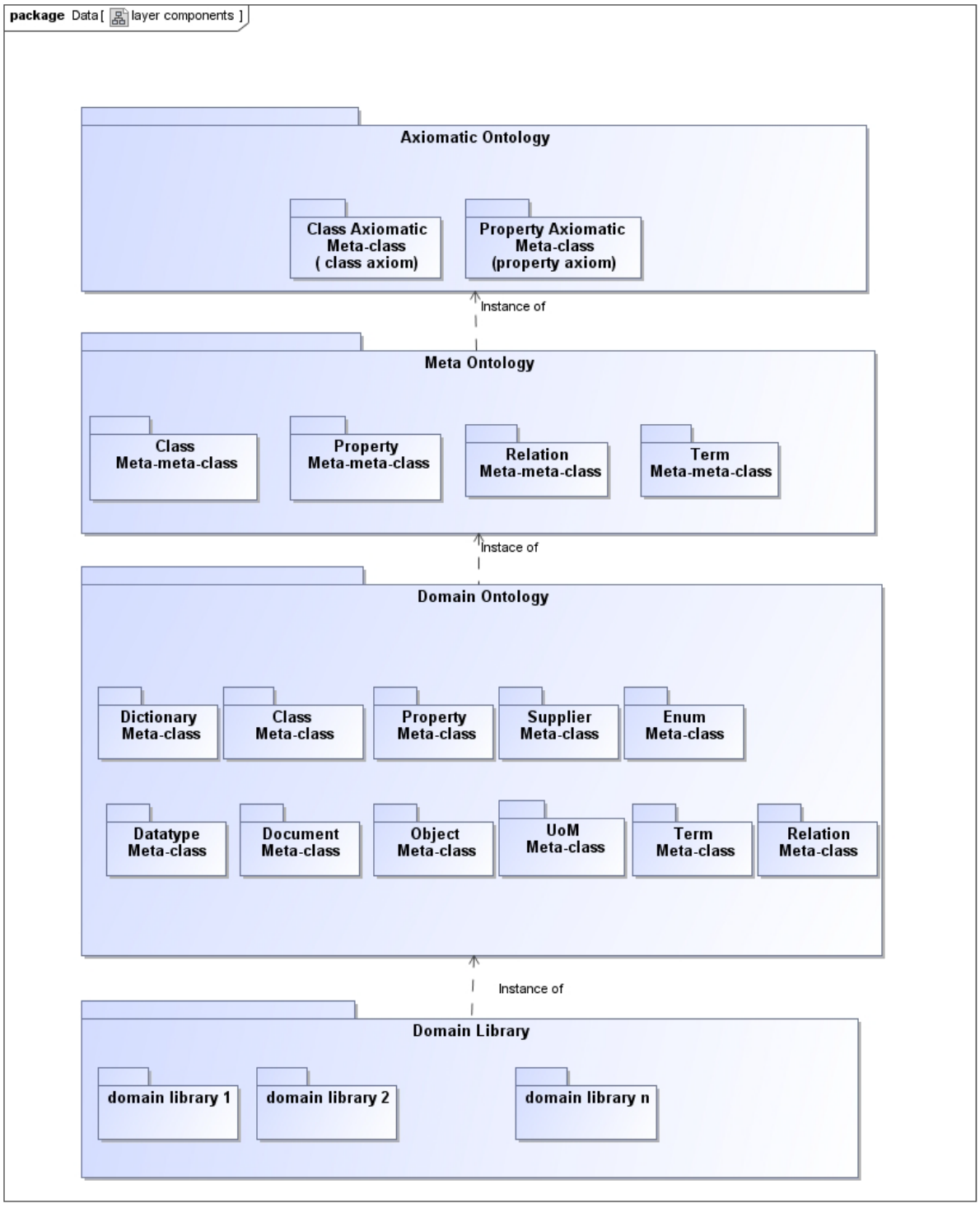


Figure 3 – Components of POM architecture depicted as packages

5.2 Meta dictionary approach

For flexibly modelling either a library or a dictionary on the same base structure as a spreadsheet file, this part of IEC 62656 employs a “meta dictionary approach” with a special emphasis on separating the modelling layer of meta-ontology (meta-dictionary) from that of Domain Ontology (dictionary). In other words, the axiomatic meta-class of M4-M3 as an observer language gives syntax to the meta-ontologies at M3-M2 as a target language. Whereas the meta-ontology as an observer language gives syntax to the Domain Ontology at

M2-M1 as the target dictionary, which in turn describes the semantics of Domain Library, i.e., the real world objects at M1-M0. This layered logical architecture is rooted in Reduction Theorem (See Annex O) in mathematical logic or formal metaphysics, but its application in data modelling is relatively rare. The advantage of taking this approach is that the data model for the reference dictionary represented in this part of IEC 62656 may be flexibly updated or modified without changing the underlying base-structure, modelled and represented as parcels. Since all modelling constructs, such as EXPRESS entities in IEC 61360-2/ISO 13584-42 with a specific name, can be reduced to a set of instances of the meta-ontology layer (depicted in the second column from the left, as MO) defining the meta-classes and meta-properties being identified by a globally unique identifier based on ISO/IEC 6523-1 and ISO/IEC 6523-2, most of the updates and changes in an ontology (dictionary) model may be realized through an addition or a deletion of the instances, or a modification of the values of those instances defined in the meta-meta-classes, i.e. a set of parcels at MO layer.

The meta-dictionary approach explained above is schematically depicted in Figure 4 just for the ease of understanding of the method.

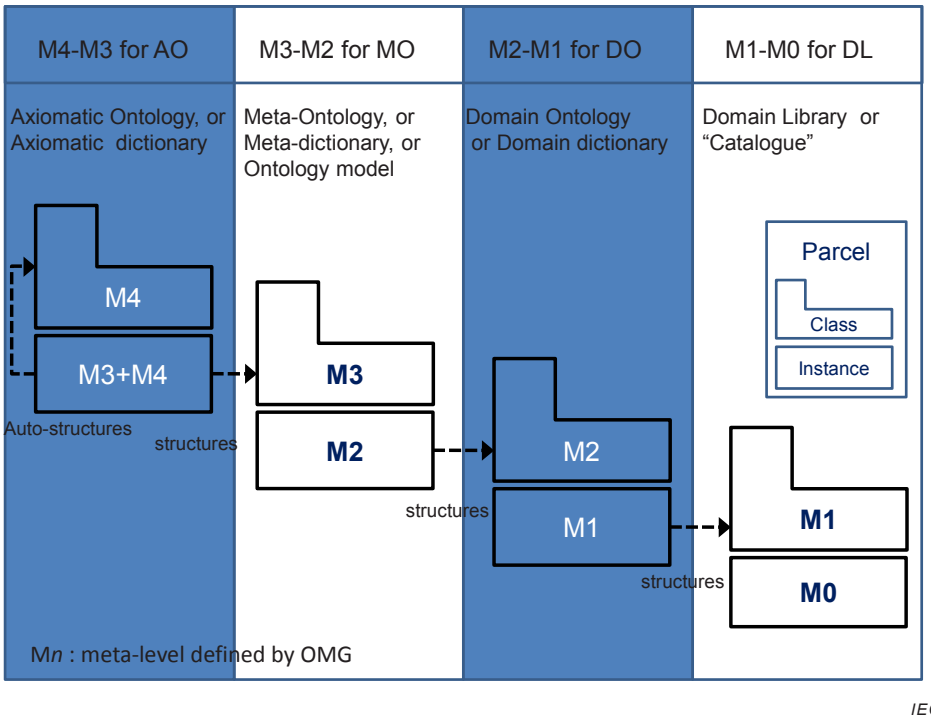


Figure 4 – Schematic diagram of Parcellized Ontology Model (POM)

5.3 Identification structure

This part of IEC 62656 references ISO/IEC 11179 series of standards, in particular ISO/IEC 11179-3 and ISO/IEC 11179-5, as the basic framework for the identification of the types of concepts used to describe elements of data in a parcel sheet. Each data element plays a role similar to "administered item" defined in ISO/IEC 11179-3 as the identifier of a registered item.

Hence, the parcel sheets defined in IEC 62656 may simply be used as a medium for gathering or collecting metadata, or data elements of a reference dictionary, to be registered into a registration authority based on ISO/IEC 11179 series of standards. However, in this standard, such an identifier is also applied and assigned to each construct that structures metadata, and thus its functions need to be further differentiated from that of IRDI (International registration Data Identifier) mechanism for the administered item described in the ISO/IEC 11179-5 standard.

In this standard, we call an element to be identified within a set of metadata or a meta-model as “concept element”. Examples of the concept elements include not only the concepts of class and property, but their constituent key attributes, such as their name, ID, data type, unit of measurement, definition, symbol, note, etc., regardless of their appellations. Conversely, such an identifier of the concept element is referred to as “international concept identifier” or “concept identifier” for short where the context is clear and “ICID” by acronym. However, except for the cases of classifying and identifying the constructs of a metamodel, the function of ICID is identical to IRDI. So it is regarded as an extension of the IRDI.

In parcel format, each ICID is unique across the boundary of organizations, and it has the following sequence:

ICID::= RAI‘#’DI’##’VI

where RAI stands for the registration authority identifier, DI for the data identifier, and VI for the version identifier as described in ISO/IEC 11179-5, and RAI, DI, and VI themselves may be composed of several data elements. For the coding of RAI of an ICID, ISO/IEC 6523-1 and ISO/IEC 6523-2 shall be referenced. Note that in this standard, two consecutive pound signs, i.e., “##” are used to separate DI and VI, while RAI and DI are separated by one single “#” character. This allows us a short hand notation of the concept identifier eliminating either RAI or VI or both where the context for DI is clear.

In addition to the significant sequence as an IRDI, an ICID may entail a notational appendix, starting with “###”, as specified below;

ICID::= RAI‘#’DI’##’VI### NOTE

where the segment starting with “###” is just an appended note as mnemonics for human reader guidance on the referenced ICID and it shall not affect the meaning of the ICID. In other words, this may be removed and added anytime. This is a trivial addition, however, quite useful as a mnemonics to indicate what a property or a class an ICID refers to. Needless to say an ICID with or without a note shall have exactly the same effect on machine interpretation.

DI may be entailed by a language identifier, such as “.en”, “.fr” as if it were an extension of DI. Sometimes, there is a necessity to cleary mark a national language variant in the same language. For this purpose, a 2-letter language code based on ISO 639-1 and a 2-letter country code based on ISO 3166-1 can be combined with a hyphen to form a joint-identifier for the language variant. This may be further extended to form a regional, a communal, or an organizational language variant, as long as the third extension contains distinctly different numbers of letters. However, for practical reasons, the ICID restricts the range of the third extension letters from 4 to 10 alphanumeric characters. As a consequence, for example, “en-US-Hawaii” is an example of a correct extension for ICID. To keep the language variant specification interoperable with the one specified by the Internet Engineering Task Force (IETF), it is recommendable to see the reference [6][2].

Thus, “P501_P000170##001” means an identification of property whose property identification, so called “property_BSU” in ISO 13584 standard, is “P501_P000170” and its version is 1(one) whose information supplier shall be determined by the context, while “0120/1///13584_501_1#P501_P000170” means a property whose information supplier code is “0120/1///13584_501_1” and its property identification is “P501_P000170”, while the version is to be determined by the context. This shorthand convention is extremely useful in a query when a user does not know exactly the current version of a property or class in a reference dictionary maintained in a dictionary server for retrieving missing pieces of information.

———————

[2] Numbers in square brackets refer to the Bibliography.

For the meta-classes and meta-properties defined in this standard, the RAIs and versions also apply. They are assumed to have “0112/2///62656_1” as for RAI, and they are assumed to start from version 1 (one), in the initial release of this standard. If the descriptions of RAIs and versions are missing from the meta- classes and meta-properties, it should be understood that they are supplied as default values as above.

5.4 Typical modelling constructs of POM

5.4.1 Specialization tree versus composition tree

Like other ontology modelling languages, the POM may represent both an “is-a” tree and a “has-a” tree for describing a relation among classes. The is-a tree may be referenced in another literature as “a hierarchy of specialization” or “a classification tree”, while the has-a tree may also be referenced as a “composition tree” or “part-whole relationship”. The so called “block structure” as described in IEC 61987-10 is a special use of the latter structure. In this regard, some philosophers consider the composition as a subject belonging to another branch of science called “mereology” than “ontology”. In particular, a proposition such that an ontology description system should be equipped with a capacity to represent a has-a tree has been often debated when the number of classes being held by another class, i.e., the “cardinality” or “multiplicity” is under focus. Nevertheless the number of elements in a product sometimes plays a pivotal role in describing the nature of the product: Having two wheels is an essential characteristic of a transport vehicle called “bicycle”. Meanwhile, a four-wheel-drive car must have 4 motorized wheels for its raison d’être. In these cases, the structural capability to describe the number of components or parts is indispensable for a practical ontology description. Thus the POM defined in this standard has a capability to describe the part-whole relationship with an additional capacity to denote the cardinality. The formal terms used in IEC 62656 series of standards for an “is-a” type of tree and a “has-a” type of tree are “specialization tree” and “composition tree” respectively. A specialization relationship is achieved by setting appropriately the “super class” attribute of a class to the global identifier of its super-class. On the other hand a composition relationship between an owner class and its subject classes is achieved only by putting as many class reference type properties (maybe in a form of aggregation) in the owner class as the number of the subject classes, where each property is holding the global identifier of exactly one subject class.

5.4.2 Property specialization

There are few ontology modelling frameworks that allow property specialization. For example, the common dictionary model described in IEC 61360-ISO 13584, from which IEC 62656 is evolved, does not have this capability. However this capability is extremely useful in merging, integrating and mapping two or more different domain ontologies. The property specialization is a capacity to derive many specialized properties from one common property prototype. Conversely, many different properties can be related together by finding or adding their common prototype, i.e., a super-type. The POM simply allows this through setting the super-property attribute of each derived property to the ICID of its super-type, i.e., generalized property.

Note that any value of a specialized property may be assignable to its super-property, provided some of the specialized conditions or some details of the value may be lost in the assignment. For a complete understanding of the theoretical basis of the property specialization, some math-logical knowledge is required as a foundation. See Annex O for this purpose.

Given this special functionality, for preventing abuse, the POM stipulates several conditions for applying the property specialization.

A property *P2* is said to be a specialization of another property *P1* if any value that satisfies *P2* also satisfies *P1* although some conditions, attributes (meta properties) or relations of *P2* may be lost in the assignment. If we consider a property as a mapping of element(s) in a domain to an element in the codomain of the mapping (i.e., the “image” of the mapping), where the domain is a subset of a certain collection, or a “class”, in which a property is

applicable, and the codomain is a set of values of the property in which *P1* or *P2* find meaningful values, then it follows in comparison between the 2 mappings representing *P1* and *P2,* at least either one of the domain or codomain of *P2* must be a subset of that of *P1*. Otherwise, in case the mappings themselves are individually identifiable, then some of them are restricted and eliminated for *P2*. In terms of POM, those requirements may be translatable into the following cases:

a) Value range (codomain) of *P2* is a subset of that of *P1;*
b) Domain of *P2* is a subset of that of *P1;*
c) Name scope of *P2* is a subclass of *P1*;
d) *P2* has all the conditions of *P1* and additionally some more;
e) *P2* has all the attributes of *P1* and additionally some more;
f) Referenced classes of *P2* are subclasses of those referenced by *P1*, where *P1* and *P2* are of the class reference type;
g) Referenced instances of *P2* are a subset of the instances that are referenced by *P1* where *P1* and *P2* are of the class reference type;
h) Data type of *P2* is a specialized type of or subject to more constraints, including relational ones, than *P1*;
i) Enumerated list of values of *P2* are all contained in that of *P1;*
j) Cardinality range of *P2* is narrower than that of *P1;*
k) *P2* has a list of fewer alternative units or prefixes than *P1.*

In addition to the above, the following case is also considered as a specialization:

l) the textual definition, including the NOTE, of *P2* semantically means more specialized and narrower extent than that of *P1.*

The relation from a specialized property to its generic type is formed by an attribute named “super property”. This may be represented as a meta-property named “super property” in a property meta-class. Once defined, the ICID(s) of the super property of each property shall be collated in the row following the instruction keyword “#SUPER_PROPERTY”at DO layer.

5.4.3 Divide between specialization and generalization

Each value of a specialized property can be regarded as subdivision of a value of its generalized property. If such a relation can be found between two properties, then one is taken as a specialization of another. As a consequence, in the case of the enumeration type properties, if each of the enumerated values of a property can be mapped as a subdivision of some value of another enumeration property, then the former can be considered as a specialized property of the other. This means if the property *P1* has an enumeration (a, b, c, d), and the property *P2* has an enumeration (a, b, c, d, e), in general, *P2* is a generalization of *P1*. However, if “d” and “e” of *P2* are a subdivision of the value “d” of *P1,* then *P2* can be regarded as a specialization of the property *P2*. Such is a case if “d”s in both *P1* and *P2* are typically labelled as “unknown” or “others” and “e” is a concrete value distinct from (a, b, c), Since some of the previously unknown states for *P1* are now classified into a concrete value “e” and separated from the “d” for *P2*, therefore “d” and “e” of *P2* are subdivisions of the “d” *of P1*. The two cases are illustrated in Figure 5 and Figure 6. Instead of “d” as the value representing unknown, one may also provide a value representing the state, called “not-specified”, “no-option” or “null”, if it is appropriate as the value of the property.

In POM by default, there is no provision for relating each element in one enumeration property to another element in another enumeration property. The identification or differentiation of each element shall be done, properly using the ICID of the element and describing or explaining in text the relationship among the elements in the term meta-class. (However, note that an extension can be easily made by creating a “super-term” field in the term meta-class, if necessary).

5.4.4 Property specialization and cardinality

In general, a subclass has more or equal number of properties than its superclass. In the same manner, a class that has a property with a cardinality (multiplexity) of four must be more specialized than a class with the same property with a cardinality of one. Here, readers should not confuse the number of instances(multiplicity) used in UML and the number of foldings of a property (multiplexity), i.e., the compoundness of the property. Consequently, a property that references a compounded class shall be more specialized than a property that references a less compound or not compounded class. Note that such a property may be represented as an enumeration of class references. In this case, an enumerated class reference type that references a class with a higher cardinality shall be considered a specialization of a property that references its super-class with less cardinality. This can be translated as a case that a product family with more optional choices for its special parts shall be considered a specialization of a type that has no such options.

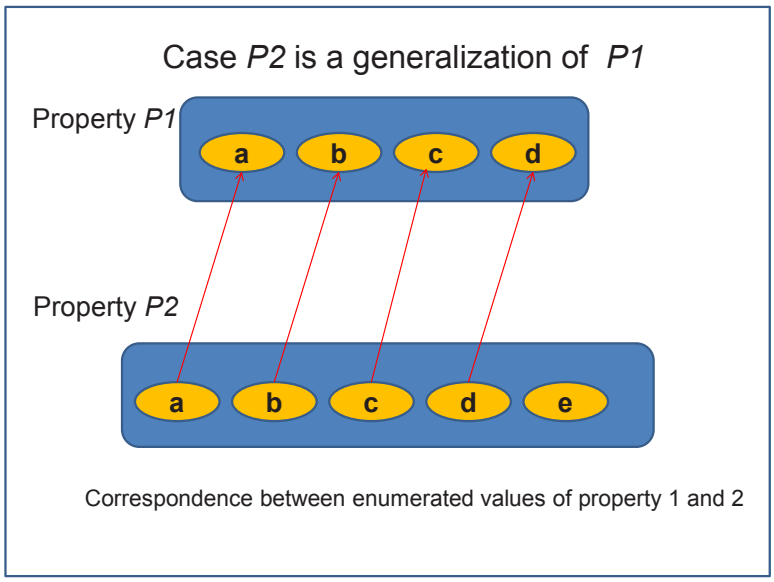


Figure 5 – A generalized enumeration

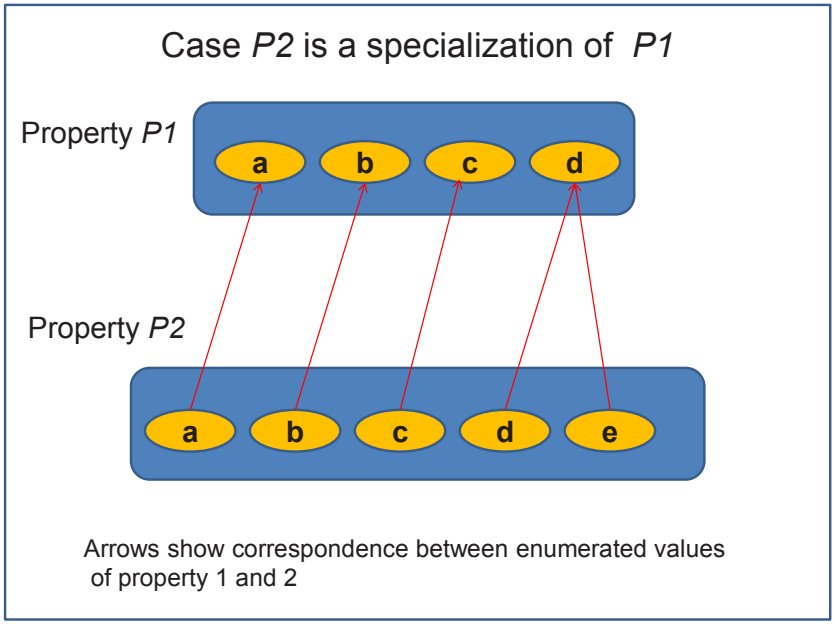


Figure 6 – A specialized enumeration

5.4.5 Property specialization and alternate ID

The property specialization is a structural and a-priori relationship that needs to be introduced when one designs or redesigns a Domain Ontology (dictionary) as part of an intent of the ontology. Subtypes of alternate ID, namely sub-alternate ID and super-alternate ID sometimes play a similar role. However, an alternate ID including a sub-alternate ID or super-alternate ID is an *a posteriori* relationship, and may be introduced or added at any time after the design of domain ontologies is complete and put into use. The application of the latter relationship is especially useful when two or more ontologies are registered into a single repository. Note that alternate IDs are not necessarily set in both two properties which are regarded as mutually alternative. Simply setting an alternate ID in one property that a user has the copyright or permission to change must suffice. In contrast, an exact and permanent structural relationship based on property specialization should be established when one needs to perpetuate the relationship and export the knowledge to other users. In other words, alternate IDs are rather temporary relationships and they do not bind the alternate property related by the alternate ID in a way that would not be updated.

Because the property specialization entails a structural change and it is achieved only when one sets permanently an attribute (meta-property), named “super-property” of a property, to the global identifier of another property as the super-property, this shall entail a version-up of the property to be specialized. Conversely the removal or addition of an alternate ID shall not affect the version of a property unless a permanent structural change of the relationship through the property specialization is introduced simultaneously. In brief, the intent of alternate IDs for property is to address the actual cases that two organizations tend to develop two similar properties in parallel and put them to use without knowing each other beforehand. Thus, such a relationship is not necessarily found by an owner of the ontology, but by a user who need to integrate them.

5.4.6 Mapping classes and properties by alternate ID

When one needs to map properties belonging to different ontologies and/or dictionaries, one can describe the mapping functions and their constraints in a relation meta-class. The details of the structure of the relation meta-class are illustrated in a later clause. However, it is important to note that both the relation among classes and the relation among properties can be represented in a relation meta-class. Thus, a sheet of relation meta-class (i.e., relation

parcel) serves as a mapping specification in many cases. It is also worthwhile to note that the relation meta-class has a capability to invoke an external program for actual control of relations, functional calculation or processing of the defined formulae, or translation from a set of property values into another. The minimum requirement for the specification of each relation is that all properties that are related are explicitly defined and the logical dependency among them is clarified.

5.4.7 Unit with variable prefix

Technology rapidly evolves over time: early designs of a circuit within an LSI (Large Scale Integrated circuit), were in the order of micro metres, however, they are currently in the order of nano-metres in state-of-art products. Likewise, when a USB memory was first put on the market, it was in KBs. However, the capacity of the storage devices has continuoulsly increased and it is currently defined in GBs and it will continue to evolve over time. In this situation, to have a fixed prefix to all the units of measurement is not a realistic solution, in particular for the characterization of the products that evolve rapidly over time in their capacity or of their performance. It is also important to recognize that such a change in magnitude of a physical quantity does not and shall not influence the classification of the products, as long as we base the principle of classification on the characterization by properties. In brief, we know that a USB memory is a USB memory whatever its storage capacity might be.

The unit with variable prefix allows a prefix to be selected from a predefined list of prefixes for the same unit of measurement. The actual prefix combined with the base unit shall be indicated in a cell alongside the value cell.

Note that this functionality does not exist in the original IEC 61360-ISO 13584 series of standards. The only possible recourse is to design all these prefixes combined with the base unit as different alternative units in the PLIB systems, but it is not a common practice, nor the true intent of the alternative unit. Meanwhile, even if such alternatives are not available in a PLIB system, one way translation from POM to PLIB is possible when an appropriate fitting of the magnitude of a value is done before an export, whereas from a PLIB property to a POM property being based on the same unit of measurement, it is only necessary to appropriately select a prefix from the list of prefixes of the unit.

5.4.8 Dependent condition

Dependent condition is simply a subtype of condition type property (condition_ DET) that relies on other condition properties. The current PLIB model assumes that a dependent property (dependent _P_DET) may sometimes rely on a set of properties called conditions. However each of these properties cannot rely on any other condition properties in the standard. The dependent condition is designed only to denote a recursive dependency among properties. Two different dependent condition properties may rely on the same condition(s), but the dependency shall not be circular. In other words, the property dependency graph shall form an acyclic graph.

Using a dependent condition does not automatically define any mathematical or physical formula that governs the dependency among the properties. If such an exact modelling of relationship needs to be specified, it shall be done in a relation meta-class in parallel.

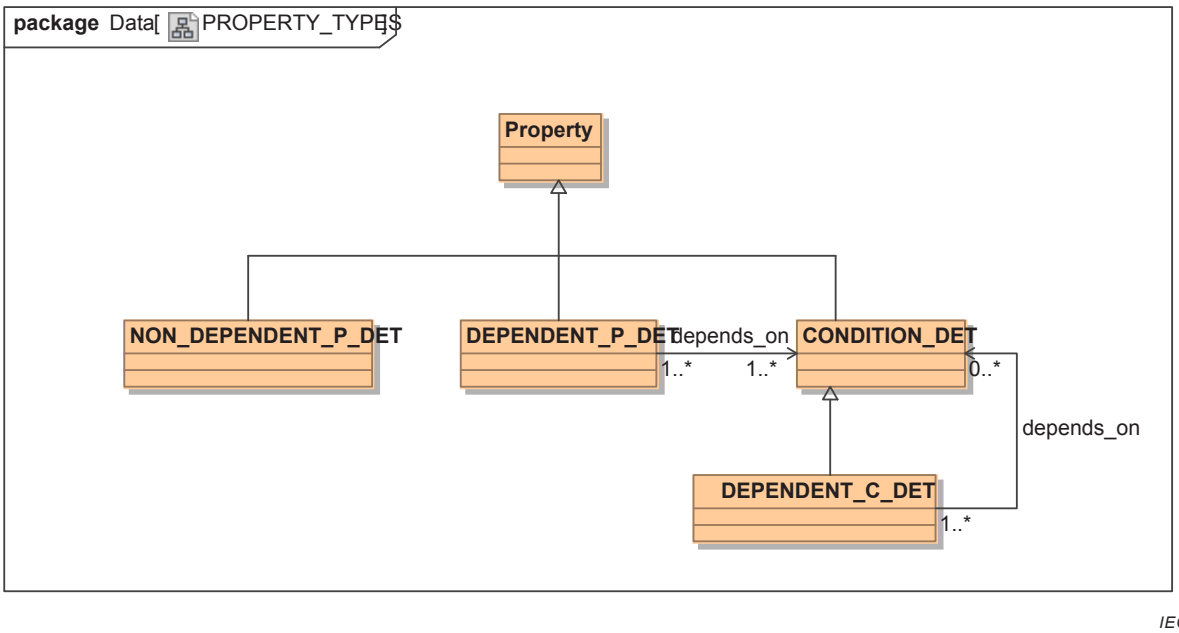


Figure 7 – Dependent property, condition, and dependent condition

Exactly like dependent properties, a dependent condition needs to be declared as “DEPENDENT_C_DET” in a meta property named “Property data element type” (MDC_P020) and its conditions to be depended on shall be set by their respective ICIDs in the meta property named “condition” (MDC_P028).

5.4.9 Use of dependent condition for time dependent property

Most of the products are described by properties which have static values, while some products are especially characterized by properties with time dependent values or values varying over time. Let the first type of the properties be called “static properties” and the latter be called “dynamic properties”. Value of a dynamic property often bears a time stamp recording the measurement time of the value, provided that the property is not designed as an average of several measurements at different moments of time. A property with such a time stamp can be derived from an existing static property by associating a time stamp as a condition to the property. Since some property already has some conditions, it is inevitable that some of the conditions shall be converted into dependent conditions. For example, power consumption of a particular electric component may be highly dependent on the hour of a day. Thus for describing a typical power consumption of the component, it is important to specify the hour of day as condition. Since the power consumption itself may have several other properties as conditions, such as voltage, current, and phase angle, assigning a time stamp to the power consumption means assigning a time stamp to those properties as conditions. In this case, some of the conditions shall be transformed into dependent conditions that depend on another condition containing a time stamp.

5.4.10 Class valued property

Class valued property is a property whose value applies uniformly to all the instances of a class. However the values are assigned in the same manner as other types of properties; being a class valued property does not automatically mean that the value of the property can be assigned simultaneously as the dictionary containing the property is made. In POM as well as PLIB including IEC 61360-2, pre-assignment of a value for a property at the time of dictionary definition is possible only with the enumeration type and CLASS_REFERENCE_TYPE (or CLASS_INSTANCE_TYPE). Thus, to give a single predefined constant value to a class valued property of a class in a given dictionary, the class valued property shall be specified as an enumeration type property with a single choice in its

enumeration list. Note that the POM can make use of any simple type as an enumeration type property, thus such a specification is relatively straightforward.

Class valued properties are assigned to a class through a meta-property of a class meta-class named “class value assignment” (MDC_P017). Note that class valued properties of a class may be inherited from one of its super classes, exactly like other properties. Thus only when those class valued properties are of enumeration type, the values of them are predefined in dictionary for the class. Conversely, in POM, the value of a class valued property other than that of enumeration type may be overloaded by the value given in a lower class into which the class valued property is inherited (as distinct from PLIB). For a class valued type of an enumeration type, the value list shall be the same in all the classes into which the property is inherited into (this is the same as PLIB). Otherwise, the property specialization shall be explicitly used for this purpose.

5.4.11 Class selector with class valued property and class reference

When a class valued property is combined with an enumeration of class references, i.e., ENUM_REFERENCE (or, ENUM_INSTANCE) type, the property of enumerated class references can serve as a class selector for the classes that are referenced. Since the code (value code) of such a type is for human-eye selection, and its paired meaning (value) can be a separate object, the code may be specified as the STRING-encoded value of the class valued property of a class to be referenced (typically the preferred name of the class), and the meaning could be the ICID of the same class by which the latter actual class selection is done. For example, as a class valued property, an enumeration of current types {AC, DC} are given, in a class called “electric motor”, and its subclasses, i.e., AC motor or DC motor, that inherit the property. In this case, either one of the values shall be chosen, representing AC motor or DC motor. In another class, there is a class reference type property that has a list of value pairs, <current type, ICID>. In brief, selecting a current type means selecting an ICID of one of the motor classes to be referenced. This accomplishes a simple class selection using the value code of the class valued property. Note that selecting the value of the class valued property means the selection of the ID of the class to be referenced.

5.4.12 Metamorphic or polymorphic classes

“Polymorphism” is a term used in the IEC 61987 series of standards to describe a kind of selection mechanism similar to that explained in 5.4.11, but rather within an application context. On the other hand, there are some products called “multi-purpose products” each of which is in fact a union of a set of different product classes. Nevertheless, one purpose must be selected when the product is actually put to use. It looks somewhat similar to All-In-One (AIO) products such as a modern facsimile which may serve simultaneously as a fax-machine, a scanner, and a printer, however slightly different in nature. In the case of multi-purpose products, the product provides only one type of functionality in actual use, while in the latter the product is simultaneously multi-functional, or instantly switches between functions. So at any time the functionality of the product may change or at any moment one of the functionalities can be selected. The former type of product may be expressed in this standard as a “metamorphic” product, while the latter as an AIO product. Note that in both cases, the same physical product serves as different types of products, and such a product concept needs to be modelled as a class because it physically exists. Actually some analogue ICs are marketed as multi-purpose ICs and their functionality must be selected before use, for which pin-connections should be selected. A system product or modular product sometimes has a similar characteristic, although in this case, some modules or components of the product are physically replaced to make the product serve for different purposes.

The metamorphic products or concepts can be modelled easily with ENUM_REFERENCE type, or ENUMERATED_INSTANCE type, as explained in 5.4.11.

5.5 Type system extension for data parcels

5.5.1 Extended data types and updates from IEC 61360-2:2002

In POM, there are some extensions to the data type system inherited from IEC 61360-ISO 13584 dictionary model. Extended data types are the following:

- ICID_STRING and IRDI_STRING as extended from STRING_TYPE;
- ENUM_REAL, ENUM_STRING, ENUM_BOOLEAN, ENUM_REFERENCE and ENUM_INSTANCE extended from NON_QUANTITATIVE_TYPE (ENUM_TYPE);
- Measure types of all number types and currency types can be also enumerated.

Note also that for the compatibility with ISO 13584-42:2010 (to-be IEC 61360-2:2012), following subtypes are added for STRING_TYPE:

- DATE_TYPE, TIME_TYPE, and DATE_TIME_TYPE as subtypes of STRING_TYPE:

For the list of data types available for POM, see the tables in Annex D.

5.5.2 ICID_STRING

The ICID_STRING type is a subtype of STRING_TYPE, with a value that has the pattern of an ICID code. In implementation, it is expected that a data validation may be applied to check whether an object surely exists for the referenced ICID code.

5.5.3 IRDI_STRING

The IRDI_STRING type is a subtype of STRING_TYPE, of which the value has the pattern of an IRDI code including ISO 29002-5 compatible code and ICID code. In implementation, it is expected that a data validation may be applied to check whether an object exists for the referenced IRDI code. For ISO_29002_IRDI type, it is virtually treated as an IRDI_STRING, though its type name is different, and no further check on its internal structure is intended in this standard. Note that the IRDI defined in ISO/IEC 11179-6 does not specify what delimiter shall be used for separation among RAI, DI, and VI.

5.5.4 STRING_TYPE and its extensions

The relationships among the newly introduced subtypes of the STRING_TYPE are illustrated in the following diagram. In Figure 8 the subtypes of ENUM_STRING are omitted.

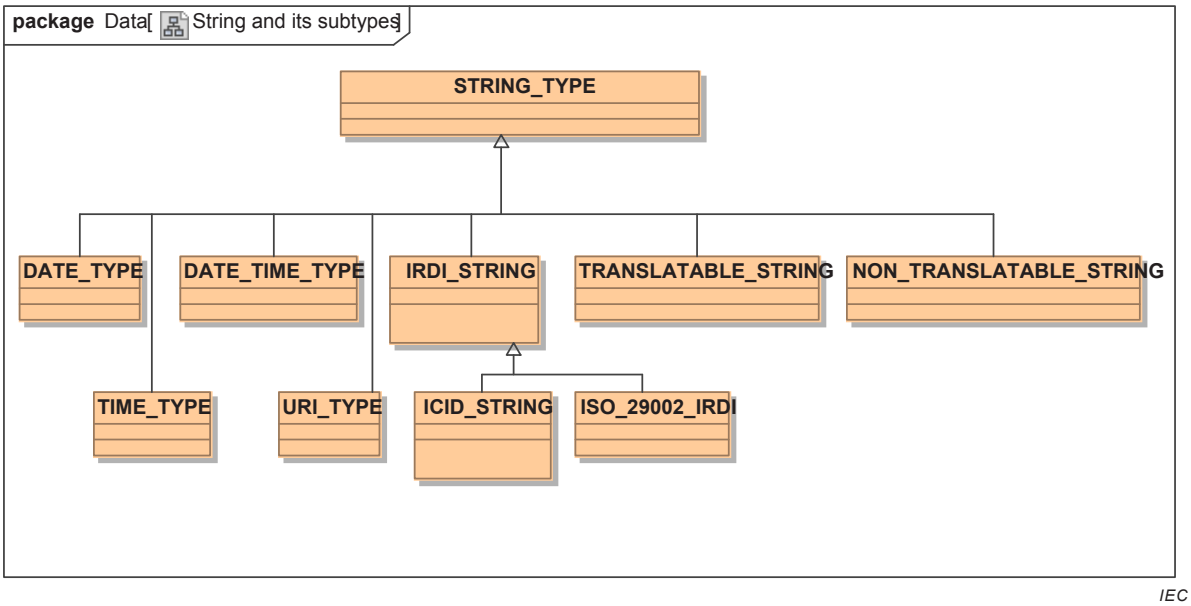


Figure 8 – STRING_TYPE and its extensions

5.5.5 STRING_TYPE and its enumerated simple subtypes

As all the simple types are enumerable in the POM data model, the STRING_TYPE and all its subtypes are also enumerable in the POM data model. In Figure 9, only ENUM_STRING type is shown, however, all the subtypes of the STRING_TYPE may have both the STRING_TYPE and ENUM_TYPE as their parents. Moreover, for number types, all the measure-subtypes and currency-subtypes of what are shown in Figure 9 are also enumerable. For the complete list see Annex D.

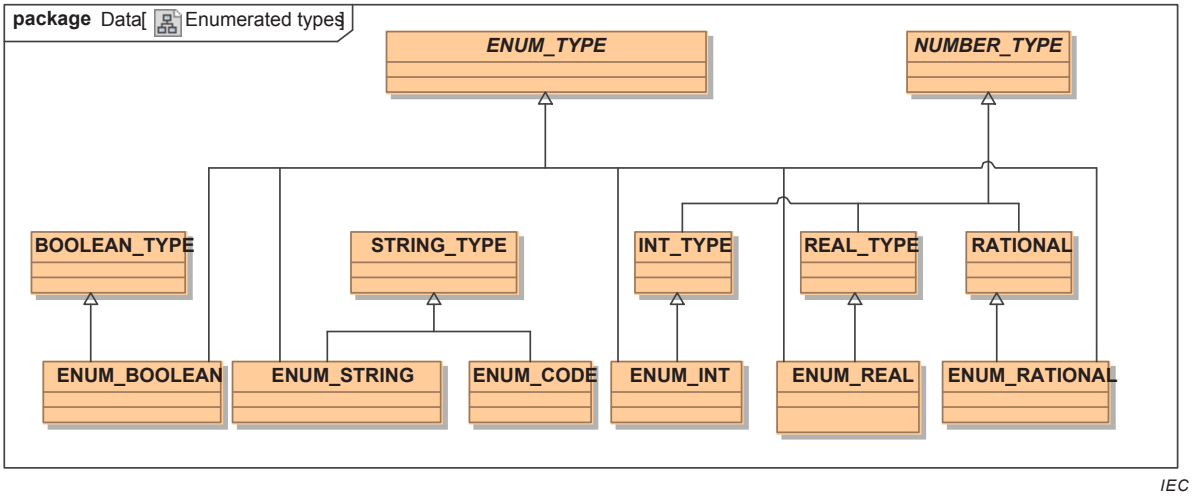


Figure 9 – ENUM_TYPE and its simple subtypes

5.5.6 STRING_TYPE and its enumerated reference subtypes

Among so called “complex types” in PLIB, only CLASS_REFERENCE_TYPE and CLASS_INSTANCE_TYPE are enumerable in POM data model. The reason is simply that the references are of ICID_STRING for both types, although each string signifies a class which may contain several properties. In this sense, ENUM_REFERENCE and ENUM_INSTANCE are no different from ENUM_STRING in their treatment.

Note that in PLIB data model, there is no difference between the two data types, or to be more precise, the CLASS_REFERENCE_TYPE is simply a renaming of the CLASS_INSTANCE_TYPE that has been used in an older edition of ISO 13584-42 or IEC 61360-2, for ease of understanding to end users. However, in the POM data model, they are similar but have a significant difference in the management of their instances:

In the case of CLASS_INSTANCE_TYPE, the instances of the class (i.e., the “part” class) pointed by a property of CLASS_INSTANCE_TYPE are embedded in the class where the property resides, i.e, the “whole” class. Thus, if the whole class where the property of CLASS_INSTANCE_TYPE is destroyed then the part class where the instances are stored shall be destroyed. While in case of CLASS_REFERENCE_TYPE, no such constraint on instances exists. The lives of instances of both the whole and part classes are regarded as independent or not particularly specified. This roughly corresponds to the difference between composition and aggregation relationships in UML model. See Figure 10 for understanding the relationship that complex subtypes of ENUM_TYPE have with their super types.

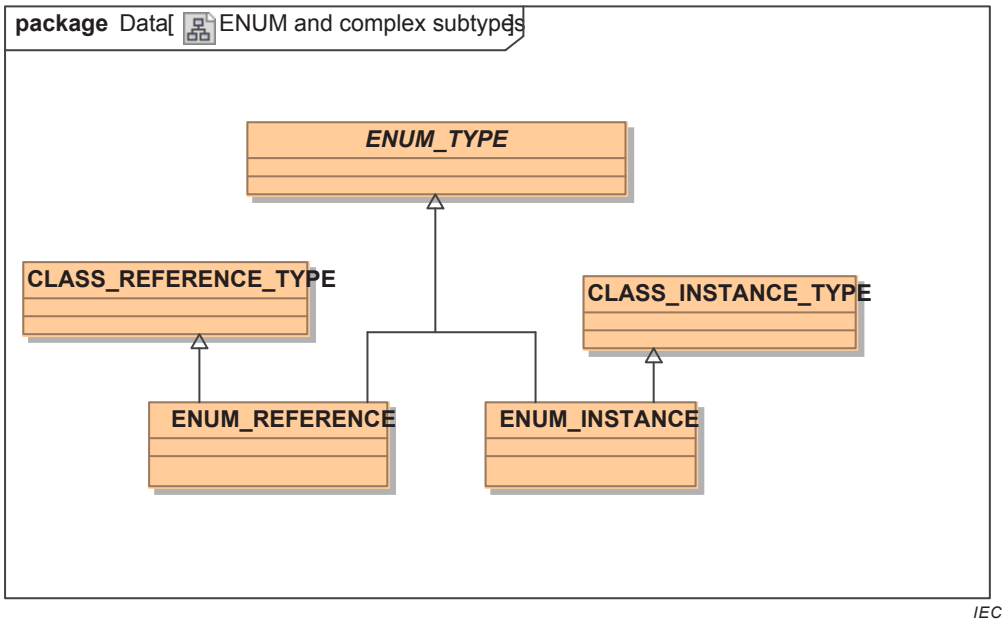


Figure 10 – ENUM_TYPE and its complex subtypes

5.6 Structure of a parcelling sheet

Figure 11 illustrates the general structure of a parcelling sheet which is independent of the type of data content, i.e., dictionary or library, carried by the parcelling sheets. For ease of illustration, both information supplier and version are omitted from the property ID.

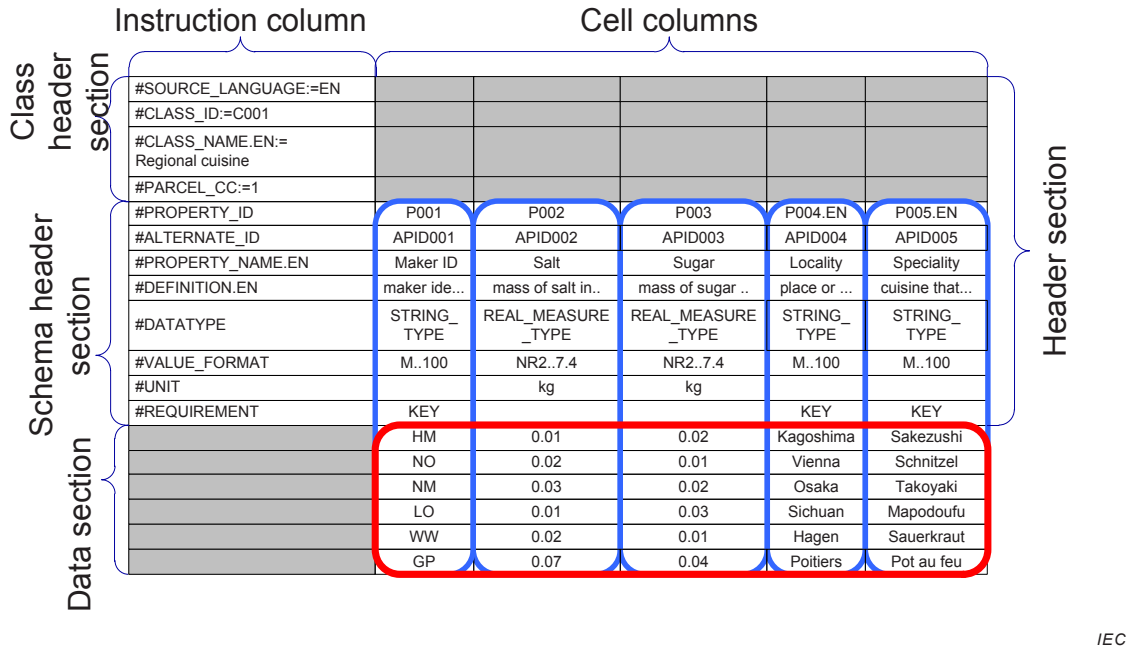


Figure 11 – Structure of a parcelling sheet

As shown in the figure, a parcel sheet may be divided into two horizontal sections; a header section and a data section. The parcel sheet may also be divided vertically into an instruction column, and a set of cell columns. As a further subdivision of the horizontal division, the header section may be divided into a class header section, and a schema header section. The class header section describes the information about this parcelling sheet, thus it may also be called the “class header”, while the schema header section describes the data schema of this spreadsheet as a class, thus it gives the “schema” of the instance data described in the data section of this parcelling sheet.

5.7 File name extension

When it is necessary to differentiate the parcel format conforming to this standard from a spreadsheet for general use, the file name of a file for library parcelling shall be terminated by a file name extension “.pcl”, signifying the parcel format for library. For the specification of dictionary, on the other hand, the extension “.pcd” shall be used.

5.8 CSV representation of parcel format

When representing a parcel sheet in CSV format, data elements in each definition line are separated by a comma “,”, and all the values for a property delimited by a pair of commas shall be arranged in a column. If a data value itself contains comma, then the data value shall be enclosed between a start and an end double-quotation mark, such as “ , “10,000”, ” where 10,000 is a value and the outer-most pair of double quotation marks are just added for marking the quotation into this document.

It is also assumed that each line shall be terminated by a line feed code used in many popular spreadsheet software applications, i.e., “0d0a” in hexadecimal coding which is automatically added by most of the typical spreadsheet applications that are compatible with a leading spreadsheet software commercially available on the market.

Note that a parcel may be exchanged using other formats, such as Office Open XML or some other proprietary formats of software [3] vendors. Hence this standard only defines the spreadsheet view of the parcel data, and lists the CSV representation as a referential example. As long as data may be convertible and it generates the same spreadsheet view, any other format may be used for actual transaction.

5.9 Basic use of parcels

The library use of the spreadsheet structure, or library parcel format for short, shall consist of a set of parcelling sheets, each of which contains in its header section, a list of applicable properties that characterize a class of a given dictionary, and some rows of the instance data in data section that belong to this class. When some of those properties are of enumeration type, i.e., either non_quantitative_string type or non_quantitative_int type, then an additional parcelling sheet called “enumeration parcel” may accompany the parcel sheet as a library of instances, and it may list all the option codes and meanings for the enumeration types used in the library parcel. But this accompaniment of enumeration parcel is not mandatory for the use of library parcel, because although the presence of such an enumeration sheet may help users to enter a precise option value into a property of enumeration type, it is not indispensable as long as the users know the enumerated values. And in more typical cases, the sender and receiver will share the dictionary, prior to the library data exchange.

As noted earlier, in each parcelling sheet, there is a header section and a data section. The header section consists of lines each of which shall start with a pound sign “#” and then entail a reserved word. If the word is not recognized as a reserved word, then the line shall be simply treated as a comment line.

5.10 Header section

5.10.1 Categories of instructions

Each line shall be interpreted using the semantic function described in the cell of the first column. As noted above, instructions to and from an information system conformant to this standard which reads or writes a parcel format file shall be started with a letter “#”. There are four categories of instructions:

- Mandatory;
- Optional – functional;
- Optional – informative;
- Comment.

The words following the symbol “#” and designated in the above categories except the one for comment line shall be treated as reserved words, and shall not be used for other purposes.

5.10.2 Mandatory

Reserved words of this category following the letter “#” are mandatory, therefore, they shall be present in a library exchange file. They are required for any system conformant to this part of IEC 62656 for analysing the property definitions specified by a user. Thus they are also functional in nature.

---

[3] One of such software tools is the MS-EXCEL ® of Microsoft ® Corporation.

The MS-EXCEL ® of Microsoft ® Corporation is an example of a suitable product available commercially. This information is given for the convenience of users of this standard and does not constitute an endorsement by IEC of this product.

5.10.3 Optional - functional

If a reserved keyword of this category following the letter “#” is present in a file, the values of the cells in the line following the second element shall be processed according to the function implied by the keyword. Thus changing the description in the line may change the behaviour of the system which interprets the parcelling sheet. An instruction of this category may or may not exist in an exchange file.

5.10.4 Optional - informative

If a reserved keyword of this category following the letter “#” is present in a file, the values of the cells in the line following the second element are provided to the users of this parcelling sheet just as informative messages by a system conformant to this standard. Since the instruction is informative, changing the information contained in the line would not change the behaviour of the system which interprets the parcelling sheet. An instruction of this category may or may not exist in an exchange file.

5.10.5 Comment

The line started with a letter “#” and not followed by a reserved word shall be interpreted as a comment line. It is used to note a comment within the line. This means an instruction of the categories of mandatory, optional - functional, and optional - informative may be commented out with an additional “#” inserted before the instruction word.

5.10.6 Reserved words

The reserved words of the categories specified in 5.10.1 are summarized in Table C.1 of Annex C.

5.11 Instruction Column

5.11.1 General rule

There are three general description rules that every keyword in the instruction column shall follow:

- Rule 1 – comment out
  - Any line including a line with a keyword may be commented out using “#”.
- Rule 2 – precedence of mandatory instructions
  - Mandatory instructions shall be described before the start of data section used for the description of instances, i.e., library data.
- Rule 3 – precision for the description of keywords
  - Keyword shall be written exactly as specified in this document.

The following 5.11.2 to 5.11.38 define the specific rules and meanings of each keyword.

5.11.2 Class ID

Keyword #CLASS_ID

Name: Class identifier

Definition: globally unique identifier of a class based on ISO/IEC 6523, which is characterized by the properties described in the same parcel, and to which the rows of instance data contained in the data section belong

Description: Character sets used for the class ID shall be in accordance with the class BSU specification defined in IEC 61360/ISO 13584-42 standards, in case

the parcel is used for the exchange of dictionaries conformant to ISO 13584-IEC 61360 common dictionary schema.

Category: mandatory

Format: The keyword “#CLASS_ID” shall be described in the first column and the class code shall be described after the keyword separated by the symbol “:=” (colon-equal). The cells in the second column and after shall be ignored.

Example 1: #CLASS_ID:=0112/1///13584_501_1#P501_C000001##000000001,

Example 2: #CLASS_ID:=0112/1///13584_501_1#P501_C000001,

5.11.3 Preferred name of the class

Keyword: #CLASS_NAME.<lang>

Name: Preferred name of the class

Definition: preferred name of the class specified by the class ID, optionally in the language designated by a language code

Description: The information is optional and informative in a library parcel format because by the specification of a class ID already designates mandatory pieces of information about the class, including the preferred name(s). When an appropriate name is not available for the corresponding language, the cell may be kept open.

Category: optional - informative

Format: For IEC 61360/ISO 13584 conformant classes, the class name shall follow the naming convention that is defined in IEC 61360-1/ISO 13584-42.

Example: #CLASS_NAME.EN:=Environment measuring instrument,

5.11.4 Definition of the class

Keyword: #CLASS_DEFINITION.<lang>

Name: Definition of the class

Definition: statement about the meaning and concept of the class, in the language designated by a language code

Description: The definition available in this field is informative, and supposed to be provided by a parcelling tool which holds the dictionary to which the class belongs. When an appropriate definition is not available in the corresponding language, the cell may be kept open.

Category: optional - informative

Format: Definition shall follow the rules and principles described in Part 2 of the ISO/IEC Directives

Example: #CLASS_DEFINITION.EN:= set of voltage amplifiers of which each amplifier can be described with the same group of data element types,

5.11.5 Note for the class

Keyword: #CLASS_NOTE

Name: Note for the class

Definition: statement that provides additional information about the definition of the class that is essential for the understanding of the definition of the class designated by the class ID described in the same parcel

Description: This may be used for describing information useful for the interpretation of the definition of the class. The information is provided by a system conformant to this standard, and changing the value for this keyword shall not change the behaviour of the system.

Category: optional - informative

Format: A string of alphanumeric characters of any length, except ":=" as a part of the string.

Example: #CLASS_NOTE.EN:= This class corresponds to the class in IEC 61360-4 CDD

5.11.6 Alternate class ID

Keyword: #ALTERNATE_CLASSID

Name: Alternate class ID

Definition: alternate class identifier of the class specified by the class ID noted in the same parcel

Description: This may be used for or for identification within a local system, or for mapping with an external system, another International standard or some other standard that uses a different identification scheme than that is used for the class ID, for historical or domain specific reasons.

Category: optional - informative

Format: A string of alphanumeric characters of any length, excluding ":=" or "," as a constituent.

Example: #ALTERNATE_CLASS_ID:=147/101001///ECALS_10_1#XJA418,

5.11.7 Super alternate class ID

Keyword: #SUPER_ALT_CLASSID

Name: Super-alternate class ID

Definition: alternate class identifier of the class specified by the class ID in the same parcel, where the extent of instances of the class indicated by this alternate ID includes the extent of instances of the class indicated by the class ID

Description: This may be used for identification within a local system, or for mapping with an external system, another international standard or some other standard that uses a different identification scheme than that is used for the class ID, for historical or domain specific reasons.

Category: optional - informative

Format: A string of alphanumeric characters of any length, excluding ":=" or "," as a constituent.

Example: # SUPER_ALT_CLASSID:=147/101001/// ECALS_10_1#XJA418,

5.11.8 Sub-alternate class ID

Keyword: #SUB_ALT_CLASSID

Name: Sub-alternate class ID

Definition: alternate class identifier of the class specified by the class ID in the same parcel, where the extent of instances of the class identified by this alternate ID is included in the extent of instances of the class identified by the class ID

Description: This may be used for identification within a local system, or for mapping with an external system, another International standard or some other standard that uses a different identification scheme than that is used for the class ID, for historical or domain specific reasons.

Category: optional – informative

Format: A string of alphanumeric characters of any length, excluding ":=" or "," as a constituent.

Example: #SUB_ALT_CLASSID:=147/101001/// ECALS_10_1#XJA418,

5.11.9 Source language

Keyword: #SOURCE_LANGUAGE

Name: Source language

Definition: designation of the source language in line with ISO 639, in which the text of dictionary definition was originally made

Category: optional - informative

Format: The keyword “#SOURCE_LANGUAGE” and its assigned value shall be described in the instruction column. The cells in and after the second column shall be ignored. The language code according to ISO 639 enables the identification of the language used as the original.

Mapping: It shall be mapped to the following entity of IEC 61360-2 or ISO 13584-42;
“dictionary_element.administration\administrative_data.source_language”.

Example1: #SOURCE_LANGUAGE:=EN-US,

Example2: #SOURCE_LANGUAGE:=FR,

Example3: #SOURCE_LANGUAGE:=JA,

5.11.10 Parcel mode

Keyword: #PARCEL_MODE

Name: Parcel use mode

Definition: Designation of the use mode of the parcel indicating if the parcel is intended for defining a full content of an ontology or a library or just part of it for updating the content or for test purpose

Description: When FULL is specified it is possible to conduct an integrity check within the conjunctive parcels. If UPDATE is specified, combination of the conjunctive parcels and the target content to be updated forms a complete content and an integrity check should be possible. If PARTIAL is specified, no integrity check is intended for the content.

Category: optional - functional

Format: Either FULL, UPDATE or PARTIAL may be assigned as value

Example: #PARCEL_MODE:=PARTIAL,

5.11.11 Parcel identifier

Keyword: #PARCEL_ID

Name: Parcel identifier

Definition: designation of conjunctive parcels, i.e., the parcels that are used together in the same unit of exchange, and in parts they describe a dictionary or library

Description: Conjunctive parcels are required to have the same alphanumeric letter sequence for this identifier. PARCEL_ID may include neither a comma nor a double quotation mark. When this ID is omitted for dictionary parcelling, the other parcels processed together shall be construed as conjunctive parcels. In case of library parcelling, it may be used to couple library instances with part of the relevant dictionary information, especially with an enumeration parcel, in order for an application to property display the meanings of option codes for non-quantitative (enumeration) type property values.

Category: optional - functional

Format: alphanumeric

Example: #PARCEL_ID:=2006-06-25 08:19:49,

5.11.12 Parcel conformance class identifier

Keyword: #PARCEL_CC

Name: Parcel conformance class identifier

Definition: designation of the conformance class number according to either IEC 62656 or ISO 13584-35 conformance classes, for the data carried in the data section, in the parcel

Description: When there is a local extension, the value for parcel conformance class identifier shall be clearly marked for such an extension.

Category: optional - informative

Format: In case of IEC 62656-1 based conformance classes, identifiers assignable to this meta property shall be selected according to the Table 5. If the conformance class ID = 2 or 2x is selected and the assumed meta-ontology (MO) is a specialized version of IEC 61360-2, a token may be passed in a pair of parentheses placed immediately after the conformance class ID (CCID). In case of ISO 13584-35 based conformance classes an integer value between one (1) and ten (10) shall be used, and the acronym “P35” shall be noted in the parentheses

Example: #PARCEL_CC:=9(P35),

5.11.13 Default supplier

Keyword: #DEFAULT_SUPPLIER

Name: Default supplier

Definition: prefix to be added to the shorthand notations of class ID and property ID in the header section, with an aim to make a full supplier identifier sequence designating the information supplier

Description: This shorthand notation of IDs shall be used only in the header section. This short cut notation is allowed for definition purpose only, and it is assumed that the parcelling file containing the shorthand notation would be pre-processed to yield a full identifier sequence, before it is actually sent to external systems compliant to this part of IEC 62656.

Category: optional - functional

Example: #DEFAULT_SUPPLIER:=0112/1///62656_1_1,

5.11.14 Default version

Keyword: #DEFAULT_VERSION

Name: Default version

Definition: postfix to be appended to the shorthand notations of the class ID and property ID in the header section to denote the version of them, with an aim to make a full identifier sequence

Description: This shorthand notation of IDs shall be used only in the header section. This short cut notation is allowed for definition purpose only, and it is assumed that the parcelling file containing the shorthand notation would be pre-processed to yield a full identifier notation, before it is actually sent to external systems compliant to this part of IEC 62656. The version shall be issued in an ascending order.

Category: optional - functional

Format: Only positive integer number of one to maximum nine digits shall be used. Thus the allowed version is an integer between 1 and 999999999. Leading zeros before the first non-zero digit shall be ignored. Thus both 000001 and 001 are equal to 1.

Example: #DEFAULT_VERSION:= 1,

5.11.15 Default data supplier

Keyword: #DEFAULT_DATA_SUPPLIER

Name: Default data supplier

Definition: prefix to be added to the shorthand notations of ICID in the data section, with an aim to make a full identifier sequence

Description: This prefix shall be specified and applied column by column to the shorthand notations of instances in the data section of a parcel. This shorthand notation is for the convenience of definition and for the ease of visual recognition of the instance data by human users. If a parcel containing this shorthand notation is to be sent to an external system that is not capable of processing the shorthand notation, the data should be pre-processed to yield a full notation, before it is actually sent to

the external system.

The shorthand notation of the data is possible only when the property is of STRING_TYPE, and transformable to an ICID_STRING

Category: optional - functional

Example: #DEFAULT_DATA_SUPPLIER, 0112/2///61987_11_1, , ,

Spreadsheet view: It will be displayed by a spreadsheet application as in Figure 12.

#CLASS_ID:= MDC_C003;				
#CLASS_NAME:= Property meta class;				
#DEFAULT_SUPPLIER:= 0112/2///IEC62656_1_1;				
#PROPERTY_ID	MDC_P001_6	MDC_P002_1	MDC_P004_1	MDC_P022
#PROPERTY_NAME	Code	Version Number	Preferred name	Data type
#DATA_TYPE	ICID_STRING	STRING	STRING	STRING
#DEFAULT_DATA_SUPPLIER	0112/2///61987_11_1			
#DEFAULT_DATA_VERSION	001			
	AAD001	001	die ID	STRING
	AAD002	001	die name	STRING
	AAD003	001	die version	STRING

Figure 12 – Display example of Default data supplier used for IEC 61968-11

5.11.16 Default data version

Keyword: #DEFAULT_DATA_VERSION

Name: Default data version

Definition: postfix to be appended to the shorthand notations of class ID and property ID in the data section to denote the version of the items, with an aim to yield a full identifier sequence

Description: This postfix shall be specified and applied column by column to the shorthand notations of instances in the data section of a parcel. This shorthand notation is for the convenience of definition and for the ease of visual recognition of the instance data by human users. If a parcel containing this shorthand notation is to be sent to an external system that is not capable of processing the shorthand notation, the data should be pre-processed to yield a full notation, before it is actually sent to the external system. The shorthand notation of the data is possible only when the property is of STRING_TYPE, and transformable to an ICID_STRING

Category: optional - functional

Format: Only positive integer numbers of one to maximum nine digits shall be used. Thus the allowed version is an integer between 1 and 999999999.

In case of a number with preceding zeros, such as 001 or 000000010, zeros placed before a positive integer number such as 1 or 10, shall be ignored, when it is converted into a value for the version.

Example: #DEFAULT_DATA_VERSION, ,001

Spreadsheet view: It will be displayed by a spreadsheet application as in Figure 12.

5.11.17 Data object identifier name

Keyword: #OBJECT_ID_NAME

Name: Data object identifier name

Definition: name of the data object identification system, by which each row of instance in the data section shall be uniquely and globally identified

Description: Currently either GUID or UUID may be selected for use.

Category: optional - functional

Example: #OBJECT_ID_NAME:=GUID,

5.11.18 Property ID

Keyword: #PROPERTY_ID

Name: Property identifier

Definition: globally unique identifier of a property based on ISO/IEC 6523 referenced for the definition of instances in the data section of the same parcel

Description: This corresponds to the specification of the property BSU of a known applicable property that is used for library exchange in ISO 13584-25 in accordance with the common dictionary schema defined in ISO 13584-42 and IEC 61360-2 standards.

When a property ID listed in the #PROPERTY_ID line is not recognizable as an applicable property of the class designated by the class ID described in the same parcel, all the information in the cells in the same column designated by the property ID shall be ignored as comments.

Category: mandatory

Format: The keyword “#PROPERTY_ID” is described in the first column. The ICIDs of properties are described in the second and following columns . A

property ID shall be encoded as in the following table.

Table 1 – Description of the property ID code

level	property ID code description
level 1	supplier_bsu.code + sep_id + property_bsu.code + sep_cv + property_bsu.version
level 2	supplier_bsu.code + sep_id + class_bsu.code + sep_cv + class_bsu.verison + sep_id + property_bsu.code + sep_cv + property_bsu.version
level 3	supplier_bsu.code + sep_id + property_bsu.code
level 4	property_bsu.code + sep_cv + property_bsu.version
level 5	property_bsu.code

NOTE 1 If the data type of the property is level_type, a column is allocated for each level defined in the data type. Such property code is described as “property code” + “sep_id” + “levels”.The levels are identified by three-letter alphabets and are described as in the following way: “MIN”, “NOM”, “TYP”, “MAX”.

NOTE 2 If the data type of the property is translatable_string, a column is allocated for each language. Such property code is described as “property code” + “sep_id” + “language code”. The language code identifies a language according to ISO 639-1. Values are e.g. “EN” for English in general, “FR” for French, “RU” for Russian, “DE” for German. If a language variant needs to be specified, a country code based on ISO 3166-1 canbe added to the ISO 639-1 code, as in “en-GB” for UK English, and “en-US” for US English, with a hyphen being inserted between the two codes.

For the use of level 4 and level 5 notations, default supplier ID shall be declared prior to the use of level 3 and level 4 types of property code description. See “#DEFAULT_SUPPLIER” for more detail.

Level 2 should be used only for maintaining compatibility with the legacy dictionaries based on ISO 13584-42:1998, and IEC 61360-2:1998 and 2002. For any other purposes, the use of level 2 shall be strictly prohibited.

Example: This example is described using the level 3 notation (i.e. the versions of property are omitted).

#PROPERTY_ID,0140/TOPAS#P000001,0140/TOPAS#P001089.MAX,
0140/TOPAS#P001089.MIN,0140/TOPAS#000894.EN,
0114/TOPAS#P000894.FR,

Spreadsheet view: It will be displayed by a spreadsheet application as in Figure 13.

#PROPERTY_ID	0140/TOPAS# P000001	0140/TOPAS# P001089.MAX	0140/TOPAS# P001089.MIN	0140/TOPAS# P000894.EN	0140/TOPAS# P000894.FR

Figure 13 – Display example of property ID

5.11.19 Preferred name of the property

Keyword: #PROPERTY_NAME.<lang>

Name: Preferred name of the property

Definition: preferred name of the property specified by the property ID, optionally in the language designated by a language code

Description: The information is provided by a parcel server, and any change of the information of this attribute by user shall not affect the behaviour of the system. Language for the preferred name may be specified using ISO 639. A further extension may be specified by attaching a country

code to the language code after a hyphen. Thus, a preferred name in English spoken in the United Kingdom may be specified as "#PROPERTY_NAME.EN-GB". When an appropriate name in the specified language is not available, the cell need not be filled. The keyword corresponds to the preferred_name of property specified in IEC 61360-2 or ISO 13584-42, when applied to the exchange of libraries based on the common dictionary schema. If the data type of the property is level_type or translatable_string_type, the preferred name is not needed for each level or each language code of the property.

Category: optional - informative

Format: The keyword "#PROPERTY_NAME.<lang>" is described in the first column. The extension "<lang>" means a two-letter language code defined in ISO 639-1. The preferred names are described in the second and following columns. Each preferred name relates to the property designated by the property ID code which is described in the line #PROPERTY_ID.

Example: #PROPERTY_NAME.EN,CONSORTIUM STANDARD,PH MEASURING,PH MEASURING,COMPANY NAME,COMPANY NAME,

#PROPERTY_NAME.FR,NORME D'ASSOCIATION,MESURE PH, MESURE PH,NOM ENTREPRISE,NOM ENTREPRISE,

Spreadsheet view: It will be displayed by a spreadsheet application as in Figure 14.

#PROPERTY_ID	0140/TOPAS# P000001	0140/TOPAS# P001089.MAX	0140/TOPAS# P001089.MIN	0140/TOPAS# P000894.EN	0140/TOPAS# P000894.FR
#PROPERTY_ NAME.EN	CONSORTIUM STANDARD	PH MEASURING	PH MEASURING	COMPANY NAME	COMPANY NAME
#PROPERTY_ NAME.FR	NORME D'ASSOCIATION	MESURE PH	MESURE PH	NOM ENTREPRISE	NOM ENTREPRISE

Figure 14 – Display example of preferred name

5.11.20 Definition

Keyword: #DEFINITION.<lang>

Name: Definition

Definition: statement about the meaning or the concept of the property specified by the property ID, optionally in the language designated by a language code

Description: The information is provided by parcel severs, and any change of the information by user shall not affect the behaviour of the systems. Language for the definition may be specified using ISO 639-1, possibly followed by a country code, separated by a hyphen. Thus, a definition in English (in general) may be specified as "#DEFINITION.EN". When an appropriate definition is not available in the corresponding language, the cell may be kept open.

Category: optional - informative

Format: The keyword "#DEFINITION.<lang>" is described in the first column. The extension "<lang>" means a two-letter language code defined in

ISO 639. The definitions are described in the second and following columns. Each definition relates to the property designated by the property ID code which is described in the line #PROPERTY_ID.

Example: #DEFINITION,“referential standard of a consortium, association, or organization excluding international standard and national standard”, “measuring span for pH measuring instrument”, “measuring span for pH measuring instrument”, “name of the company which manufactures the product”,

Spreadsheet view: It will be displayed by a spreadsheet application as in Figure 15.

#PROPERTY_ID	0140/TOPAS# P000001	0140/TOPAS# P001089.MAX	0140/TOPAS# 001089.MIN	0140/TOPAS# P000894.EN
#PROPERTY_NAME.EN	CONSORTIUM STANDARD	PH MEASURING	PH MEASURING	COMPANY NAME
#DEFINITION.EN	referential standard of a consortium, association, or organization excluding international standard and national standard	measuring span for pH measuring instrument	measuring span for pH measuring instrument	name of the company which manufactures the product

Figure 15 – Display example of definition

5.11.21 Note

Keyword: #NOTE

Name: Note for the property

Definition: statement that provides additional information about the definition that is essential for the understanding of the latter

Category: optional - informative

Format: The keyword “NOTE” shall be described in the first column. The note statements shall be described in the cells in and after the second column. Each note statement shall apply to the definition of the property specified by the value #PROPERTY_ID that is in the same column as the note statement.

Example: #NOTE,“referential standard of a consortium, association, or organization excluding international standard and national standard”, “measuring span for pH measuring instrument”, “measuring span for pH measuring instrument”, “name of the company which manufactures the product”,

5.11.22 Data type

Keyword: #DATATYPE

Name: Data type

Definition: attribute that designates the predefined classification of a unit of data for computer processing, of the property specified by the property ID

Description: The information is provided by the system that implements the parcelling spreadsheet structure, and any change of the information by user shall not affect the behaviour of the system.

Category: optional - informative

Format: The keyword “#DATATYPE” shall be described in the first column. The data types shall be described in the second and following columns. Each data type shall correspond to the property of the property ID specified in the line of #PROPERTY_ID in the same column.

Example: #DATATYPE,STRING_TYPE,LEVEL(MIN,MAX) OF REAL_MEASURE_TYPE, LEVEL(MIN,MAX) OF REAL_MEASURE_TYPE, TRANSLATED_STRING, TRANSLATED_STRING,

Spreadsheet view: It will be displayed by a spreadsheet application as in Figure 16.

#PROPERTY_ID	0140/TOPAS# P000001	0140/TOPAS# P001089.MAX	0140/TOPAS# P001089.MIN	0140/TOPAS# P000894.EN	0140/TOPAS# P000894.FR
#DATATYPE	STRING_ TYPE	LEVEL(MIN,MAX) OF REAL_MEASURE_ TYPE	LEVEL(MIN,MAX) OF REAL_MEASURE_ TYPE	TRANSLATED_ STRING	TRANSLATED_ STRING

Figure 16 – Display example of data type

NOTE Data types can be constructed from the primitive types. Examples of primitive types are shown in Annex D.

5.11.23 Unit of measurement

Keyword: #UNIT

Name: Unit of measurement

Definition: information about the unit of measurement of the property specified by the property ID

Description: The information will be provided by the parcel servers with an aim to help user to understand the property specification, and any change of the information by user shall not affect the behaviour of the parcel servers.

Category: optional - informative

Format: The keyword “#UNIT” will be described in the first column. The units will be described in the cells corresponding to the second and following columns. Each such unit corresponds to the property which is described in the #PROPERTY_ID line.

Example: #UNIT,,pH,pH,,,

Spreadsheet view: It will be displayed by a spreadsheet application as in Figure 17.

#PROPERTY_ID	0140/TOPAS# P000001	0140/TOPAS# P001089.MAX	0140/TOPAS# P001089.MIN	0140/TOPAS# P000894.EN	0140/TOPAS# P000894.FR
#UNIT		pH	pH		

Figure 17 – Display example of unit of measurement

5.11.24 Requirement

Keyword: #REQUIREMENT

Name: Requirement

Definition: designation of the necessity of the value of property in data section

Description: The values in the cells shall be either blank or one of the followings; ”CONST”, “KEY”, “NOT_NULL”, “MANDATORY”, “OPTIONAL” or “OBSOLETE”. When it is blank, it is equivalent to the designation as “OPTIONAL” while the values “MANDATORY”, “OPTIONAL”, “OBSOLETE” may be shorthanded as “MAND”, “OPT”, and ”OBS”, respectively. OBSOLETE is a kind of OPTIONAL, but an item at hand is not appropriate for use in a new definition. NOT_NULL means that the value is mandatory and it shall not be null. A blank field shall be equivalent to OPT, except the case, when the requirements for the properties are not determined yet such as in a parcel template for determining the requirements.”CONST” means the value is fixed at a constant value within the (meta-)class.

Category: optional - functional

Format: The keyword “#REQUIREMENT” shall be stated in the instruction column and the cases of the letters of the keyword shall be ignored. In the following cells a reserved word “KEY” shall be noted in every cell where the property corresponding to the property ID shall be treated as the key or an element of the (composite) key of the database to be created.

Example: #REQUIREMENT,KEY,,,,MANDATORY,

Spreadsheet view: It will be displayed by a spreadsheet application as in Figure 18.

#PROPERTY_ID	0140/TOPAS#P000001	0140/TOPAS# P001089.MAX	0140/TOPAS# P001089.MIN	0140/TOPAS# P000894.EN	0140/TOPAS# P000894.FR
#REQUIREMENT	KEY				MANDATORY

Figure 18 – Display example of key

5.11.25 Alternative units of measurement

Keyword: #ALTERNATIVE_UNITS

Name: Alternative units

Defintion: information about other units of measurement that may be used for the property specified by the property ID

Description: The information will be provided by the parcel servers with an aim to help user to understand the property specification, and any change of the information by user shall not affect the behaviour of the parcel servers.

Category: optional - informative

Format: The keyword “#ALTERNATIVE_UNITS” will be described in the first column. Lists of alternative units will be described in the cells

corresponding to the second and following columns. Each such list of alternative units corresponds to the property which is designated by an identifier in the #PROPERTY_ID line.

Example: #ALTERNATIVE_UNITS,,(bar, Torr),,,

Spreadsheet view: It will be displayed by a spreadsheet application as in Figure 19.

#PROPERTY_ID	0140/TOPAS# P000001	0140/TOPAS# P001089.MAX	0140/TOPAS# P001089.MIN	0140/TOPAS# P000894.EN	0140/TOPAS# P000894.FR
#UNIT		Pa	Pa		
#ALTERNATIVE _UNITS		(bar, Torr)	(bar, Torr)		

Figure 19 – Display example of alternative units

5.11.26 Variable prefix for the unit

Keyword: #VARIABLE_PREFIX_UNIT

Name: Variable prefix for the unit

Definition: Unit whose prefix other than default one can be selected

Description: Prefix other than the default may be selected from the given list for description of the value.

A pair of cells are used for the description: In the header section, the first cell describes the unit with default prefix. The second cell describes the selectable prefixes, enclosed in a pair of parentheses. In the data section, the first cell section describes the value, the second cell describes the unit with the actual prefix, not just the prefix itself. In case of level type, the same prefix shall be applied to all the elements of the data type. Only when the Variable prefix for the unit is used, this keyword should appear in the keyword column.

For the unit without variable prefixes, the default prefix may be written in the cell corresponding to the line with the keyword.

Category: optional - informative

Format: The keyword “#VARIABLE_PREFIX” shall be described in the keyword column.

Example: #VARIABLE_PREFIX_UNIT , μm, (μm, nm)

Spreadsheet view: It will be displayed by a spreadsheet application as in Figure 20.

#PROPERTY_ID	0140/TOPAS#P000200.val	0140/TOPAS#P000200.pfx	0140/TOPAS#P000201	0140/TOPAS#P000202
#PROPERTY_NAME	Circuit width	Circuit width	Manufacturer	Production date
#DATA TYPE	REAL_TYPE	STRING_TYPE	STRING_TYPE	DATE_YEAR_TYPE
#UNIT	μm			
#VARIABLE_PREFIX_UNIT	μm	(μm, nm)		
	0.060	μm	ABC Corporation	2010-10-10
	50	nm	BCD Corporation	2001-02-30

Figure 20 – Display example of variable prefix unit

5.11.27 Super property

Keyword: #SUPER_PROPERTY

Name: Super property

Definition: Identifier of the super property of which this property is a specialization

Description: It is assumed that the instantiated values of this property are also the values of the super property, albeit some of the specialized attributes or conditions may not be applicable for the super property.

Super property shall be defined a priori as a permanent structural relationship, while an alternate ID including a super alternate ID is assignable posteriori, for local designation.

Category: optional - informative

Example: #SUPER_PROPERTY,

Spreadsheet view: It will be displayed by a spreadsheet application as in Figure 21.

#PROPERTY_ID	MDC_P001_6	MDC_P004_1	MDC_P004_2	MDC_P004_3
#SUPER_PROPERTY	MDC_P001	MDC_P004	MDC_P004	MDC_P004
#PROPERTY_NAME	Code	Preferred name	Synonymous name	Short name
#PATTERN	[A-Z_0-9]+			
#DATATYPE	STRING_TYPE	STRING_TYPE	STRING_TYPE	STRING_TYPE

Figure 21 – Display example of Super-property for properties

5.11.28 Alternate property ID

Keyword: #ALTERNATE_ID

Name: Alternate property ID

Definition: List of alternate property identifiers of the properties each of which is specified by a property ID.

Description: Generally the values of a property designated by its alternate property ID are assignable to the property designated by the property ID, and vice versa. Alternate property ID may be used for mapping between the two properties that have virtually the same extent but are assigned different IDs for historical reasons or for different prevailing practices of industries. If there are several alternate IDs for one property, the alternate IDs shall be enclosed in a pair of parentheses “(“ and “)”. In a special case, one of the two properties to be mapped is a specialization of the other, while in another special case the two properties exactly share the same extent. For such special cases, see #SUB_ALTERNATE_ID, #SUPER_ALTERNATE_ID, and #EQUIVALENT_ID.

Category: optional - informative

Example: #PROPERTY_ID,MDC_P001_5,MDC_P002_1,MDC_P004_1.DE, MDC_P005.DE,

#ALTERNATE_ID,C01,(C02,C03),A01(DE),A04(DE),

Spreadsheet view: It will be displayed by a spreadsheet application as in Figure 22.

#PROPERTY_ID	MDC_P001_5	MDC_P002_1	MDC_P004_1.DE	MDC_P005.DE
#ALTERNATE_ID	C01	(C02, C03)	A01(DE)	A04(DE)
#PROPERTY_NAME.EN	Code	Version	Preferred name	Definition

Figure 22 – Display example of alternate property ID

5.11.29 Super alternate ID

Keyword: #SUPER_ALTERNATE_ID

Name: Super alternate ID

Definition: alternate property identifier(s) of the property specified by the property ID, where all the values of a property designated by the property ID are assignable to the super property designated by the super alternate ID, and some of the values of the super property may be assignable to the property designated by the property ID

Description: Super alternate property ID is a specialization of the alternate property ID, where the extent of the property designated by #SUPER_ALTERNATE_ID is larger than the one designated by # ALTERNATE_ID. When the exact relationship between the two properties to be mapped is not known, use of this key word should be refrained. When there are several properties recognized as superset of this property, the IDs are listed in a pair of parentheses, “(“ and “)”.

The keywords, #SUPER_ALTERNATE_ID, #SUB_ALTERNATE_ID, #EQUIVALENT_ID and # ALTERNATE_ID may exist together in one parcel, however an IRDI appearing in one of those shall not reappear in another.

Category: optional - informative

Example: #PROPERTY_ID,MDC_P001_5,MDC_P002_1,MDC_P004_1.DE, MDC_P005.DE,

#ALTERNATE_ID,C01,(C02,C03),A01(DE),A04(DE),

#SUPER_ALTERNATE_ID, J01,J02 ,(A01.JP, A01.KR), (A04.JP, A04.KR)

Spreadsheet view: It will be displayed by a spreadsheet application as in Figure 23.

#PROPERTY_ID	MDC_P001_5	MDC_P002_1	MDC_P004_1.DE	MDC_P005.DE
#ALTERNATE_ID	C01	(C02, C03)	A01(DE)	A04(DE)
#SUPER_ALTERNATE_ID	J01	J02	(A01.JP, A02.KR)	(A04.JP, A04.KR)
#PROPERTY_NAME.EN	Code	Version	Preferred name	Definition

Figure 23 – Display example of super alternate property ID

5.11.30 Sub-alternate ID of property

Keyword: #SUB_ALTERNATE_ID

Name: Sub-alternate ID of property

Definition: alternate property identifier(s) of the property specified by the property ID, where all the values of the sub-property designated by the sub-alternate ID are assignable to the property designated by the property ID, and some of the values of the property may be assignable to some sub-property designated by the sub-alternate ID of property

Description: Sub-alternate ID of property is a specialization of alternate property ID, where the extent of the property designated by #SUB_ALTERNATE_ID is smaller than the one designated by #PROPERTY_ID. When the exact relationship between the two properties to be mapped is not known, use of this key word should be refrained. When there are several properties recognized as sub-property, the IDs are listed in a pair of parentheses, “(“and “)”.

The keywords, #SUPER_ALTERNATE_ID, #SUB_ALTERNATE_ID, #EQUIVALENT_ID and # ALTERNATE_ID may exist together, but an ID appearing in one of those shall not reappear in another.

Category: optional - informative

Example: #PROPERTY_ID,MDC_P001_5,MDC_P002_1,MDC_P004_1.DE, MDC_P005.DE,

#ALTERNATE_ID,C01,(C02,C03),A01(DE),A04(DE),

#SUB_ALTERNATE_ID, J01,J02 ,(A01.JP, A01.KR), (A04.JP, A04.KR)

Spreadsheet view: It will be displayed by a spreadsheet application as in Figure 24.

#PROPERTY_ID	MDC_P001_5	MDC_P002_1	MDC_P004_1.DE	MDC_P005.DE
#ALTERNATE_ID	C01	(C02, C03)	A01(DE)	A04(DE)
#SUB_ALTERNATE_ID	J01	J02	(A01.JP, A02.KR)	(A04.JP, A04.KR)
#PROPERTY_NAME.EN	Code	Version	Preferred name	Definition

Figure 24 – Display example of sub-alternate property ID

5.11.31 Equivalent property ID

Keyword: #EQUIVALENT_ID

Name: Equivalent property ID

Definition: alternate property identifier(s) of the property specified by the property ID, where all the values of the property designated by the equivalent property ID are assured to be assignable to the property designated by the property ID, and vice versa

Description: Equivalent ID is a specialization of alternate property ID, where the extent of the property designated by #EQUIVALENT_ID and the one designated by # PROPERTY_ID share precisely the same extent, albeit some superficial differences in textual explanations.

When the exact relationship between the two properties to be mapped is not exactly known, use of this key word should be refrained. When there are several properties recognized as equivalent the IDs are listed in a set of parentheses, “(“and “)”.

The keywords, #SUPER_ALTERNATE_ID, #SUB_ALTERNATE_ID, #EQUIVALENT_ID and #ALTERNATE_ID may exist together, but an ID appearing in one of those shall not reappear in another.

Category: optional - informative

Example: #PROPERTY_ID,MDC_P001_5,MDC_P002_1,MDC_P004_1.DE, MDC_P005.DE,

#EQUIVALENT_ID,C01,(C02,C03),A01(DE),A04(DE),

Spreadsheet view: It will be displayed by a spreadsheet application as in Figure 25.

#PROPERTY_ID	MDC_P001_5	MDC_P002_1	MDC_P004_1.DE	MDC_P005.DE
#EQUIVALENT_ID	C01	(C02, C03)	A01(DE)	A04(DE)
#PROPERTY_NAME.EN	Code	Version	Preferred name	Definition

Figure 25 – Display example of equivalent property ID

5.11.32 ID for the unit of measurement

Keyword: #UNIT_ID

Name: ID for the unit of measurement

Definition: identifier to uniquely reference the unit of measurement (UoM) used in a property, being specified by UNIT_ID listed in the #UNIT_ID line, noted in the same column as the property ID in a parcel

Description: The ID for the unit of measurement may be recognized as a globally unique identifier only by parcelling tools. Thus, it requires translation into the descriptions available in IEC 61360-2 or ISO 13584-42, when conversion into ISO 13584-25 is requested, by extracting the pieces of information stored in the fields named “unit structure”, “unit in SGML” and “unit in text” in a UoM meta-class. Since there are multiple ways of

describing the same unit in a string, an automated reverse mapping is not foreseeable for the current IEC 61360-2 or ISO 13584-42.

Category: optional - informative

Example: #UNIT_ID,,0112/1///13584_35_1.U001,0112/1///13584_35_1.U123,

Spreadsheet view: It will be displayed by a spreadsheet application as in Figure 26.

#PROPERTY_ID	0140/TOPAS#P000001	0140/TOPAS#P001089	0140/TOPAS#P001234
#DATATYPE	STRING_TYPE	INT_MEASURE_TYPE	REAL_MEASURE_TYPE
#UNIT_ID		0112/1///13584_35_1#U001	0112/1///13584_35_1#U123

Figure 26 – Display example of ID for the unit of measurement

5.11.33 Property value format

Keyword: #VALUE_FORMAT

Name: Property value format

Definition: specification of the type and length of the representation of the value of a property, where it shall be interpreted as a maximum value format for storage in a server system which generated the parcel

Description: The detailed specification of value format is explained in IEC 61360-2 or ISO 13584-42.

Category: optional - informative

Example: #VALUE_FORMAT,M..14,M..70,M..70,

Spreadsheet view: The following Figure 27 illustrates how the value format shall be described for each property specified by a respective property ID.

#PROPERTY_ID	MDC_001_3	MDC_004_1.EN	MDC_004_1.DE
#PROPERTY_NAME.EN	CODE	PREFERRED NAME	PREFERRED NAME
#PROPERTY_NAME.DE	KENNUNG	BENENNUNG	BENENNUNG
#VALUE_FORMAT	M..14	M..70	M..70

Figure 27 – Display example of value format

5.11.34 Identifier encoding

Keyword: #ID_ENCODE

Name: Identifier encoding

Definition: specification of the global encoding method of identifiers

Description: When the identifier encoding is not specified, ICID encoding shall be used as default. As a value, beside ICID currently only IRDI and ISO 29002 may be specified. The difference between ICID and IRDI are from syntactical viewpoint only in the number of “#” between DI and VI. For IRDI and ISO 29002 abridging only either RAI or VI is not allowed. Namely, when some entries are abridged, both RAI and VI

shall be omitted with appropriately setting #DEFAULT_SUPPLIER and #DEFAULT_VERSION, while in data section, #DEFAULT_DATA_SUPPLIER and #DEFAULT_DATA_VERSION shall be set, accordingly. Otherwise, no abridgement shall be intended.

Category: optional – functional

Example: #ID_ENCODE:=ICID

#ID_ENCODE:=IRDI

#ID_ENCODE:= ISO_29002

5.11.35 Cell delimiter

Keyword: #DELIMITER

Name: Cell delimiter

Definition: specification of the character used for marking the boundary of neighbouring cells in a current file

Description: Among the different releases and versions of a spreadsheet application in some countries, sometimes a different cell delimiter character such as ";"is used instead of "," to mark the boundaries between two neighbouring cells (columns), for the reason that "," is used predominantly for denoting a decimal mark in the everyday life of the countries. It is all the more important when xls or OpenXML format is used instead of CSV format for parcel exchange, because users tend to see a parcel file through a specific application, in such a manner that the application pre-processes the file assuming the use of local cell delimiter character for separating the cells in the file. The designated delimiter should be a single character. To signify the delimiter, the used character shall be enclosed within a pair of single quotation marks; i.e., "'" and "'".

It is requested on a parcel tool that exports a parcel file to an external system that the designation of the value for "#DELIMITER" and the actual cell delimiters in the parcel be set identical.

Category: optional – informative

Example: #DELIMITER:=';'

5.11.36 Decimal mark

Keyword: #DECIMAL

Name: Decimal mark

Definition: character used to separate the integer part and the fractional part of a number expressed in decimal form

Description: To cater for a wider range of users and applications in many countries, this standard does not intend to normalize the decimal mark as a full

stop “.” nor as a comma “,”. Each of them is used in quite a number of countries. The allowed value(s) is (are) dependent on the regional edition of the base software on which the parcel tool is implemented.

In the case that the decimal mark is designated to be a comma, then by default, separators among data in a cell, such as one between the numbers of a set or list type property, are supposed to be replaced by a semicolon ‘;’ as well as the case, when the cell delimiter is designated to be ‘;’ instead of a comma ‘,’.

It is requested on a parcel tool that exports a parcel file to an external system that the designation of the value for “#DECIMAL” and the actual decimal mark used in a parcel should be consistent.

Category: optional – informative

Example: #DECIMAL:=’,’

5.11.37 Pattern constraint

Keyword: #PATTERN

Name: Pattern constraint

Definition: String pattern that the string type value of a property shall follow

Description: The keyword corresponds to the pattern constraint for property defined in ISO 13584-42:2010 that is to be duplicated as IEC 61360-2:2012.

It is expected that the designation for this attribute and the value format for the same property shall be well aligned.

Category: optional – informative

Example: # PATTERN,,, [0-9]{4}\-[0-9]{2}\-[0-9]{2}

Spreadsheet view: The following Figure 28 illustrates how the pattern constraint shall be described for each property specified by a respective property ID.

#PROPERTY_ID	MDC_001_3	MDC_001_4.EN	MDC_P003_1
#PROPERTY_NAME.EN	CODE	PREFERRED NAME	Date of original definition
#DATATYPE	ICID_STRING	STRING_TYPE	DATE_ TYPE
#PATTERN			[0-9]{4}\-[0-9]{2}\-[0-9]{2}

Figure 28 – Display example of pattern constraint

5.11.38 Relational constraint

Keyword: #RELATION

Name: Relational constraint

Definition: constraint that works as a predicate or functional relationship among

several properties and/or classes

Description: This constraint works on several properties and/or classes, and has as its subtype a functional relation and a predicative relation. For the detailed specification of the relationship including the function body of a functional relation, see a row of relation meta-class designated by the relation ID. A function may have codomains of other functions in its domain as arguments, but functions shall not form a circular link of calls. However, forming an acyclic graph is allowed. Moreover, a property shall not be the codomains of two distinct functions. Neither a functional relation shall have two codomains. A property may be a domain for two or more distinct predicates. Note that the concept of the relational constraint includes a unary relation that works on a single property.

Category: optional – informative

Format: In the header section, in the row marked “#RELATION” in the instruction column, a function or predication designator shall be marked, in the following way:

For a function: “FCOD(rel_id)” or “FDOM(rel_id)” , and

for a predication: “PRED(rel_id)”).

where “rel_id” means the ICID of the relation defined in a relation meta-class and FDOM and FCOD mean domain and co-domain (function value) of a function, respectively. PRED means a domain of a predicative relation. Properties that have the same relation ID signify that they are controlled by the same relation, of which the relation type and its role are indicated by abbreviated terms, FCOD, FDOM, and PRED. When there are two terms in a cell, they are enclosed in a pair of quotation marks in a CSV file.

Example: #PROPERTY_ID,EXCIM_001,EXCIM_002,EXCIM_003,EXCIM_004
#RELATION,FCOD(rel001),FDOM(rel001),"FDOM(rel001), PRED(rel002)", PRED(rel002)

Spreadsheet view: The following Figure 29 illustrates how the relational constraint shall be described and applied for each property specified by a respective property ID.

#PROPERTY_ID	EXCIM_001	EXCIM_002	EXCIM_003	EXCIM_004
#PROPERTY_NAME.EN	Electricity consumption	Electric voltage at measurement	Electric current at measurement	Rated capacitance of the circuit
#UoM	W	V	A	F
#RELATION	FCOD (rel001)	FDOM(rel001)	FDOM(rel001), PRED(rel002)	PRED(rel002)

Figure 29 – Display example of relational constraint

5.12 Data section for instances

5.12.1 General

In this standard, each line in the data section designates a set of property values that collectively characterize an instance, i.e., a list of property-value pairs that one part or one product belonging to a class has. If in the first column of the line, namely on the instruction column within the data section, any “#” is marked at the head of a sequence of letters or

values, the line shall be treated as a comment line. Users may insert as many comment lines as they wish. This functionality may also be used for commenting out some of the existing value instances.

This part of IEC 62656 uses the data type notations that are analogous to IEC 61360-2 or ISO 13584-42. The detailed comparison and correspondence between the data type notations, used in this standard and in IEC 61360-2/ISO 13584-42 are elucidated in Annex D.

The following sections explain how an instance value shall be described for each data type, by a human user, or by a parcelling tool.

5.12.2 Enumeration types or non quantitative types

For such data types as non_quantitative_code_type and non_quantitative_int_type, only one value code within a list of the value codes predetermined for each of those data types shall be written in the data section as an instance value. In other words, the meaning of the code shall not be entered as the value.

Figure 30 gives a typical display example on a spreadsheet application.

#PROPERTY_ID	0140/TOPAS#P001	0140/TOPAS#P002	0140/TOPAS#P003	0140/TOPAS#P004
#PROPERTY_NAME	Color of the wall	Color of the roof	Type of the door	Type of the window
#DATATYPE	ENUM_CODE_TYPE(EXX_001)	ENUM_CODE_TYPE(EXX_001(red,blue))	ENUM_INT_TYPE(EXX_999)	ENUM_INT_TYPE(EXX_999(1,2))
	blue	red	2	1
	yellow	blue	3	2

Figure 30 – Display example of ENUM_INT_TYPE or ENUM_CODE_TYPE

NOTE 1 For example, in the description “ENUM_CODE_TYPE(EXX_001(red, blue))”, “EXX_001” is the ID for the enumeration list and “red” and “blue” are the value codes available for it. The value codes are listed just for information purpose, and they can be omitted in the “#DATATYPE” row, just like the expression, “ENUM_CODE_TYPE(EXX_001)”.

NOTE 2 In the POM, when an attribute named ”the number of selections” (MDC_P045) is specified as more than one (for example, from one to two by “(1,2)”), a list of codes (in this case 2 codes) at most enclosed within a pair of parentheses can be entered in the data section as values. For example, ”(red, white)” for the values of a property that explains the colours of a national flag.

NOTE 3 In the case, an attribute named ”type of the list” (MDC_P046) is specified simultaneously as “PERM” (for permutation) with the ”the number of selections”, the order of the codes of the list has a significance.

5.12.3 Level type

For each element of a level type (MIN, MAX, TYP, NOM) to be used in a library exchange file, one column shall be used. If some of the elements are not used in all the instances of the file, the columns for those elements are not necessary. The elements of the level type are identified by a dot extension with a three-letter code added after the property ID codes aligned in the #PROPERTY_ID row.

Figure 31 gives a typical display example on a spreadsheet application, which uses only “MIN.” and “MAX.” among the four optional attributes of the level type.

#PROPERTY_ID	0140/TOPAS# P000001	0140/TOPAS# P001089.MAX	0140/TOPAS# P001089.MIN	0140/TOPAS# P000894.EN	0140/TOPAS# P000894.FR
	JIS	1000	2000	JAPAN Corporation	JAPON SA
	CEN	20	23	FRENCH Ltd.	FRANÇAIS SA

Figure 31 – Display example of LEVEL_TYPE

NOTE JAPAN Corporation and FRENCH Ltd., and their equivalents in French language are used just as examples and their names are totally fictious. Neither JAPAN Corporation nor FRENCH Ltd., exists in the real world.

5.12.4 String type

For each property of STRING_TYPE to be used in a library exchange file, one column shall be allocated. If the language to construe the meaning of the value of string type needs to be designated, it shall be done so by assigning a two-letter language code according to ISO 639-1 to #SOURCE_LANGUAGE, as described in 5.11.9.

5.12.5 Translatable string type

For each language of a property of TRANSLATABLE_STRING_TYPE to be used in a library exchange file, one column shall be allocated. The language element of the TRANSLATABLE_STRING_TYPE is specified by a dot extension with a two-letter code added after property ID codes aligned in the #PROPERTY_ID row.

Figure 32 gives a typical display example on a spreadsheet application.

#PROPERTY_ID	0140/TOPAS# P000001	0140/TOPAS# P001089.MAX	0140/TOPAS# P001089.MIN	0140/TOPAS# P000894.EN	0140/TOPAS# P000894.FR
	JIS	1000	2000	JAPAN Corporation	JAPON SA
	CEN	20	23	FRENCH Ltd.	FRANÇAIS SA

Figure 32 – Display example of TRANSLATABLE_STRING_TYPE

NOTE JAPAN Corporation and FRENCH Ltd., and their equivalents in French language are used just as examples and their names are totally imaginary. Neither JAPAN Corporation nor FRENCH Ltd., exists in the real world.

5.12.6 Boolean type

For a Boolean type property, either “TRUE” or “FALSE” is expected for its value.

Figure 33 gives a typical display example on a spreadsheet application.

#PROPERTY_ID	TOPAS_Pxxx
#PROPERTY_NAME.EN	Export restriction
#DEFINITION.EN	presence of an export restriction on the product
#DATATYPE	BOOLEAN_TYPE
	TRUE

Figure 33 – Display example of BOOLEAN_TYPE

5.12.7 Class reference type (Class instance type)

In the library exchange file (hereafter designated as “Main file”), an instance value of this data type shall contain only one class_ID and its reference conditions as a list of property-value pairs. Class reference type is be also called as “class instance type” and may be so specified instead of class_reference_type.

Figure 34 gives a typical display example on a spreadsheet application.

#PROPERTY_ID	0140/TOPAS#P000001
#DATATYPE	CLASS_REFERENCE_TYPE(0140/TOPAS#C000999)
	0140/TOPAS#C000888,{(0140/TOPAS#P000101,Color),(0140/TOPAS#P00 0102, TRUE),(0140/TOPAS#P000103, 200)}
	0140/TOPAS#C000888,{(0140/TOPAS#P000101, Red),(0140/TOPAS#P000102, FALSE),(0140/TOPAS#P000103, 300)}

Figure 34 – Display example of CLASS_INSTANCE_TYPE

NOTE 1 In describing the reference mechanism by class_reference type, in the first place, the full ICID of the referenced class or the ICID except VI is noted, and a list of property-value pairs follows the ICID, being placed within a pair of parentheses.

NOTE 2 0140/TOPAS#C000888 is the class under the class whose code is 0140/TOPAS#C000999.

NOTE 3 Shortcut for the notation of supplier_ID is available. Setting “#DEFAULT_SUPPLIER:= 0140/TOPAS/”, an ICID notation like “0140/TOPAS#C000999” can be reduced to “P501_C000999”.

NOTE 4 Shortcut for the notation of instances in the data section containing an iCID is possible with an appropriate setting of the “#DEFAULT_DATA_SUPPLIER” and “#DEFAULT_DATA_VERSION”.

5.12.8 Aggregate type

Rules of description for aggregate types are summarized in the following list:

Data elements of aggregate types shall be inserted within “{ }” or “( )”;

– A pair of brace “{ }” signifies that there is no specified order among the data elements. Thus it shall be applied to the data of SET or BAG data types;
– A pair of parentheses “( )” signifies that there is a specified order among the data elements. Thus it shall be applied to the data of LIST or ARRAY data type;
– Data elements shall be separated by commas. Spaces before and between characters shall be interpreted as intended spaces for STRING type properties;
– If data elements themselves include “{”, “}”, “(” or “)” they shall be inserted within a pair of double quotation marks;
– If data elements of aggregate types include a double quotation, it shall be placed between a pair of double quotation marks.
– Following are the examples of description compatible with the above notation rules including some of the above cases:

EXAMPLE 1 SET OF STRING_TYPE

Text definition of instructions in this part of IEC 62656 shall be as follows:

#PROPERTY_ID,TOPAS_Pxxx,
#PROPERTY_NAME.EN,COLOR,
#DATATYPE, “SET(1,4) OF STRING_TYPE”,
, “{red,white,blue,black}”,

Figure 35 gives a display example of the Example 1 on a spreadsheet application.

#PROPERTY_ID	TOPAS_Pxxx
#PROPERTY_NAME.EN	COLOR
#DATATYPE	SET(1,4) OF STRING_TYPE
	{red,white,blue,black}

Figure 35 – Display example of SET OF STRING_TYPE

EXAMPLE 2 LIST OF STRING_TYPE

Text definition of instructions in this part of IEC 62656 shall be as follows:

#PROPERTY_ID,TOPAS_Pxxx,
#PREFNEME.EN,COLOR,
#DATATYPE, “LIST(1,5) OF STRING_TYPE”,
, “(red,white,blue,black)”,

Figure 36 gives a typical display example of the Example 2 on a spreadsheet application.

#PROPERTY_ID	TOPAS_Pxxx
#PROPERTY_NAME.EN	COLOR
#DATATYPE	LIST(1,4) OF STRING_TYPE
	(red,white,blue,black)
	.....

Figure 36 – Display example of LIST OF STRING_TYPE

EXAMPLE 3 LIST OF TRANSLATABLE_STRING

Text definition of instructions in this part of IEC 62656 shall be as follows:

#PROPERTY_ID,TOPAS_Pxxx.EN,TOPAS_Pxxx.JA,
#PREFNEME.EN,Traded product,Traded product,
#PROPERTY_NAME.FR,Produits échangés,Produits échangés,
#DATATYPE, “LIST(1,4) OF TRANSLATABLE_STRING”, “LIST(1,4) OF TRANSLATABLE_STRING”,
, “(motor, computer, display, memory}”, “(moteur, ordinateur, écran, mémoire)”,

Figure 37 gives a typical display example of the Example 3 on a spreadsheet application.

#PROPERTY_ID	TOPAS_Pxxx.EN	TOPAS_Pxxx.FR
#PROPERTY_NAME.EN	Traded product	Traded product
#PROPERTY_NAME.FR	Produits échangés	Produits échangés
#DATATYPE	LIST(1,4) OF TRANSLATABLE_STRING	LIST(1,4) OF TRANSLATABLE_STRING
	(motor, computer, display, memory)	(moteur, ordinateur, écran, mémoire)

Figure 37 – Display example of LIST OF TRANLATABLE_STRING_TYPE

The order of the appearance of the elements shall be the same in the languages used.

Thus motor (EN) corresponds to moteur (FR), and so on, as in the following Table 2.

Table 2 – Example of correspondence within multiple languages

English	French
motor	moteur
computer	ordinateur
display	écran
memory	mémoire

EXAMPLE 4 SET OF LEVEL OF INT_MEASURE_TYPE

In case that a set of 2 triplets, i.e., (Maximum input voltage 240 V, Rated input voltage 120 V, Minimum input voltage100 V), and (Maximum input voltage 300 V, Rated input voltage 200 V, Minimum input voltage 110 V) need to be expressed, specification shall be as follows:

#PROPERTY_ID,TOPAS_Pxxx,
#DATATYPE,“SET(1,3) OF LEVEL(MIN,NOM,MAX) OF INT_MEASURE_TYPE”,
#UNIT,V,
,“{(100,120,240),(110,200,300)}”,

Figure 38 gives a typical display example of the EXAMPLE 4 on a spreadsheet application.

#PROPERTY_ID	TOPAS_Pxxx
#DATATYPE	SET OF LEVEL(MIN,NOM,MAX) OF INT_MEASURE_TYPE
#UNIT	V
	{(100,120,240),(110,200,300)}
	.....

Figure 38 – Display example of SET OF LEVEL OF INT_MEASURE_TYPE

NOTE The order of description is MIN, NOM, TYP,MAX. Any element in the order can be skipped.

EXAMPLE 5 SET OF SET OF STRING_TYPE

Text definition in this part of IEC 62656 shall be as follows:

#PROPERTY_ID,TOPAS_Pxxx,
#PROPERTY_NAME.EN,COLOR,
#DATATYPE,”SET(1,2) OF SET(1,2) OF STRING_TYPE”,
,”{{AAA,BBB},{AAA,XXX}}”,

Figure 39 gives a typical display example of the Example 5 on a spreadsheet application PROPERTY_ID	TOPAS_Pxxx
#PROPERTY_NAME.EN	COLOR
#DATATYPE	SET(1,2) OF SET(1,2) OF STRING_TYPE
	{{AAA,BBB},{AAA,XXX}}
	.....

Figure 39 – Display example of SET OF SET OF STRING_TYPE

5.12.9 Named type

The named_type entity provides for referring to other types constructed or renamed using data_type_BSU. The actual construction or renaming shall be done using data type parcel, whose specification is given in Datatype meta-class. In the following Figure 40,

data_type_BSU of the constructed type defined in datatype meta-class is used in the parentheses placed after the keyword NAMED_TYPE.

#PROPERTY_ID	TOPAS_Pxxx
#DATATYPE	NAMED_TYPE(type_id)
#UNIT	
	650
	.....

Figure 40 – Display example of NAMED TYPE

5.12.10 Placement types

Placement types are used to define a geometric point, placement of an object and its direction in 2D (2-dimensional) or 3D (3-dimensional) environment. The following subtypes are available:

– PLACEMENT_2D_TYPE,
– PLACEMENT_3D_TYPE,
– AXIS1_PLACEMENT_2D_TYPE,
– AXIS1_PLACEMENT_3D_TYPE ,
– AXIS2_PLACEMENT_2D_TYPE,
– AXIS2_PLACEMENT_3D_TYPE.

For all the types in the above list, "_TYPE" may be omitted from the text of specification. Moreover, for "PLACEMENT_3D_TYPE", "AXIS1_PLACEMENT_3D_TYPE" and "AXIS2_PLACEMENT_3D_TYPE" may be shorthanded to "PLACEMENT", "AXIS1_PLACEMENT" and "AXIS2_PLACEMENT". In brief, when the letters indicating the dimensionality of an embedding space are omitted, it always assumes a placement in 3-dimensional space. For the use of these data types, see Table D.2 for more details.

In the case of ISO 10303-42 standard, where these data types are originally defined, AXIS1_PLACEMENT_2D, AXIS1_PLACEMENT_3D, AXIS2_PLACEMENT_2D and AXIS2_PLACEMENT_3D may inherit an attribute named "location" that is in fact of CARTESIAN_POINT type in 2D or 3D space, from a generic geometric entity named "placement". The attribute is useful and even indispensable in defining an object shape resulting from a small number of Boolean operations (set-theoretic Boolean operations) of primitive shapes. In the case of the POM, it is achieved by associating a condition type property of the placement type(i.e., CARTESIAN_POINT) to the geometric shape that uses either one of AXIS1_PLACEMENT_2D, AXIS1_PLACEMENT_3D, AXIS2_PLACEMENT_2D or AXIS2_PLACEMENT_3D data types. See Annex N for additional information.

5.12.11 Entity instance type

Entity instance type is used to instantiate entities defined in ISO 10303 (STEP) standards. In fact, the placement types are selections of entities defined in ISO 10303-42 standard. In order to explicitly define all the attributes of a STEP entity, the user must predefine the entity structure using the datatype meta-class. Otherwise, standard parcel systems will leave it to the interpretation by the recipient system, just exporting the field as if it were text data.

6 Use of parcel for Domain Ontology description

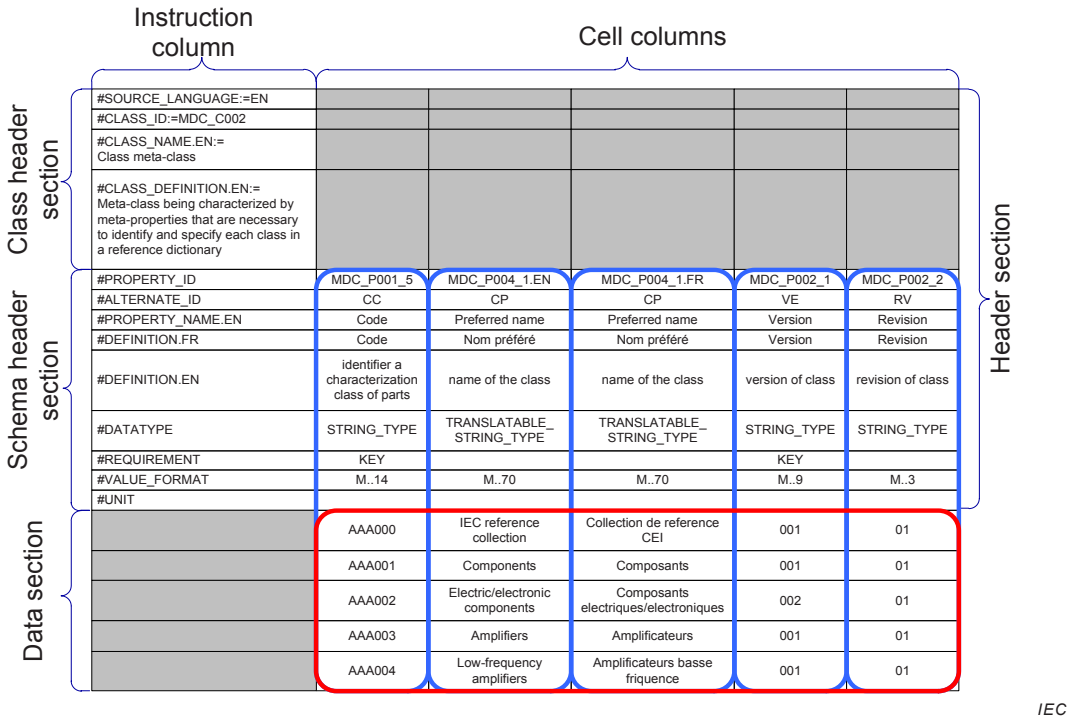
6.1 Dictionary as an instance of meta-dictionary

In case of parcel compliant spreadsheet structures, use for domain dictionary definition and use for library definition only differ in the lists of properties that each use of the parcels

deploys, but they remain the same in their basic structure. Thus, most of the descriptions and remarks about the instruction columns in the foregoing sections are still pertinent.

In terms of a dictionary conformant to IEC 61360-2/ISO 13584-42, a basic domain dictionary can be described with a set of 4 parcels that are meta-classes, while a typical domain dictionary requires a set of 6 parcels, with an addition of enumeration meta-class and term meta-class, which collectively define one complete reference dictionary. For some complex dictionary, however, three other types of meta-classes, i.e., Data type meta-class, Document meta-class, and UoM meta-class are extremely useful and important. In case of a use for importation of other types of dictionaries, such as for IEC 61968/61970(CIM) describing electric grid equipment, addition of the Relation meta-class which allows creation of various associations among the values of several properties will be helpful. Also, for practical database management of dictionary data entities, such as for IEC 61360 CDD, addition of an Object meta class to the panoply will be found useful for adding administrative attributes to each dictionary data object.

Such a dictionary, described as a set of instances of a higher level dictionary and transported in parts as a payload of a parcel, will be called hereafter “dictionary data” ,“domain dictionary”, or “Domain Ontology”, while the dictionary of the higher level that gives syntax to the dictionary data shall be called hereafter a “meta-dictionary”. This meta-dictionary corresponds to M3-M2 layer as previously depicted in Figure 2. Each class of the meta-dictionary in a form of parcel shall be called in its own right, a “meta-class”, when it needs to be distinguished from an ordinary class in a domain dictionary which latter is created as an instance of a set of meta-classes of the meta-dictionary. Likewise, each property used by the meta-classes to describe an attributes of a dictionary element of a reference dictionary shall be called a “meta-property”, when a distinction is needed in appellation, between a property of a meta-class of a meta-dictionary and a property of a class of a reference dictionary. A spreadsheet that represents one of the eleven meta-classes being characterized by some meta-properties in its schema header section may be referenced in general as a “dictionary parcel”. A typical configuration of the dictionary parcel is shown in Figure 41.



Instruction column	Cell columns				
#SOURCE_LANGUAGE:=EN					
#CLASS_ID:=MDC_C002					
#CLASS_NAME.EN:= Class meta-class					
#CLASS_DEFINITION.EN:= Meta-class being characterized by meta-properties that are necessary to identify and specify each class in a reference dictionary					
#PROPERTY_ID	MDC_P001_5	MDC_P004_1.EN	MDC_P004_1.FR	MDC_P002_1	MDC_P002_2
#ALTERNATE_ID	CC	CP	CP	VE	RV
#PROPERTY_NAME.EN	Code	Preferred name	Preferred name	Version	Revision
#DEFINITION.FR	Code	Nom préféré	Nom préféré	Version	Revision
#DEFINITION.EN	identifier a characterization class of parts	name of the class	name of the class	version of class	revision of class
#DATATYPE	STRING_TYPE	TRANSLATABLE_ STRING_TYPE	TRANSLATABLE_ STRING_TYPE	STRING_TYPE	STRING_TYPE
#REQUIREMENT	KEY			KEY	
#VALUE_FORMAT	M..14	M..70	M..70	M..9	M..3
#UNIT					
	AAA000	IEC reference collection	Collection de reference CEI	001	01
	AAA001	Components	Composants	001	01
	AAA002	Electric/electronic components	Composants electriques/electroniques	002	01
	AAA003	Amplifiers	Amplificateurs	001	01
	AAA004	Low-frequency amplifiers	Amplificateurs basse friquence	001	01

Figure 41 – Configuration of a dictionary parcel

Each row of the instance data in the data section of a dictionary parcel, where a value in each cell vertically aligned in a column corresponds to some of the meta-properties defined in the

schema header section of the parcel, describes the attribute values of the dictionary elements of a reference dictionary. As noted earlier, a meta-dictionary consists of the following 7 meta-classes into or out of which each specialized spreadsheet, i.e., a partitive dictionary parcel is milled:

- Dictionary;
- Supplier;
- Class;
- Property;
- Enumeration;
- Document;
- Datatype (named_type).

In addition to the above seven normative meta-classes, the following meta-classes may be supplied as an option:

- Object;
- UoM;
- Term;
- Relation.

Among the seven standardized parcelling sheets listed above, the first four parcels are mandatory for the exchange of a dictionary by means of the parcel format. Moreover if a property parcel sheet includes so called “enumeration type” properties, or to be exact, “non_quantitative” types as they are called in IEC 61360 and ISO 13584 series of standards, an enumeration parcel usually accompanies the property parcel, for clarifying the meaning of each enumeration code, namely, for the textual explanation of the option codes, used in those types of properties.

A parcel for object meta-class may be added to enable identification and specification of each row of data instance per se, as a data object, in the data section of a parcel. The object meta-class is used to define purely administrative attributes of data instance per se, that is not a inherent characteristic of the instances modelled in a library parcel. For example, when there is a library of motors. The name of a person who entered the data for an instance of motor is not an inherent characteristic of the motor, but rather an administrative piece of information about the data record. Thus, such piece of information may be modelled in the object-meta class by the name of, say, creator. The content of this parcel will be further extended to enable tracing of the data source or “provenance”, in a future edition of this standard.

A parcel for UoM (Unit of Measurement) meta-class may be supplemented by a user, pro re nata. This means, the set of meta-properties of the UoM meta-class are not standardized in this part of IEC 62656, but the appellation of the meta-class and the reference structure to the UoM parcel are. As long as it defines an identifier for each unit of measurement and it provides more information about the unit of measurement than the current ISO 13584-IEC 61360 data model requires, any UoM meta-class specification may fit the purpose. Note that the information about the unit of measurement in the header section below “#UNIT” in this part of IEC 62656 is only informative, and any omission of this information shall not change the behaviour of the system. This is because a property ID eventually determines all the specifications about a property including the unit of measurement. When a UoM parcel is provided by a user, the parcel shall include all the information necessary to fill the attributes related to the unit of measurement of property described in the property meta-class.

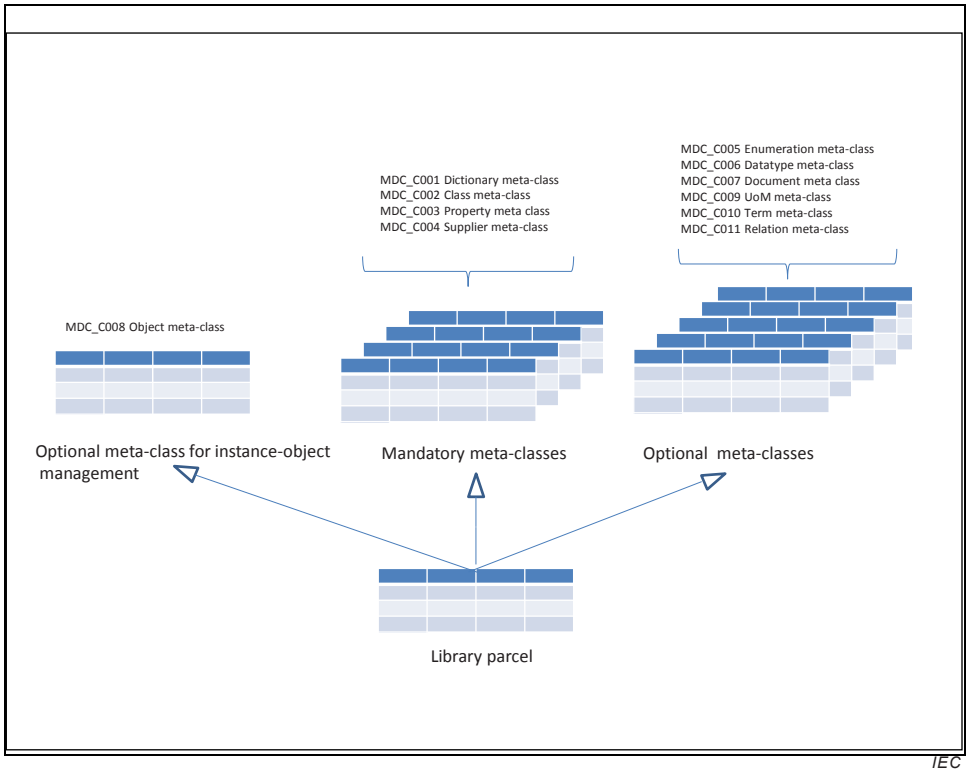


Figure 42 – Parcels for Domain Library and Domain Ontology (Dictionary)

6.2 Identification of conjunctive parcels

When it is necessary to identify conjunctive parcels, i.e., parcelling sheets that collectively describe a reference dictionary as instances, it shall be done through identifying the parcel ID of each parcel. If the values of the parcel IDs noted in the respective parcels are the same, then the parcels are regarded as conjunctive. A parcel ID shall be an identifier of STRING_TYPE, where no double quotation nor comma shall appear in the letter sequence of the identifier. However, in difference to BSU, hyphens (“-”) and colons (“:”) are allowed in the sequence, thus an ISO 8601 compliant time stamp such as “2006-06-25 08:19:49” may be used for the #PARCEL_ID.

6.3 Roles and definition of dictionary parcels

Roles and definitions of the dictionary parcels (at DO layer) are summarized in the following Table 3. Note that the ICIDs of the meta-classes are a normative part of the information defined by this part of IEC 62656. For the meta-classes and meta-properties defined in this standard, the RAI and version mechanisms also apply. They are assumed to have “/0112/2///IEC62656_1” as RAI, and they are assumed to start from version 1, in their initial release. If the RAIs and versions are missing from the meta-classes and properties, it should be understood that they are supplied as default values as above.

Table 3 – Meta-classes for building a domain-dictionary

(meta) Class ID	Preferred name in English	Definition
MDC_C001	Dictionary meta-class	meta-class being characterized by meta-properties that are necessary to identify and specify the information about the owner of the dictionary in a reference dictionary
MDC_C002	Class meta-class	meta-class being characterized by meta-properties that are necessary to identify and specify each class in a reference dictionary
MDC_C003	Property meta-class	meta-class being characterized by meta-properties that are necessary to identify and specify each property in a reference dictionary
MDC_C004	Supplier meta-class	meta-class being characterized by meta-properties that are necessary to identify and specify each information supplier in a reference dictionary
MDC_C005	Enumeration meta-class	meta-class being characterized by meta-properties that are necessary to identify and specify each option code used in an enumeration type property, including non_quantitative_code or non_quantitative_int type properties in a reference dictionary based on IEC 61360 or ISO 13584 series of standards,
MDC_C006	Datatype meta-class	meta-class being characterized by meta-properties that are necessary to identify and specify each named data-type in a reference dictionary
MDC_C007	Document meta-class	meta-class being characterized by meta-properties that are necessary to identify and specify each external document in a reference dictionary
MDC_C008	Object meta-class	meta-class being characterized by meta-properties that are necessary to identify and specify each row of instance, as a data object, in the data section of a parcel
MDC_C009	UoM meta-class	meta-class being characterized by meta-properties that are necessary to identify and specify each unit of measurement in a reference dictionary
MDC_C010	Term meta-class	meta-class being characterized by meta-properties that are necessary to identify and specify terms used in a header section of another parcel
MDC_C011	Relation meta-class	meta-class being characterized by meta-properties and is used to identify and specify relations and functions that span over several properties and/or classes

6.4 Properties of meta-dictionary (meta-ontology)

6.4.1 Overview of meta-classes

In the following 6.4.2 to 6.4.12, the structure of a class of meta-dictionary, namely, the header section of each meta-class is explained. The corresponding table in Annex E, lists the meta properties usable for defining the items in data section.

Note that what are defined in the data section of each meta-class are not the meta-properties, but data constructs that appear in an ontology modelling standard, such as IEC 61360-2 or

ISO 13584-42. Thus, for example, the class meta-class lists all the classes in a reference data dictionary, in its data section, while the property meta-class lists all the properties in the same dictionary.

6.4.2 Meta-properties for dictionary meta-class

Dictionary meta-class defines the following list of attributes, mainly originating in IEC 61360 or ISO 13584 as the meta-properties of the meta-class defining dictionaries contained in a conjunctive set of parcels, i.e., parcels used together:

– Dictionary Code;
– Code;
– Version number;
– Revision number.<lang>;
– Preferred name.<lang>;
– Synonymous name;
– Short name.<lang>;
– Name icon;
– Note.<lang>;
– Remark.<lang>;
– Supplier;
– LIIM source document identifier;
– LIIM status;
– LIIM name;
– LIIM date;
– LIIM application;
– LIIM level;
– Global language;
– Source language;
– Identifier encoding;
– Description.

The following attribute of a dictionary entity is not modelled in this parcel and is modelled as instances of the supplier parcel:

– Referred suppliers.

The following attribute of a dictionary entity is not modelled in this parcel and is modelled as instances of the class parcel:

– Contained classes.

The following attributes of ISO 13584-42 are not modelled in the spreadsheet structure, defined in this part of IEC 62656:

– Is complete;
– Updates;
– Update agreement;
– Referenced dictionaries;
– External file protocols;
– Base protocols;

– Supported vep;
– A posteriori semantic relationships.

Detailed specification of each meta-property of the parcel is summarized in IEC 61360-2 and in Parts 24, 25 and 42 of ISO 13584.

6.4.3 Meta-properties for class meta-class

Class parcel defines the following list of attributes of IEC 61360 or ISO 13584 as the (meta-) properties of a (meta-) class defining domain classes:

– Code;
– Version number;
– Revision number.<lang>;
– Date of original definition;
– Date of current version;
– Date of current revision;
– Preferred name.<lang>;
– Synonymous name;
– Short name.<lang>;
– Name icon;
– Definition.<lang>;
– Source document of definition;
– Note.<lang>;
– Remark.<lang>;
– Simplified drawing;
– Superclass;
– Class type;
– Supplier;
– Is case of;
– Applicable properties;
– Applicable types;
– Applicable documents;
– Description;
– Sub-class selection properties;
– Class value assignment;
– Imported properties;
– Imported types;
– Imported documents;
– Coded name;
– Property classification.

The following attributes of class are added in this parcel for advanced data modelling requirements:

– Requirement;
– Identification method for parcel;
– Alternate class ID;

– Applicable relations;
– Applicable terms;
– Segment.

The following attribute of a dictionary entity is not modelled in this parcel and is modelled as instances of the property parcel:

– Visible properties.

The following attribute of a dictionary entity is not modelled in this parcel and is modelled as instances of the data type parcel:

– Visible types.

Detailed specification of each meta-property of the parcel is summarized in Annex E and Annex G.

6.4.4 Meta-properties for property meta-class

Property parcel defines the following list of attributes of IEC 61360 or ISO 13584 as the meta-properties of a meta- class defining domain properties:

– Code;
– Version number;
– Revision number.<lang>;
– Date of original definition;
– Date of current version;
– Date of current revision;
– Preferred name.<lang>;
– Synonymous name;
– Short name.<lang>;
– Name icon;
– Definition.<lang>;
– Source document of definition;
– Note.<lang>;
– Remark.<lang>;
– Graphics;
– Property data element type;
– Definition class;
– Data type;
– Unit structure;
– Unit in text;
– Unit in SGML;
– Value format;
– Preferred letter symbol in text;
– Preferred letter symbol in SGML;
– Synonymous letter symbol;
– Formula in text;
– Formula in SGML;
– Condition;

– DET classification;
– Alternative units;
– Code for unit;
– Codes for alternative units;
– Property constraint.

Besides the attributes of properties defined in the common dictionary model, the following properties are added for the ease of data management:

– Alternate property ID;
– Super property;
– Description;
– Example;
– Quantity;
– Segment;
– Applicable relations;
– Applicable terms.

The following attribute remains in the standard just for archiving the legacy dictionaries. It is now obsolete and shall not be applied to a newly created dictionary or ontology.

– Property type classification.

Detailed specification of each meta-property of the parcel is summarized in Annex E and Annex G.

6.4.5 Meta-properties for supplier meta-class

Supplier parcel defines the following list of attributes of IEC 61360 or ISO 13584 as the (meta-) properties of a (meta-) class defining a supplier parcel:

– Supplier code;
– Revision number.<lang>;
– Date of original definition;
– Date of current version;
– Date of current revision;
– Organization id;
– Organization name;
– Organization description;
– Internal location;
– Street number;
– Street;
– Postal box;
– Town;
– Region;
– Postal code;
– Country;
– Facsimile number;
– Telephone number;
– E-mail;

– Telex number.

In addition to the attributes defined in IEC 61360 or ISO 13584, the following attributes are defined:

– Segment

Detailed specification of each meta-property of the parcel is summarized in Annex E and Annex G.

6.4.6 Meta-properties for enumeration meta-class

Enumeration parcel defines the following list of attributes of IEC 61360 or ISO 13584 as the (meta-)properties of a (meta-)class defining an enumeration parcel:

– Enumeration code list;
– Source document of value.

In addition to the attributes defined in IEC 61360 or ISO 13584, the following attributes are defined:

– Code;
– Enumerated list of terms;
– Version number;
– Revision number.<lang>;
– Date of original definition;
– Date of current version;
– Date of current revision;
– Preferred name.<lang>;
– Synonymous name;
– Short name.<lang>;
– Name icon;
– Definition.<lang>;
– Source document of definition;
– Note.<lang>;
– Remark.<lang>;
– Definition class;
– Description;
– Example;
– Number of selections;
– Type of list;
– Segment;
– Applicable terms.

The enumeration meta-class is mandatory, if an enumeration type is used in property meta-class.

Detailed specification of each meta-property of the parcel is summarized in Annex E and Annex G.

6.4.7 Meta-properties for data-type meta-class

Data type parcel defines the following list of attributes of IEC 61360 or ISO 13584 as the (meta-)properties of a (meta-)class defining a data-type parcel:

– Code;
– Version number;
– Revision number.<lang>;
– Date of original definition;
– Date of current version;
– Date of current revision;
– Preferred name.<lang>;
– Synonymous name;
– Short name.<lang>;
– Name icon;
– Definition class;
– Unit structure;
– Unit in text;
– Unit in SGML;
– Value format;
– Code for unit;
– Codes for alternative unit;
– Alternative_units;

In addition to the attributes defined in IEC 61360 or ISO 13584, the following attributes are defined:

– Description;
– Segment.

Detailed specification of each meta-property of the parcel is summarized in the Annex E and Annex G.

6.4.8 Meta-properties for document meta-class

Document parcel defines the following list of attributes of IEC 61360 or ISO 13584 as the (meta-)properties of a (meta-)class defining a document parcel:

– Code;
– Version number;
– Revision number.<lang>;
– Date of original definition;
– Date of current version;
– Date of current revision;
– Preferred name.<lang>;
– Synonymous name;
– Short name.<lang>;
– Name icon;
– Definition.<lang>;
– Remark.<lang>;

– Definition class;
– Document organization id;
– Document organization name;
– Document organization description;
– Remote location.<lang>;
– Character encoding.<lang>;
– Main content file.<lang>;
– Main content encoding.<lang>;
– Main content mime.<lang>;
– Main content exchange format.<lang>;
– Main content format RFC.<lang>;
– Main content http file name.<lang>.

In addition to the attributes defined in IEC 61360 or ISO 13584, the following attributes are defined:

– Segment;
– Applicable terms.

Detailed specification of each meta-property of the parcel is summarized in Annex E and Annex G.

6.4.9 Meta-properties for object meta-class

When the object meta-class is provided, the following set of (meta-)properties shall be present in the meta-class to appropriately model the attributes characterizing each row of data instance in the data sections of meta-dictionary, dictionary or library parcels.

– Data object identifier;
– Time stamp.

In addition to the attributes defined in IEC 61360 or ISO 13584, the following attributes are defined:

– Segment

Detailed specification of each meta-property of the parcel is summarized in Annex F.

6.4.10 Meta-properties for UoM meta-class

When a UoM meta-class is provided, at least the following set of (meta-)properties shall be present in the meta-class to appropriately model the attributes concerning the unit of measurement of an object property:

– Code;
– Version number;
– Revision number.<lang>;
– Date of original definition;
– Date of current version;
– Date of current revision;
– Preferred name.<lang>;
– Synonymous name;
– Short name.<lang>;

– Definition.<lang>;
– Source document of definition;
– Note.<lang>;
– Remark.<lang>;
– Definition class;
– Unit structure;
– Unit in text;
– Unit in SGML.

In addition to the attributes defined in IEC 61360 or ISO 13584, the following attributes are defined:

– Description;
– Segment;
– Applicable terms.

Detailed specification of each meta-property of the parcel is summarized in Annex F and Annex G.

6.4.11 Meta-properties for term meta-class

When a term meta-class is provided, the following set of (meta-) properties shall be present in the meta-class to appropriately model the specific terms that are used in a dictionary:

– Code;
– Version number;
– Revision number.<lang>;
– Date of original definition;
– Date of current version;
– Date of current revision;
– Preferred name.<lang>;
– Synonymous name;
– Short name.<lang>;
– Name icon;
– Definition.<lang>;
– Source document of definition;
– Note.<lang>;
– Remark.<lang>;
– Graphics;
– Definition class;
– Data type;
– Preferred letter symbol in text;
– Preferred letter symbol in SGML;
– Synonymous letter symbols;

In addition to the attributes defined in IEC 61360 or ISO 13584, the following attributes are defined:

– Description;
– Example;

– Quantity;
– Segment.

Detailed specification of each meta-property of the parcel is summarized in Annex F and Annex G.

The aim of this structure is not to design a generic terminology or vocabulary exchange format but to provide a means to define repeatedly used terms, constants or individuals used in a data dictionary.

- The term meta-class is mandatory, if the enumeration meta-class is present in conjunctive parcels.
- Terms resemble properties, however differ in that they do not have instance values, but the terms themselves appear as predefined instances in many places in a dictionary or in a document for a similar objective.

6.4.12 Meta-properties for relation meta-class

6.4.12.1 Relation, function, and predication

Relation meta-class is used to describe a named relation with a globally unique identifier. Relation is divided into two subtypes; one is called “predication”, and the other is called “function”. The difference between them is that the predication is supposed to have a domain but no codomain, whilst the function is supposed to have a domain and a codomain. When a predication is selected instead of a relation, in POM, this expressly states that the relation does not use a codomain. Note that when a relation R exists among $S_1, S_2, \cdots, S_n$ then R may be mathematically equated as a set R that is a subset of the Cartesian product of the n sets i.e. $R \subseteq S_1 \times S_2 \times \cdots \times S_n$.

Of course, from a mathematical point of view, a function as well as a predication is a kind of relation, and thus the basic characteristics of the relation, as explained above, shall be maintained in both the function and predication.

6.4.12.2 Meta properties for the relation

When a relation meta-class is provided, the following set of (meta-) properties shall be present in the meta-class to appropriately model the attributes of the relation meta-class:

– Code;
– Version number;
– Revision number.<lang>;
– Date of original definition;
– Date of current version;
– Date of current revision;
– Preferred name.<lang>;
– Synonymous name;
– Short name.<lang>;
– Name icon;
– Definition.<lang>;
– Source document of definition;
– Note.<lang>;
– Remark.<lang>;
– Graphics;

– Graphic Properties;
– Definition class;
– Letter symbol in text;
– Description;
– Example;
– Relation type;
– Domain of the relation;
– Domain of the function;
– Codomain of the function;
– Formula;
– Language for formula interpretation;
– External solver for the formula;
– Trigger event;
– Domain element type;
– Codomain element type;
– Super relation;
– Role of the relation.

In addition to the attributes defined in IEC 61360 or ISO 13584, the following attributes are defined:

– Segment.

Detailed specification of each meta-property of the parcel is summarized in Annex F and Annex G.

6.4.12.3 Role of the relation

A number of different relationships among ontological entities can be modelled as instances of relation using one and the same parcel, i.e., the relation meta-class. Macroscopically, they play 3 types of roles:

a) Constraint on one or several properties, as an extension of a single property constraint;
b) Grouping mechanism among ontological entities, in particular among heterogeneous ones;
c) Mapping, correspondence or transition among ontological entities.

These are different roles played by the relation, and each role is designated by an attribute named “role of the relation” (MDC_P210).

There are several predefined values for the “role of the relation” (MDC_P210), such as “arrow”, “constraint”, ”package” and “quantity”. Since the data type of this data type is currently a simple STRING_TYPE, user may add other values as options, provided such designations are effective for building a specialized interface with the relation.

The value “arrow” may be designated when a functional relation is required for representing a mapping from one category of collection of items to another. Thus it is a case c). In the Category Theory in mathematics, this type of mappings, often associated with a graphical presentation of an arrow, is a function to map one or several items in a category or categories to an item in another category, where a category is an arbitrary collection of items. Thus this type of functional relation is marked as an “arrow” in MDC_P210. Needless to say that , when this type of function is selected, simultaneously the attribute representing the “relation type” (MDC_P200) shall be set to “FUNCTION”. The source of mapping shall be specified in the “domain of the function” (MDC_P201) and the destination shall be designated in the “codomain of the function “(MDC_P203).

The value “package” shall be designated, when the items listed in the “domain of the function” (MDCP201) belong to the same organism, such as a “Package” used by an object-oriented computer language called “Ada”. This is also an equivalent of “Module” in computer languages known as “Modula-2” and “C++”. They might also appear as a technical basis in some other computer languages. In Part 3 of IEC 62656 series, this mechanism is used to model a package in UML referenced as a grouping mechanism of software artefacts in some Smart Grid related standards. In the case of IEC 62656-3, a hierarchy among packages needs to be modelled, therefore, the relation type (MDC_P200) is set to FUNCTION and the codomain (MDC_P203) of the function is set to a “parent package”, to which a number of subpackages belong.

The value “constraint” shall be designated when the relation is used to model a value constraint on a property or a set of properties. Specifically when it models a value constraint on a property in the manner of IEC 61360-2/ISO 13584-42, the attribute “role of the relation” (MDC_P210) shall be set to “constraint” to mean an IEC 61360-2/ISO 13584-42 compatible property constraint. Otherwise it shall be treated as a relational constraint that applies to a number of properties. As a special case of the relational constraint, it includes a unary constraint on a property, specified by a set of formulae in the “formula” attribute (MDC_P204). This attribute may be used in both the predication and function to specify the condition or intent of the relation. When the IEC 61360-2/ISO 13584-42 compatible constraint is selected, the formula of the relation requires to be set in a predefined manner. The specification is summarized in Table 4.

Table 4 – Formula specification for property constraint

Notation in the formula	Applied data type
enum_constraint(value1, value2,...) [a]	ENUMERATION
string_pattern_constraint(pattern) [b]	STRING_TYPE
string_range_constraint(min,max) [c]	STRING_TYPE
range_constraint(min, max, OPTIONAL min_inclusive, OPTIONAL max_inclusive, [d] OPTIONAL complement [e])	INT, REAL, RATIONAL
cardinality_constraint(bound1,bound2) [f]	Aggregate types
subclass_constraint(icid1,icid2,...) [g]	CLASS_REFERENCE, or CLASS_INSTANCE
subtype_constraint(type1,type2,...) [h]	ENTITY INSTANCE

[a] value1, value2: one of the elements of an enumeration

[b] pattern: a regular expression

[c] min, max: INTEGER

[d] min_inclusive, max_inclusive: BOOLEAN, its default value is TRUE.

[e] Complement: BOOLEAN, its default value is FALSE.

[f] Bound1, bound2: cardinality boundary

[g] icid1, icid2 ...: a list of ICIDs of classes to which the CLASS REFERENCE or CLASS_INSTANCE type property is allowed to reference.

[h] type1, type2,... are data types defined in ISO 10303-11

The value “quantity” must be designated when the relation is used to model a concept of quantity to which a set of the units of measurement belong. For example, the attribute “the

domain of the function" (MDC_P201) collects the IDs of the units of measurement (UoM) with a prefix, such as "millimetre", "micrometre", and "nanometre", etc., including a UoM "metre" itself, that are used to express the quantity "length measure", whilst the attribute "codomain of the function" (MDC_P203) contains only an ID of the coherent SI derived unit expressed in terms of the SI base units, i.e., that of the "metre". Note that "metre" appears in both the domain and the codomain of the function, in this case. In a rare case, a UoM that is specified in the codomain as the coherent SI derived unit is not considered appropriate for expressing a particular quantity. For example, the reciprocal second "1/s" is a due expression of the coherent SI derived unit for radioactive decay, but this is not used for expressing the particular quantity "radioactive decay" (See IEC/TS 62720), because the UoM "1/s" is used in many other quantities and the meaning becomes ambiguous. In this case, the ID for the reciprocal second shall not be listed in the domain of the function but only in the codomain of the function.

The functional relation as a quantity is also used to model a hierarchy among the quantities, such as a quantity for a generic length measure comprising the quantity for all length measures of SI units and the quantity for all Imperial length measures of Imperial Units, comprising mile, yard, foot, inch, etc. In this case, the ICID of the generic length measure may be designated as the "super relation"(MDC_P209) of both the SI length measure and the Imperial length measure. In the above case, the domain of the function (MDC_P202) and codomain of the function (MDC_P203) of the generic length measure may be kept open, for they assume respectively the union of respective attributes of the two specialized relations.

Since the scope of this part of IEC 62656 is not to standardize the use of POM for a specific purpose, the detailed specification of the use of relation for expressing a quantity or a system of units of measurement should be left to other parts or standards. However, a practical use case of the relation meta-class (parcel) for representing both the quantity and the system of units of measurement is illustrated in Annex M. See also IEC/TS 62720 for the code and the meaning of each unit and quantity.

6.4.12.4 Relation meta-class and Relational constraint

It is desirable that when a relational constraint is defined on some properties (or meta properties) in a class, and when these properties are used in a class of the layer down, the relational constraint marked as "#RELATION" should be indicated in the instruction column of the latter class and the related properties in the same row should be marked with FCOD(***icid***), FDOM(**icid**), or PRED (***icid***) in the respective cell, where an ***icid*** signifies an ICID of a defined relation in the relation meta-class as shown in 6.4.12.5. A function may be defined as a function calling some other functions, forming a recursive function. A property (likewise, a meta-property) is a unary function; thus it is a kind of functional relation. Thus in the domain of a function, there may be the IDs of several other functions and the IDs of several properties.

The constraint on a property, available in ISO 13584-42/IEC 61360-2 may be realised using a unary relation, specifying the constraint in the meta-property, named "Formula".

Figure 43 shows a simplified description of the relations among relation, function, and predication that may be defined in a relation meta-class. Each relation has a unique ID in the form of an ICID.

A functional relation (function) has both a codomain and a domain, while a predication has only a domain. When a property is used as a function, its ID (Property ID) shall be used for the Relation ID of the function. Consequently, the domain of a relation may contain the IDs of some functions and properties.

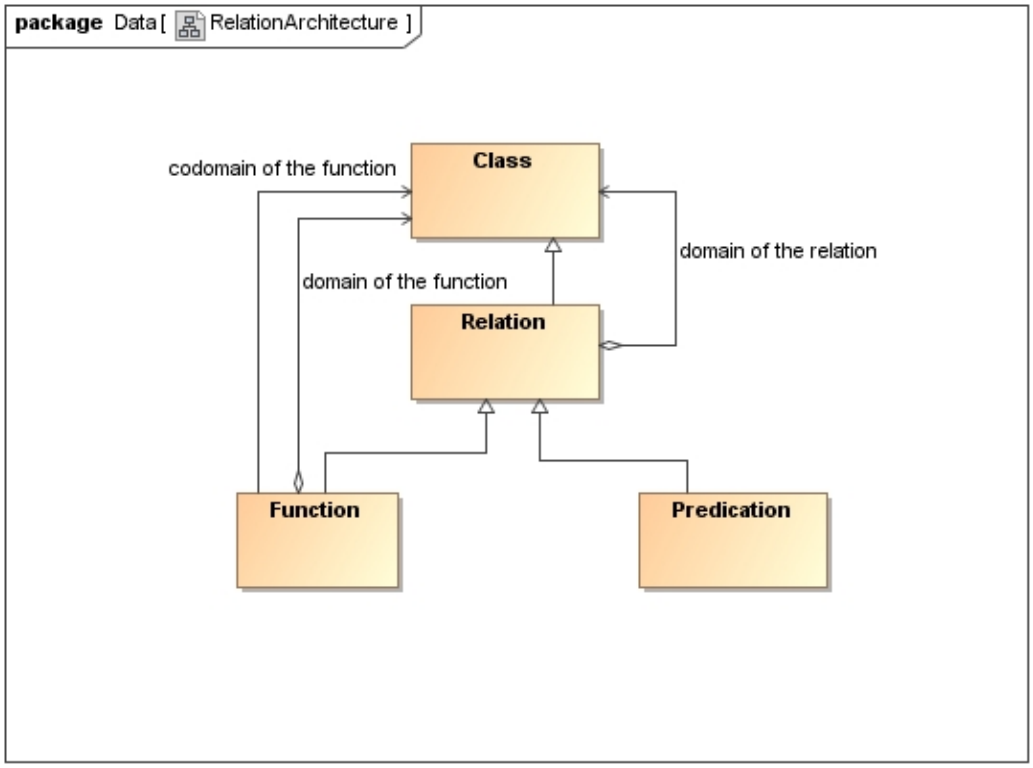


Figure 43 – Relation, function, and predication

6.4.12.5 Formula and External solver for the formula

Any formula can be used for the description, provided that the values of the meta-properties in a relation instance may be retrieved in a uniform manner and the formula can be resolved by an external solver: for a property appearing in a domain or codomain in an instance of the relation meta-class, having an icid for the CODE, shall be retrieved with FCOD[icid] or FDOM [icid] or DOM[icid] where icid means a code in ICID type. The value of other functional relation shall be accessed as FUNC[icid] or FUNCTION[icid]. The following Figure 44 shows a definition example of the relation meta-class.

#CLASS_ID:= MDC_C011				
#CLASS_NAME:= Relation meta-class				
#PROPERTY_ID	MDC_P001_13	MDC_P203	MDC_P202	MDC_P204
#PROPERTY_NAME	code	Codomain of the function	Domain of the function	Formula
#DATATYPE	ICID_STRING	ICID_STRING	LIST OF ICID_STRING	STRING_TYPE
	rel001	p001	(p002, p003)	FCOD[p001] = FDOM[p002] + FDOM[p003]

Figure 44 – Definition example of the Relation meta-class

7 Use of parcel for meta-ontology (MO) description

7.1 Overview of meta-meta-classes

In the following 7.2 to 7.5, the structure of a class of meta-meta-dictionary, namely, the header section of each meta-meta-class is explained. The corresponding table in Annex J

lists the meta-properties usable for defining the items in the data section of each meta-meta-class.

Note that what are defined in the data section of each meta-meta-class are not the meta-properties, but data constructs that appear in a meta-class. Thus, in the respective data section, for example, the class meta-meta-class lists all the types of meta-classes in a meta-dictionary, while the property meta-meta-class lists all the meta-meta properties used by the class meta-meta-class and other meta-meta-classes, that are consequently inherited into meta-classes.

7.2 Meta-properties for class meta-meta-class

When a class meta-meta-class is provided, the following set of (meta-meta-)properties shall be present in the meta-meta-class to appropriately model the class meta-class:

– Class ID;
– Property ID;
– Term ID;
– Relation ID;
– Preferred name.<lang>;
– Preferred name of the class;
– Definition.<lang>;
– Source document for definition;
– Note.<lang>;
– Data type;
– Description.<lang>;
– Version number;
– Revision number.<lang>;
– Applicable properties;
– Applicable relations;
– Applicable terms;
– Remark.<lang>;
– Requirement;
– Date of original definition;
– Date of current version.

Remind that this meta-meta class is used to designate the kinds and specifications of meta-classes used in a lower layer, i.e., at the level of meta-dictionary (M3-M2), using the attributes as meta-properties. Such meta-properties for characterizing the meta-meta-classes are defined in the property meta-meta-class.

7.3 Meta-properties for property meta-meta-class

When a property meta-meta-class is provided, the following set of (meta-)properties shall be present in the meta-meta-class to appropriately model the attributes concerning the property meta-meta-class used to describe meta-properties:

– Class ID;
– Property ID;
– Term ID;
– Preferred name.<lang>;

– Preferred name of the class.<lang>;
– Preferred name of the term.<lang>;
– Preferred name of the relation.<lang>;
– Definition.<lang>;
– Source document for definition;
– Note.<lang>;
– Data type;
– Definition class;
– Description.<lang>;
– Short name.<lang>;
– Version number;
– Revision number.<lang>;
– Remark.<lang>;
– Requirement;
– Date of original definition;
– Date of current version.

7.4 Meta-properties for term meta-meta-class

When a term meta-meta-class is provided, the following set of (meta-)properties shall be present in the meta-meta-class to appropriately model the attributes concerning the term meta-meta-class used to describe terms that have special meaning in the meta-meta-class:

– Class ID;
– Property ID;
– Term ID;
– Preferred name.<lang>;
– Preferred name of the class.<lang>;
– Definition.<lang>;
– Source document for definition;
– Note.<lang>;
– Data type;
– Definition class;
– Version number;
– Short name.<lang>;
– Revision number.<lang>;
– Remark.<lang>;
– Applicable relations;
– Requirement;
– Date of original definition;
– Date of current version.

The aim of this structure is not to design a terminology exchange format but to provide a means to define terms in Meta-Ontology or special terms effective among the structures of meta-classes (i.e., in their header sections) of Domain Ontology, not among the instances of the meta-classes in Domain Ontology. Thus, they are needed to be defined as instances of the term meta-meta-class in the Meta-Ontology.

- Terms resemble properties, however differ in that they do not have instance values, but the terms themselves appear as instances in many places in an ontology modelling standard.
- Typical example of such is a basic set of data types in a modelling system, such as Integer, Real, String, or Boolean. Although their details of implementation may vary among operating systems, computer languages, and applications, for the description of a basic modelling system, some data types must be considered as given (constant terms), and must be usable for the definition of the modelling system.

7.5 Meta-properties for relation meta-meta-class

When a relation meta-meta-class is provided, the following set of (meta-)properties shall be present in the meta-meta-class to appropriately model the attributes concerning the relation meta-meta-class used to describe relations:

– Class ID;
– Property ID;
– Term ID;
– Relation ID;
– Preferred name.<lang>;
– Preferred name of the class.<lang>;
– Definition.<lang>;
– Source document for definition;
– Note.<lang>;
– Data type;
– Description.<lang>;
– Short name.<lang>;
– Version number;
– Revision number.<lang>;
– Remark.<lang>.

Note that this meta-meta-class is used to describe special relations effective among the structures of meta-classes (i.e., in their header sections) of Domain Ontology, not among the instances of the meta-classes in Domain Ontology. Thus, they are needed to be defined as instances of this meta-meta-class in Meta-Ontology.

8 Mechanism for structural extension

8.1 General

The spreadsheet structure defined in this part of IEC 62656 has an intrinsic capability to add an extension to the standard data structure that is known as the common dictionary schema.

An implementer of this part of IEC 62656 may add a local instruction or an attribute field to property, starting with a sharp letter “#” provided it is not followed by a “#” and it does not conflict with the reserved keywords standardized in this part of IEC 62656.

Such additional keywords and the data entries corresponding to the keywords in the same line shall be interpreted as comments by other systems conforming to this part of IEC 62656. When a local extension is included, it shall be clearly marked in the conformance class identifier.

8.2 Example

When it is necessary to specify the name of a relational database into which a library data must be inserted, the user may extend the format of standard library parcelling sheet by adding a customized instruction such as “#TARGET_TBL” or “#CONTENT_ID” to specify the name of a target relational table, such as “#TARGET_TBL:= flash_memory001” or “#CONTENT_ID:= motor001”. Since any line starting with a “#” that does not conflict with keywords shall be understood as a comment line, other systems having the spreadsheet interface compliant with the specification of this part of IEC 62656 shall process the line as a comment and it shall not cause any problem to those systems. In this case, however, the conformance class for the spreadsheet shall be set to an appropriate value such as two (2), as given in Table 5, with the following instruction: “#PARCEL_CC:=2”, in order to allow the receiving system to check if any extension might conflict with its own local extension.

9 Conformance classes for parcelling spreadsheet

The spreadsheet interface defined in this part of IEC 62656 may define a spreadsheet structure used in an exchange of library data, dictionary data or Domain Ontology, Meta-Ontology, or Axiomatic Ontology. When it is used for the exchange of library data, the presumed conformance class of the library in ISO 13584-25 corresponds to the conformance class CC 11 with an extension that the class_extension entity is used for storing an ordered set of instances, instead of dic_class_instance.

When it is used for dictionary data exchange or definition, the expected conformance class for dictionary exchange in ISO 13584-25 is the conformance class (CC) 4. This corresponds to the conformance class (CC) 2 of this part of IEC 62656.

The level of conformance class in accordance with this part of IEC 62656 shall be specified as the parcel conformance class, noted after the keyword “#PARCEL_CC”, in the header section.

The spreadsheet structure defined in this part of IEC 62656 has the conformance classes as specified in Table 5, where CCID means the conformance class identifier.

Table 5 – Conformance classes

CCID	Definition	MoF layers
1	Data parcel just for DL (Domain Library)	M1-M0
1X	Data parcel only for DL with local extension of properties and possibly their instance values	M1-M0
2	Data parcel just for DO (Domain Ontology)	M2-M1
2X	Data parcel just for DO with local extension of properties and possibly their instance values	M2-M1
2A	Data parcels for all layers below comprising DO and DL	M2-M1, M1-M0
2AX	Data parcels for all layers below comprising DO and DL with local extension of properties and possibly their instance values	M2-M1, M1-M0
3	Data parcel just for MO (meta-ontology)	M3-M2
3X	Data parcel only for MO with local extension of properties and possibly their instance values	M3-M2
3A	Data parcel with all layers below, comprising MO, DO, and DL	M3-M2, M2-M1, M1-M0
3AO	Data parcel with all layers below, comprising MO, DO, and DL with local extension of properties and possibly their instance values	M3-M2, M2-M1, M1-M0
3B	Data parcels with the layer below comprising MO and DO	M3-M2, M2-M1
3BX	Data parcels with the layer below comprising MO and DO with local extension of properties and possibly their instance values	M3-M2, M2-M1
4	Data parcel just for AO (Axiomatic Ontology)	M4-M3
4X	Data parcel just for AO with local extension of properties and possibly their instance values	M4-M3
4A	Data parcels with all layers below, comprising AO, MO, DO, and DL	M4-M3, M3-M2, M2-M1, M1-M0
4AX	Data parcels with all layers below, comprising AO, MO, DO, and DL with local extension of properties and possibly their values	M4-M3, M3-M2, M2-M1, M1-M0
4B	Data parcels with the layer below comprising AO and MO	M4-M3, M3-M2
4BX	Data parcels with the layer below comprising AO and MO with local extension of properties and possibly their instance values	M4-M3, M3-M2
4C	Data parcels with all layers except DL comprising AO, MO and DO.	M4-M3, M3-M2, M2-M1
4CX	Data parcels with all layers except DL comprising AO, MO and DO with local extension of properties and possibly their instance values	M4-M3, M3-M2, M2-M1

Annex A
(normative)

Information object registration

In order to provide for unambiguous identification of an information object in an open system, the object identifier;

{iec standard 62656 part (1) version (1) }

is assigned to this part of IEC 62656. The meaning of this value is defined in ISO/IEC 8824-1.

Annex B
(normative)

Meta-dictionary file and updates

A physical file that contains the latest release of the meta-dictionary will be maintained by Subcommittee 3D and will be available on the IEC SC 3D dashboard on the IEC website.

The files may also contain associated human-readable documents that explain class and property definitions of the latest formal release of the dictionary.

The files will become available after this part of IEC 62656 becomes an IEC standard.

In the period between the formal publications of this part of IEC 62656, the latest version of the meta-dictionary in the parcel format will be available on the IEC website.

The files are intended to keep the meta-dictionary information always up to date during the standardization period of IEC 62656-1 or ISO 13584-35, which latter is a subset of the former. It may also include related documents or information for correcting errors and discrepancies found in IEC 62656-1 and ISO 13584-35 during and after the publication of this standard, until a next edition is formally issued, as a next edition or maybe in the form of a corrigendum or amendment.

Annex C
(normative)

Reserved words

The reserved words for parcels defined in this part of IEC 62656 are listed in the following Table C.1.

Table C.1 – Key words for instruction in class header *(1 of 3)*

Keywords	Category	Simplified description
#CLASS_ID	mandatory	Globally unique identifier of the class which is characterized by the properties described in the same parcel, and to which the instance data contained in the parcel belong
#CLASS_NAME.<lang>	optional – informative	Preferred name of the class specified by the class ID in the language optionally designated by <lang> information
#CLASS_DEFINITION.<lang>	optional – informative	Textual definition in the language specified by <lang> of the class designated by the class ID in the instruction column
#CLASS_NOTE.<lang>	optional – informative	Statement that provides additional information about the definition of the class that is essential for the understanding of the definition of the class specified by the class ID described in the same parcel
#ALTERNATE_CLASSID	optional – informative	Alternate class identifier of the class specified by the class ID noted in the same parcel
#SUPER_ALT_CLASSID	optional – informative	Alternate class identifier of the class specified by the class ID in the same parcel, where the extent of instances of the class indicated by this alternate ID includes the extent of instances of the class indicated by the class ID
#SUB_ALT_CLASSID	optional – informative	Alternate class identifier of the class specified by the class ID in the same parcel, where the extent of instances of the class identified by this alternate ID is included in the extent of instances of the class identified by the class ID
#SOURCE_LANGUAGE	optional – informative	Information about the SOURCE_LANGUAGE specified in the dictionary. The information is provided by an IEC 62656 or ISO 13584-35 compliant system, and any change of the information by user shall not affect the behaviour of the system.
#PARCEL_MODE	optional – functional	Designation of the use mode of the parcel indicating if the parcel is intended for defining a full content of an ontology or a library or just part of it for updating the content or for test purpose
#PARCEL_ID	optional – functional	Designation of the conjunctive parcels, i.e., the parcels which describe a dictionary in part in the same unit of exchange, where the PARCEL_ID may include neither comma nor double quotation mark. Conjunctive parcels are required to have the same alphanumeric letter sequence for the identifier. When this ID is omitted for dictionary parcelling, the other parcels processed together shall be construed as conjunctive parcels.
#PARCEL_CC	optional – functional	Designation of the conformance class, defined in the IEC 62656 or ISO 13584-35 structure, with respect to the data contained in the parcel sheet. When there is a local extension, the PARCEL_CC must be clearly marked for such the extension.

Table C.1 *(2 of 3)*

Keywords	Category	Simplified description
#DEFAULT_SUPPLIER	optional – functional	Prefix to be added to the shorthand notations of class ID and property ID , with an aim to make a full identifier sequence
#DEFAULT_VERSION	optional – functional	Postfix to be appended to the shorthand notations of the class ID and property ID in the header section to denote the version of them, with an aim to make a full identifier sequence
#DEFAULT_DATA_SUPPLIER	optional – functional	Prefix to be added to the shorthand notations of ICID in the data section, with an aim to make a full identifier sequence
#DEFAULT_DATA_VERSION	optional – functional	Postfix to be appended to the shorthand notations of class ID and property ID in the data section to denote the version of the items, with an aim to yield a full identifier sequence
# OBJECT_ID_NAME	optional – informative	name of the data object identification system, by which each row of instance in the data section shall be uniquely and globally identified
#PROPERTY_ID	mandatory	Globally unique identifier of a property based on ISO/IEC 6523 referenced for the definition of instances in the data section of the same parcel
#PROPERTY_NAME.<lang>	optional – informative	Preferred name of the property specified by the property ID in the language optionally designated by the <lang> information
#DEFINITION.<lang>	optional – informative	This line provides information about the definition of the properties specified by the property IDs listed in the #PROPERTY_ID line. The information is provided by an IEC 62656 or ISO 13584-35 compliant system, and any change of the information by user shall not affect the behaviour of the system. If necessary, the language for the definition may be specified using ISO 639-1, by supplementing the <lang> field.
#NOTE.<lang>	optional –informative	Statement that provides additional information about the definition that is essential for the understanding of the latter
#DATATYPE	optional – informative	This line provides information about the data type of the property specified by the property ID in the #PROPERTY_ID line. The information is provided by a parcel tool, and any change of the information by user shall not affect the behaviour of the system.
#UNIT	optional – informative	Information about the unit of measurement of the property specified by the property ID.
#REQUIREMENT	optional – functional	The reserved word “key” in the line designates the key properties corresponding to the property IDs in the #PROPERTY_ID line.
#ALTERNATIVE_UNITS	optional – informative	Information about other units of measurement that may be used for the property specified by the property ID
#VARIABLE_PREFIX_UNIT	optional – informative	Unit whose prefix other than default one can be selected
#SUPER_PROPERTY	optional – informative	Identifier of the super property of which this property is a specialization

Table C.1 *(3 of 3)*

Keywords	Category	Simplified description
#ALTERNATE_ID	optional – informative	Generally the values of a property designated by its alternate property ID are assignable to the property designated by the property ID, and vice versa.
#SUPER_ALTERNATE_ID	optional – informative	Alternate property identifier(s) of the property specified by the property ID, where all the values of a property designated by the property ID are assignable to the super property designated by the alternate ID of super-property, and probably some of the values of the super property may be assignable to the property designated by the property ID
#SUB_ALTERNATE_ID	optional – informative	Alternate property identifier(s) of the property specified by the property ID, where all the values of the sub-property designated by the alternate ID of sub-property are assignable to the property designated by the property ID, and probably some of the values of the property may be assignable to some sub-property designated by the alternate ID of sub-property
#EQUIVALENT_ID	optional – informative	Alternate property identifier(s) of the property specified by the property ID, where all the values of the sub-property designated by the sub-alternate ID are assignable to the property designated by the property ID, and some of the values of the property may be assignable to some sub-property designated by the sub-alternate ID of property
#UNIT_ID	optional – informative	Identifier to uniquely reference the unit of measurement (UoM) used in a property, being specified by a UNIT_ID listed in the #UNIT_ID line, noted in the same column as the property ID in a parcel.
#VALUE_FORMAT	optional – informative	Value format of the property value specified by the property ID.
#ID_ENCODE	optional – functional	Specification of global encoding method of identifiers
#DELIMITER	optional – informative	Specification of the character used for marking the boundary of neighbouring cells in a current file
#DECIMAL	optional – informative	Character used to separate the integer part and the fractional part of a number expressed in decimal form
#PATTERN	optional – informative	String pattern that the string type value of a property shall follow. The keyword corresponds to the pattern constraint for property defined in ISO 13584-42:2010 that is duplicated in IEC 61360-2:2012.
#RELATION	optional – informative	Constraint that works as a predicate or functional relationship among several properties and/or classes

Annex D
(normative)

Description examples of data types

Table D.1 – Description examples for simple data types

Description in ISO 13584-25, -42	Description in IEC 62656
STRING_TYPE	STRING_TYPE
DATE_DATA_TYPE	DATE_TYPE
TIME_DATA_TYPE	TIME_TYPE
DATE_TIME_DATA_TYPE	DATE_TIME_TYPE
N/A	IRDI_STRING_TYPE
N/A	ICID_STRING_TYPE
N/A	ISO_29002_IRDI
N/A	URI_TYPE
TRANSLATABLE_STRING_TYPE	TRANSLATABLE_STRING_TYPE
NON_TRANSLATABLE_STRING_TYPE	NON_TRANSLATABLE_STRING_TYPE
BOOLEAN_TYPE	BOOLEAN_TYPE
NUMBER_TYPE	NUMBER_TYPE
INT_TYPE	INT_TYPE INTEGER
INT_MEASURE_TYPE	INT_MEASURE_TYPE
INT_CURRENCY_TYPE	INT_CURRENCY_TYPE
REAL_TYPE	REAL_TYPE
REAL_MEASURE_TYPE	REAL_MEASURE_TYPE
REAL_CURRENCY_TYPE	REAL_CURRENCY_TYPE
NON_QUANTITATIVE_CODE_TYPE	ENUM_CODE_TYPE(*enum_id*), *or* ENUM_CODE_TYPE(*enum_id*(*code1*, *code2*, …))
NON_QUANTITATIVE_INT_TYPE	ENUM_INT_TYPE (*enum_id*), *or* ENUM_INT_TYPE (*enum_id*(*code1*, *code2*, …))
N/A	ENUM_REAL (*enum_id*), *or* ENUM_REAL(*enum_id*(*code1*, *code2*, …))
N/A	ENUM STRING(*enum_id*), *or* ENUM_STRING(*enum_id*(*code1*, *code2*, …))

NOTE 1 In each designation of type for IEC 62656 where "_TYPE" is found, "_TYPE" can be omitted.

NOTE 2 IRDI_STRING is a subtype of STRING_TYPE and is designed as ISO/IEC 11179 conformant global identifier sequence. ICID_STRING is a subtype of IRDI_STRING, where the delimiter between RAI and DI is "#" while the delimiter between DI and VI is confined to "##".

NOTE 3 ENUM_REAL_TYPE is not available in ISO 13584-42/IEC 61360, instead a constraint mechanism named "enumeration constraint" may be used for the purpose. However the specification of the enumeration constraint is much more complex than enum_real, for the former needs to define first the data type, and then constrain the selectable subset, with an implicit understanding that only one of them can be selected. The POM recommends instead the use of enum_real type, for it can be defined in one step, and it is analogous to the style of definition of enum_int_type.

NOTE 4 For the specification of the data types, any postfix word, "_type", such as in ENUM_CODE_TYPE or REAL_TYPE can be omitted and simplified as ENUM_CODE or REAL, respectively.

NOTE 5 "N/A" means exactly corresponding data type is not available in IEC 61360-2, nor in ISO 13584-42 or ISO 13584-25.

Table D.2 – Description examples for complex data types

Description in ISO 13584-25, -42	Description in IEC 62656
N/A	ENUM_RATIONAL(*enum_id*)*, or* ENUM_RATIONAL (*enum_id*(*code1*, *code2*, …))
CLASS_REFERENCE_TYPE (CLASS_INSTANCE_TYPE)	CLASS_REFERENCE_TYPE(*icid*),or CLASS_INSTANCE_TYPE(*icid*),
N/A	ENUM_REFERENCE(*enum_id*)*, or* ENUM_INSTANCE(enum_id),or ENUM_REFERENCE(*enum_id*(*code1*, *code2*, …)), or ENUM_INSTANCE(*enum_id*(*code1*, *code2*, …))
N/A	ENUM_BOOLEAN(enum_id) or ENUM_BOOLEAN((enum_id(code1, code2))
LEVEL_TYPE	LEVEL_TYPE(MIN,NOM,TYP,MAX)
SET_TYPE	SET(*b1*, *b2*)
LIST_TYPE – uniqueness TRUE	UNIQUE_LIST(*b1*, *b2*)
LIST_TYPE – uniqueness FALSE	LIST(*b1*, *b2*)
ARRAY_TYPE – uniqueness TRUE – optional TRUE	UNIQUE_OPTIONAL_ARRAY(*b1*, *b2*)
ARRAY_TYPE – uniqueness TRUE – optional FALSE	UNIQUE_ARRAY(*b1*, *b2*)
ARRAY_TYPE – uniqueness FALSE – optional TRUE	OPTIONAL_ARRAY(*b1*, *b2*)
ARRAY_TYPE – uniqueness FALSE – optional FALSE	ARRAY(*b1*, *b2*)
BAG_TYPE	BAG(*b1*, *b2*)
SET_WITH_SUBSET_CONSTRAINT_VALUE	CONSTRAINED_SET(*b1*, *b2*, *cmn*, *cmx*)
NAMED_TYPE	NAMED_TYPE(*type_id*)
PLACEMENT_TYPE	PLACEMENT_2D
	PLACEMENT_3D
AXIS1_PLACEMENT_TYPE	AXIS1_PLACEMENT_2D(ref_id),
	AXIS1_PLACEMENT_3D(ref_id),
AXIS2_PLACEMENT_2D_TYPE	AXIS2_PLACEMENT_2D(ref_id)
AXIS2_PLACEMENT_3D_TYPE	AXIS2_PLACEMENT_3D(ref1_id, ref2_ id)
ENTITY_INSTANCE_TYPE	ENTITY_INSTANCE (type_id)
NOTE 1 In each designation of type for IEC 62656 where "_TYPE" is found, "_TYPE" can be omitted. NOTE 2 b1 is the value of attribute "bound_1" and b2 is the value of attribute "bound_2" of aggregate_type. NOTE 3 cmn is the value of attribute "cardinal_min" and cmx is the value of attribute "cardinal_max". NOTE 4 enum_id is a global ID for the enumeration list. NOTE 5 code1, code2, are value_codes of dic_value. They can appear in header section as informative purpose. NOTE 6 If the data type is a aggregate type, such as LIST, SET, BAG, or ARRAY, then it connects to a simple type by using the keyword "OF". NOTE 7 The "icid" in CLASS_REFERENCE_TYPE can omit VI (version identifier), in this case the class having the specified ID with any version can be referenced. NOTE 8 AXIS2_PLACEMENT_2D takes one reference direction and interprets it as the direction of local X-axis.	

For the definition of AXIS1_PLACEMENT, specification of the dimensionality of the embeddding space, i.e, either 2D(2-dimensional) or 3D(3-dimensional), is necessary, because both a sphere and a circle may be drawn with just a specification of the radius value on one axis. However, the meaning of a sphere defined on a 2D radius is dubious. In this regard, in the case of geometric modelling systems, the dimensionality of the embedding space (modelling space) is obvious from other contexts, for such a parameter is normally set at the

outset of a session, or fixed unique by the modelling system. However, in a library of geometries, such as a library of primitive shapes in IEC 61360 CDD, the context shall be specified within the library. To avoid any confusion, 2D and 3D cases are cleary differentiated in the POM.

For the use of ENTITY_INSTANCE_TYPE, to associate an ICID to the entity to be instantiated, the data type shall be defined in the data type meta-class with its explicit name, and a list of attributes.

Annex E
(normative)

Meta-properties used by normative meta-classes

The following tables summarize the use of meta-properties in normative meta-classes, of which the meta-classes for dictionary, supplier, class, and property are mandatory. Namely, they all shall exist when a dictionary is exchanged based on the dictionary parcel format. The definitions themselves of meta-properties and meta-classes are accommodated in Annex G. For an updated and complete listing of meta-properties in a form of parcelling sheet, it is recommended that the readers refer to Annex B.

Most of the properties of meta-classes are expressed using STRING type. In most of the cases, the length of the codes are not specified in this standard, but if the content is brought in from a specific standard, the properties shall follow the specification stipulated in a respective standard. For example, IEC 61360-1 recommends the use of 6-character identifiers for class code and property code, whilst such is not a syntactical requirement in IEC 61360-ISO 13584 Common Dictionary Model.

In case that the language used for the representation of a property needs to be specified using a “<lang>” extension, the extension shall be substituted by a two-letter country code defined by ISO 639-1, possibly being followed by a two-letter country code based on ISO 3166-1. The latter code is used to specify a regional variant of the same language.

Table E.1 – Meta-properties used by dictionary meta-class *(1 of 2)*

MMDC_P001	MMDC_P102	MMDC_P004_1.EN	MMDC_P004_1.FR	MMDC_P013	MMDC_P014	MMDC_P014.FR
Property ID	**Requirement**	**Preferred name in English**	**Preferred name in French**	**Version number**	**Revision number**	**Revision number**
Identificateur de propriété	**Exigence**	**Nom préférentiel en anglais**	**Nom préférentiel en français**	**Numéro de version**	**Numéro de révision**	**Numéro de révision**
MDC_P001_1	KEY	Dictionary code	Code de dictionnaire	001	001	001
MDC_P001_14	MAND	Code	Code	001	001	001
MDC_P002_1	KEY	Version number	Numéro de version	001	001	001
MDC_P002_2.<lang>	MAND	Revision number	Numéro de révision	001	001	001
MDC_P004_1.<lang>	MAND	Preferred name	Nom préférentiel	001	001	001
MDC_P004_2	OPT	Synonymous name	Nom synonyme	001	001	001
MDC_P004_3.<>	OPT	Short name	Nom abrégé	001	001	001
MDC_P004_4	OPT	Name icon	Icône de nom	001	001	001
MDC_P007_1.<lang>	OPT	Note	Note	001	001	001
MDC_P007_2.<lang>	OPT	Remark	Remarque	001	001	001
MDC_P012	KEY	Supplier	Fournisseur	001	001	001

Table E.1 *(2 of 2)*

MMDC_P001	MMDC_P102	MMDC_P004_1.EN	MMDC_P004_1.FR	MMDC_P013	MMDC_P014	MMDC_P014.FR
Property ID	**Requirement**	**Preferred name in English**	**Preferred name in French**	**Version number**	**Revision number**	**Revision number**
Identificateur de propriété	**Exigence**	**Nom préférentiel en anglais**	**Nom préférentiel en français**	**Numéro de version**	**Numéro de révision**	**Numéro de révision**
MDC_P071	OPT	LIIM source document identifier	Identificateur LIIM du document source	001	001	001
MDC_P072	MAND	LIIM status	État LIIM	001	001	001
MDC_P073	MAND	LIIM name	Nom LIIM	001	001	001
MDC_P074	MAND	LIIM date	Date LIIM	001	001	001
MDC_P075	OPT	LIIM application	Application LIIM	001	001	001
MDC_P076	MAND	LIIM level	Niveau LIIM	001	001	001
MDC_P080	OPT	Global language	Langue globale	001	001	001
MDC_P081	MAND	Source language	Langue source	001	001	001
MDC_P082	OPT	Identifier encoding	Codage des identificateurs	001	001	001
MDC_P112	OPT	Description	Description	001	001	001

Table E.2 – Meta-properties used by class meta-class *(1 of 4)*

MMDC_P001	MMDC_P102	MMDC_P004_1.EN	MMDC_P004_1.FR	MMDC_P013	MMDC_P014	MMDC_P014.FR
Property ID	**Requirement**	**Preferred name in English**	**Preferred name in French**	**Version number**	**Revision number**	**Revision number**
Identificateur de propriété	**Exigence**	**Nom préférentiel en anglais**	**Nom préférentiel en français**	**Numéro de version**	**Numéro de révision**	**Numéro de révision**
MDC_P001_5	KEY	Code	Code	001	001	001
MDC_P002_1	KEY	Version number	Numéro de version	001	001	001
MDC_P002_2. <lang>	MAND	Revision number	Numéro de révision	001	001	001
MDC_P003_1	MAND	Date of original definition	Date de la définition originale	001	001	001
MDC_P003_2	MAND	Date of current version	Date de la version actuelle	001	001	001
MDC_P003_3. <lang>	OPT	Date of current revision	Date de la révision actuelle	001	001	001
MDC_P004_1. <lang>	MAND	Preferred name	Nom préférentiel	001	001	001
MDC_P004_2	OPT	Synonymous name	Nom synonyme	001	001	001
MDC_P004_3. <lang>	OPT	Short name	Nom abrégé	001	001	001

Table E.2 *(2 of 4)*

MMDC_P001	MMDC_P102	MMDC_P004_1.EN	MMDC_P004_1.FR	MMDC_P013	MMDC_P014	MMDC_P014.FR
Property ID	**Requirement**	**Preferred name in English**	**Preferred name in French**	**Version number**	**Revision number**	**Revision number**
Identificateur de propriété	**Exigence**	**Nom préférentiel en anglais**	**Nom préférentiel en français**	**Numéro de version**	**Numéro de révision**	**Numéro de révision**
MDC_P004_4	OPT	Name icon	Icône de nom	001	001	001
MDC_P005.<lang>	MAND	Definition	Définition	001	001	001
MDC_P006_1	OPT	Source document of definition	Document source de définition	001	001	001
MDC_P007_1.<lang>	OPT	Note	Note	001	001	001
MDC_P007_2.<lang>	OPT	Remark	Remarque	001	001	001
MDC_P008_1	OPT	Simplified drawing	Dessin simplifié	001	001	001
MDC_P010	OPT [a]	Superclass	Superclasse	001	001	001

[a] If it does not exist, the class has no superclass.

Table E.2 *(3 of 4)*

MMDC_P001	MMDC_P102	MMDC_P004_1.EN	MMDC_P004_1.FR	MMDC_P013	MMDC_P014	MMDC_P014.FR
Property ID	**Requirement**	**Preferred name in English**	**Preferred name in French**	**Version number**	**Revision number**	**Revision number**
Identificateur de propriété	**Exigence**	**Nom préférentiel en anglais**	**Nom préférentiel en français**	**Numéro de version**	**Numéro de révision**	**Numéro de révision**
MDC_P011	MAND	Class type	type de classe	001	001	001
MDC_P012	KEY	Supplier	Fournisseur	001	001	001
MDC_P013	OPT[a]	Is case of	Est en cas de	001	001	001
MDC_P014	OPT	Applicable properties	Propriétés applicables	001	001	001
MDC_P015	OPT	Applicable types	Types applicables	001	001	001
MDC_P016	OPT	Sub-class selection properties	Propriétés de sélection de sous-classes	001	001	001
MDC_P094	OPT	Applicable documents	Documents applicables	001	001	001
MDC_P103	OPT	Alternate class ID	Identificateur alternatif de classe	001	001	001
MDC_P112	OPT	Description	Description	001	001	001

[a] Mandatory if “MDC_P011 Class type” is XXX_CASE_OF

Table E.2 *(4 of 4)*

MMDC_P001	MMDC_P102	MMDC_P004_1.EN	MMDC_P004_1.FR	MMDC_P013	MMDC_P014	MMDC_P014.FR
Property ID	**Requirement**	**Preferred name in English**	**Preferred name in French**	**Version number**	**Revision number**	**Revision number**
Identificateur de propriété	**Exigence**	**Nom préférentiel en anglais**	**Nom préférentiel en français**	**Numéro de version**	**Numéro de révision**	**Numéro de révision**
MDC_P017	OPT	Class value assignment	Affectation de valeurs de classe	001	001	001
MDC_P090	OPT	Imported properties	Propriétés importées	001	001	001
MDC_P091	OPT	Imported types	Types importés	001	001	001
MDC_P093	OPT	Imported documents	Documents importés	001	001	001
MDC_P018	OPT	Coded name	Nom codé	001	001	001
MDC_P096	OPT	Property classification	Classification de propriété	001	001	001
MDC_P097	MAND	Requirement	Exigence des propriétés	001	001	001
MDC_P098	OPT	Identification method for parcel	Méthode d'identification pour un paquet	001	001	001
MDC_P211	OPT	Segment	Segment	001	001	001
MDC_P230	OPT	Applicable relations	Relations applicables	001	001	001
MDC_P231	OPT	Applicable terms	Termes applicables	001	001	001

Table E.3 – Meta-properties used by property meta-class *(1 of 4)*

MMDC_P001	MMDC_P102	MMDC_P004_1.EN	MMDC_P004_1.FR	MMDC_P013	MMDC_P014	MMDC_P014.FR
Property ID	**Requirement**	**Preferred name in English**	**Preferred name in French**	**Version number**	**Revision number**	**Revision number**
Identificateur de propriété	**Exigence**	**Nom préférentiel en anglais**	**Nom préférentiel en français**	**Numéro de version**	**Numéro de révision**	**Numéro de révision**
MDC_P001_6	KEY	Code	Code	001	001	001
MDC_P002_1	KEY	Version number	Numéro de version	001	001	001
MDC_P002_2.<lang>	MAND	Revision number	Numéro de révision	001	001	001
MDC_P003_1	MAND	Date of original definition	Date de la définition originale	001	001	001
MDC_P003_2	MAND	Date of current version	Date de la version actuelle	001	001	001
MDC_P003_3.<lang>	OPT	Date of current revision	Date de la révision actuelle	001	001	001
MDC_P004_1.<lang>	MAND	Preferred name	Nom préférentiel	001	001	001
MDC_P004_2	OPT	Synonymous name	Nom synonyme	001	001	001

Table E.3 *(2 of 4)*

MMDC_P001	MMDC_P102	MMDC_P004_1.EN	MMDC_P004_1.FR	MMDC_P013	MMDC_P014	MMDC_P014.FR
Property ID	**Requirement**	**Preferred name in English**	**Preferred name in French**	**Version number**	**Revision number**	**Revision number**
Identificateur de propriété	**Exigence**	**Nom préférentiel en anglais**	**Nom préférentiel en français**	**Numéro de version**	**Numéro de révision**	**Numéro de révision**
MDC_P004_3.<lang>	OPT	Short name	Nom abrégé	001	001	001
MDC_P004_4	OPT	Name icon	Icône de nom	001	001	001
MDC_P005.<lang>	MAND	Definition	Définition	001	001	001
MDC_P006_1	OPT	Source document of definition	Document source de définition	001	001	001
MDC_P007_1.<lang>	OPT	Note	Note	001	001	001
MDC_P007_2.<lang>	OPT	Remark	Remarque	001	001	001
MDC_P008_2	OPT	Graphics	Graphisme	001	001	001

Table E.3 *(3 of 4)*

MMDC_P001	MMDC_P102	MMDC_P004_1.EN	MMDC_P004_1.FR	MMDC_P013	MMDC_P014	MMDC_P014.FR
Property ID	**Requirement**	**Preferred name in English**	**Preferred name in French**	**Version number**	**Revision number**	**Revision number**
Identificateur de propriété	**Exigence**	**Nom préférentiel en anglais**	**Nom préférentiel en français**	**Numéro de version**	**Numéro de révision**	**Numéro de révision**
MDC_P020	MAND	Property data element type	Type d'élément de données de propriétés	001	001	001
MDC_P021	MAND	Definition class	Classe de définition	001	001	001
MDC_P022	MAND	Data type	Type de données	001	001	001
MDC_P023	OPT [a]	Unit structure	Structure d'unité	001	001	001
MDC_P023_1	OPT [b]	Unit in text	Unité en texte	001	001	001
MDC_P023_2	OPT	Unit in SGML	Unité en SGML	001	001	001
MDC_P024	OPT	Value format	Format de valeur	001	001	001
MDC_P025_1	OPT [c]	Preferred letter symbol in text	Symbole littéral préférentiel en texte	001	001	001
MDC_P025_2	OPT	Preferred letter symbol in SGML	Symbole littéral préférentiel en SGML	001	001	001
MDC_P025_3	OPT	Synonymous letter symbols	Symboles littéraux synonymes	001	001	001

[a] Mandatory for quantitative data.

[b] Mandatory if "MDC_P023_2 Unit in SGML" has a value.

[c] Mandatory if "MDC_P025_2 Preferred letter symbol in SGML" has a value.

Table E.3 *(4 of 4)*

MMDC_P001	MMDC_P102	MMDC_P004_1.EN	MMDC_P004_1.FR	MMDC_P013	MMDC_P014	MMDC_P014.FR
Property ID	**Requirement**	**Preferred name in English**	**Preferred name in French**	**Version number**	**Revision number**	**Revision number**
Identificateur de propriété	**Exigence**	**Nom préférentiel en anglais**	**Nom préférentiel en français**	**Numéro de version**	**Numéro de révision**	**Numéro de révision**
MDC_P027_1	OPT [d]	Formula in text	Formule en texte	001	001	001
MDC_P027_2	OPT	Formula in SGML	Formule en SGML	001	001	001
MDC_P028	OPT [e]	Condition	Condition	001	001	001
MDC_P040	OPT	DET classification	DET classification	001	001	001
MDC_P041	OPT	Code for unit	Code pour l'unité	001	001	001
MDC_P042	OPT	Codes for alternative units	Codes pour les unités alternatives	001	001	001
MDC_P068	OPT	Property constraint	Contrainte de propriété	001	001	001
MDC_P095	OBS	Property type classification	Classification du type de propriété	001	001	001
MDC_P101	OPT	Alternate property ID	Identificateur alternatif de propriété	001	001	001
MDC_P110	OPT	Super property	Super propriété	001	001	001
MDC_P111	OPT	Alternative units	Unités alternatives	001	001	001
MDC_P112	OPT	Description	Description	001	001	001
MDC_P113	OPT	Example	Exemple	001	001	001
MDC_P114	OPT	Quantity	Quantité	001	001	001
MDC_P211	OPT	Segment	Segment	001	001	001
MDC_P230	OPT	Applicable relations	Relations applicables	001	001	001
MDC_P231	OPT	Applicable terms	Termes applicables	001	001	001

[d] Mandatory if "MDC_P027_2 Formula in SGML" has a value.

[e] Mandatory for context dependent characteristics.

Table E.4 – Meta-properties used by supplier meta-class *(1 of 2)*

MMDC_P001	MMDC_P102	MMDC_P004_1.EN	MMDC_P004_1.FR	MMDC_P013	MMDC_P014	MMDC_P014.FR
Property ID	**Requirement**	**Preferred name in English**	**Preferred name in French**	**Version number**	**Revision number**	**Revision number**
Identificateur de propriété	**Exigence**	**Nom préférentiel en anglais**	**Nom préférentiel en français**	**Numéro de version**	**Numéro de révision**	**Numéro de révision**
MDC_P001_2	KEY	Supplier code	Code du fournisseur	001	001	001
MDC_P002_2.<lang>	MAND	Revision number	Numéro de révision	001	001	001
MDC_P003_1	MAND	Date of original definition	Date de la définition originale	001	001	001
MDC_P003_2	MAND	Date of current version	Date de la version actuelle	001	001	001
MDC_P003_3.<lang>	OPT	Date of current revision	Date de la révision actuelle	001	001	001
MDC_P050_1	OPT	Organization id	Identificateur d'organisation	001	001	001
MDC_P050_2	MAND	Organization name	Nom d'organisation	001	001	001
MDC_P050_3	OPT	Organization description	Description d'organisation	001	001	001

Table E.4 *(2 of 2)*

MMDC_P001	MMDC_P102	MMDC_P004_1.EN	MMDC_P004_1.FR	MMDC_P013	MMDC_P014	MMDC_P014.FR
Property ID	**Requirement**	**Preferred name in English**	**Preferred name in French**	**Version number**	**Revision number**	**Revision number**
Identificateur de propriété	**Exigence**	**Nom préférentiel en anglais**	**Nom préférentiel en français**	**Numéro de version**	**Numéro de révision**	**Numéro de révision**
MDC_P051_1	OPT [a]	Internal location	Emplacement interne	001	001	001
MDC_P051_2	OPT [a]	Street number	Numéro de la rue	001	001	001
MDC_P051_3	OPT [a]	Street	Rue	001	001	001
MDC_P051_4	OPT [a]	Postal box	Boîte postale	001	001	001
MDC_P051_5	OPT [a]	Town	Ville	001	001	001
MDC_P051_6	OPT [a]	Region	Région	001	001	001
MDC_P051_7	OPT [a]	Postal code	Code postal	001	001	001
MDC_P051_8	OPT [a]	Country	Pays	001	001	001
MDC_P051_9	OPT [a]	Facsimile number	Numéro de fac-similé	001	001	001
MDC_P051_10	OPT [a]	Telephone number	Numéro de téléphone	001	001	001
MDC_P051_11	OPT [a]	E-mail	Adresse courriel	001	001	001
MDC_P051_12	OPT [a]	Telex number	Numéro de télex	001	001	001
MDC_P211	OPT	Segment	Segment	001	001	001

[a] At least one attribute of address is mandatory.

Table E.5 – Meta-properties used by enumeration meta-class *(1 of 2)*

MMDC_P001	MMDC_P102	MMDC_P004_1.EN	MMDC_P004_1.FR	MMDC_P013	MMDC_P014	MMDC_P014.FR
Property ID	**Requirement**	**Preferred name in English**	**Preferred name in French**	**Version number**	**Revision number**	**Revision number**
Identificateur de propriété	**Exigence**	**Nom préférentiel en anglais**	**Nom préférentiel en français**	**Numéro de version**	**Numéro de révision**	**Numéro de révision**
MDC_P001_12	KEY	Code	Code	001	001	001
MDC_P043	MAND	Enumerated list of terms	Liste énumérée de termes	001	001	001
MDC_P044	MAND	Enumeration code list	Liste des codes d'énumération	001	001	001
MDC_P045	OPT	Number of selections	nombre de sélections	001	001	001
MDC_P046	OPT	Type of list	Type de liste	001	001	001
MDC_P002_1	KEY	Version number	Numéro de version	001	001	001
MDC_P002_2.<lang>	MAND	Revision number	Numéro de révision	001	001	001
MDC_P003_1	MAND	Date of original definition	Date de la définition originale	001	001	001
MDC_P003_2	MAND	Date of current version	Date de la version actuelle	001	001	001
MDC_P003_3.<lang>	OPT	Date of current revision	Date de la révision actuelle	001	001	001

Table E.5 *(2 of 2)*

MMDC_P001	MMDC_P102	MMDC_P004_1.EN	MMDC_P004_1.FR	MMDC_P013	MMDC_P014	MMDC_P014.FR
Property ID	**Requirement**	**Preferred name in English**	**Preferred name in English**	**Version number**	**Revision number**	**Revision number**
Identificateur de propriété	**Exigence**	**Nom préférentiel en anglais**	**Nom préférentiel en français**	**Numéro de version**	**Numéro de révision**	**Numéro de révision**
MDC_P004_1.<lang>	MAND	Preferred name	Nom préférentiel	001	001	001
MDC_P004_2	OPT	Synonymous name	Nom synonyme	001	001	001
MDC_P004_3.<lang>	OPT	Short name	Nom abrégé	001	001	001
MDC_P004_4	OPT	Name icon	Icône de nom	001	001	001
MDC_P005.<lang>	MAND	Definition	Définition	001	001	001
MDC_P006_1	OPT	Source document of definition	Document source de définition	001	001	001
MDC_P006_2	OPT	Source document of value	Document source de valeur	001	001	001
MDC_P007_1.<lang>	OPT	Note	Note	001	001	001
MDC_P007_2.<lang>	OPT	Remark	Remarque	001	001	001
MDC_P021	MAND	Definition class	Classe de définition	001	001	001
MDC_P112	OPT	Description	Description	001	001	001
MDC_P113	OPT	Example	Exemple	001	001	001
MDC_P211	OPT	Segment	Segment	001	001	001
MDC_P231	OPT[a]	Applicable terms	Termes applicables	001	001	001

[a] Not necessary in the case that an enumeration directly references terms in terms meta-class.

Table E.6 – Meta-properties used by datatype meta-class *(1 of 2)*

MMDC_P001	MMDC_P102	MMDC_P004_1.EN	MMDC_P004_1.FR	MMDC_P013	MMDC_P014	MMDC_P014.FR
Property ID	**Requirement**	**Preferred name in English**	**Preferred name in French**	**Version number**	**Revision number**	**Revision number**
Identificateur de propriété	**Exigence**	**Nom préférentiel en anglais**	**Nom préférentiel en français**	**Numéro de version**	**Numéro de révision**	**Numéro de révision**
MDC_P001_7	KEY	Code	Code	001	001	001
MDC_P002_1	KEY	Version number	Numéro de version	001	001	001
MDC_P002_2.<lang>	MAND	Revision number	Numéro de révision	001	001	001
MDC_P003_1	MAND	Date of original definition	Date de la définition originale	001	001	001
MDC_P003_2	MAND	Date of current version	Date de la version actuelle	001	001	001
MDC_P003_3.<lang>	OPT	Date of current revision	Date de la révision actuelle	001	001	001
MDC_P004_1.<lang>	MAND	Preferred name	Nom préférentiel	001	001	001
MDC_P004_2	OPT	Synonymous name	Nom synonyme	001	001	001

Table E.6 *(2 of 2)*

MMDC_P001	MMDC_P102	MMDC_P004_1.EN	MMDC_P004_1.FR	MMDC_P013	MMDC_P014	MMDC_P014.FR
Property ID	**Requirement**	**Preferred name in English**	**Preferred name in French**	**Version number**	**Revision number**	**Revision number**
Identificateur de propriété	**Exigence**	**Nom préférentiel en anglais**	**Nom préférentiel en français**	**Numéro de version**	**Numéro de révision**	**Numéro de révision**
MDC_P004_3.<lang>	OPT	Short name	Nom abrégé	001	001	001
MDC_P004_4	OPT	Name icon	Icône de nom	001	001	001
MDC_P021	KEY	Definition class	Classe de définition	001	001	001
MDC_P022	MAND	Data type	Type de données	001	001	001
MDC_P023	OPT	Unit structure	Structure d'unité	001	001	001
MDC_P023_1	OPT[a]	Unit in text	Unité en texte	001	001	001
MDC_P023_2	OPT	Unit in SGML	Unité en SGML	001	001	001
MDC_P024	OPT	Value format	Format de valeur	001	001	001
MDC_P041	OPT[b]	Code for unit	Code pour l'unité	001	001	001
MDC_P042	OPT	Codes for alternative unit	Codes pour les unités alternatives	001	001	001
MDC_P111	OPT	Alternative_units	Unités alternatives	001	001	001
MDC_P112	OPT	Description	Description	001	001	001
MDC_P113	OPT	Example	Exemple	001	001	001
MDC_P211	OPT	Segment	Segment	001	001	001

[a] Mandatory if "MDC_P023_2 Unit in SGML" has a value, where the units expressed in text and SGML shall be semantically the same.

[b] When the code for unit is used, it is still recommendable to write the unit in text (MDC_P023_1) at the same time for validation purposes.

Table E.7 – Meta-properties used by document meta-class *(1 of 4)*

MMDC_P001	MMDC_P102	MMDC_P004_1.EN	MMDC_P004_1.FR	MMDC_P013	MMDC_P014	MMDC_P014.FR
Property ID	**Requirement**	**Preferred name in English**	**Preferred name in French**	**Version number**	**Revision number**	**Revision number**
Identificateur de propriété	**Exigence**	**Nom préférentiel en anglais**	**Nom préférentiel en français**	**Numéro de version**	**Numéro de révision**	**Numéro de révision**
MDC_P001_8	KEY	Code	Code	001	001	001
MDC_P002_1	KEY	Version number	Numéro de version	001	001	001
MDC_P002_2.<lang>	MAND	Revision number	Numéro de révision	001	001	001
MDC_P002_3	MAND	Content revision	Révision du contenu	001	001	001
MDC_P003_1	MAND	Date of original definition	Date de la définition originale	001	001	001
MDC_P003_2	MAND	Date of current version	Date de la version actuelle	001	001	001
MDC_P003_3.<lang>	OPT	Date of current revision	Date de la révision actuelle	001	001	001
MDC_P004_1.<lang>	MAND	Preferred name	Nom préférentiel	001	001	001

Table E.7 *(2 of 4)*

MMDC_P001	MMDC_P102	MMDC_P004_1.EN	MMDC_P004_1.FR	MMDC_P013	MMDC_P014	MMDC_P014.FR
Property ID	**Requirement**	**Preferred name in English**	**Preferred name in French**	**Version number**	**Revision number**	**Revision number**
Identificateur de propriété	**Exigence**	**Nom préférentiel en anglais**	**Nompréférentiel en Français**	**Numéro de version**	**Numéro de révision**	**Numéro de révision**
MDC_P004_2	OPT	Synonymous name	Nom synonyme	001	001	001
MDC_P004_3.<lang>	OPT	Short name	Nom abrégé	001	001	001
MDC_P004_4	OPT	Name icon	Icône de nom	001	001	001
MDC_P005.<lang>	MAND	Definition	Définition	001	001	001
MDC_P007_1.<lang>	OPT	Note	Note	001	001	001
MDC_P007_2.<lang>	OPT	Remark	Remarque	001	001	001
MDC_P021	KEY	Definition Class	Classe de définition	001	001	001
MDC_P061_1	OPT	Document organization ID	Identificateur d'organisation	001	001	001

Table E.7 *(3 of 4)*

MMDC_P001	MMDC_P102	MMDC_P004_1.EN	MMDC_P004_1.FR	MMDC_P013	MMDC_P014	MMDC_P014.FR
Property ID	**Requirement**	**Preferred name in English**	**Preferred name in French**	**Version number**	**Revision number**	**Revision number**
Identificateur de propriété	**Exigence**	**Nom préférentiel en anglais**	**Nom préférentiel en français**	**Numéro de version**	**Numéro de révision**	**Numéro de révision**
MDC_P061_2	MAND	Document organization name	Nom d'organisation	001	001	001
MDC_P061_3	OPT	Document organization description	Description d'organisation	001	001	001
MDC_P062.<lang>	OPT	Remote location	Emplacement distant	001	001	001
MDC_P064.<lang>	OPT	Character encoding	Codage de caractères	001	001	001
MDC_P065_2.<lang>	OPT	Main content file	Fichier de contenu principal	001	001	001
MDC_P065_3.<lang>	OPT	Main content encoding	Codage du contenu principal	001	001	001
MDC_P065_4.<lang>	OPT	Main content mime	Mime contenu principal	001	001	001
MDC_P065_5.<lang>	OPT	Main content exchange format	Format d'échange du contenu principal	001	001	001

Table E.7 *(4 of 4)*

MMDC_P001	MMDC_P102	MMDC_P004_1.EN	MMDC_P004_1.FR	MMDC_P013	MMDC_P014	MMDC_P014.FR
Property ID	**Requirement**	**Preferred name in English**	**Preferred name in French**	**Version number**	**Revision number**	**Revision number**
Identificateur de propriété	**Exigence**	**Nom préférentiel en anglais**	**Nom préférentiel en français**	**Numéro de version**	**Numéro de révision**	**Numéro de révision**
MDC_P065_6.<lang>	OPT	Main content format RFC	Format RFC du contenu principal	001	001	001
MDC_P065_7.<lang>	OPT	Main content http file name	Nom de fichier http du contenu principal	001	001	001
MDC_P065_8.<lang>	OPT	Main content http directory	Répertoire http du contenu principal	001	001	001
MDC_P065_9.<lang>	OPT	Main content remote access	Accès distant du contenu principal	001	001	001
MDC_P211	OPT	Segment	Segment	001	001	001
MDC_P231	OPT[a]	Applicable terms	Termes applicables	001	001	001

Annex F
(normative)

Properties for optional meta-classes

Annex F is intended to show the minimum requirement for extending the standard parcel format with an aim to accommodate various needs and necessities beyond the current scope of ISO 13584-IEC 61360 common dictionary model.

Thus neither the instances of UoM meta-class, nor those of Object meta-class are an integral part of the ISO 13584-IEC 61360 standard, but is an extension mechanism of the parcel format for a UoM parcel is.

In case that the language used for the representation of a property needs to be specified using a “<lang>” extension, the extension shall be substituted by a two-letter country code defined by ISO 639-1, possibly being followed by a two-letter country code based on ISO 3166-1. The latter code is used to specify a regional variant of the same language.

Table F.1 – Meta-properties used by object meta-class

MMDC_P001	MMDC_P102	MMDC_P004_1.EN	MMDC_P004_1.FR	MMDC_P013	MMDC_P014	MMDC_P014.FR
Property ID	Requirement	Preferred name in English	Preferred name in French	Version number	Revision number	Revision number
Identificateur de propriété	Exigence	Nom préférentiel en anglais	Nom préférentiel en français	Numéro de version	Numéro de révision	Numéro de révision
MDC_P066	KEY	Data object identifier	Identificateur d'objet de données	001	001	001
MDC_P067	OPT	Time stamp	Horodatage	001	001	001
MDC_P211	OPT	Segment	Segment	001	001	001

Table F.2 – Meta-properties used by UoM meta-class *(1 of 3)*

MMDC_P001	MMDC_P102	MMDC_P004_1.EN	MMDC_P004_1.FR	MMDC_P013	MMDC_P014	MMDC_P014.FR
Property ID	**Requirement**	**Preferred name in English**	**Preferred name in French**	**Version number**	**Revision number**	**Revision number**
Identificateur de propriété	**Exigence**	**Nom préférentiel en anglais**	**Nom préférentiel en français**	**Numéro de version**	**Numéro de révision**	**Numéro de révision**
MDC_P001_10	KEY	Code	Code	001	001	001
MDC_P002_1	KEY	Version number	Numéro de version	001	001	001
MDC_P002_2.<lang>	OPT	Revision number	Numéro de révision	001	001	001
MDC_P003_1	MAND	Date of original definition	Date de la définition originale	001	001	001
MDC_P003_2	MAND	Date of current version	Date de la version actuelle	001	001	001
MDC_P003_3. <lang>	OPT	Date of current revision	Date de la révision actuelle	001	001	001
MDC_P004_1. <lang>	MAND	Preferred name	Nom préférentiel	001	001	001

Table F.2 *(2 of 3)*

MMDC_P001	MMDC_P102	MMDC_P004_1. EN	MMDC_P004_1.FR	MMDC_P013	MMDC_P014	MMDC_P014.FR
Property ID	**Requirement**	**Preferred name in English**	**Preferred name in French**	**Version number**	**Revision number**	**Revision number**
Identificateur de propriété	**Exigence**	**Nom préférentiel en anglais**	**Nom préférentiel en français**	**Numéro de version**	**Numéro de révision**	**Numéro de révision**
MDC_P004_2	OPT	Synonymous name	Nom synonyme	001	001	001
MDC_P004_3. <lang>	OPT	Short name	Nom abrégé	001	001	001
MDC_P005. <lang>	MAND	Definition	Définition	001	001	001
MDC_P006_1	OPT	Source document of definition	Document source de définition	001	001	001
MDC_P007_1. <lang>	OPT	Note	Note	001	001	001
MDC_P007_2. <lang>	OPT	Remark	Remarque	001	001	001
[a] Necessary only if some term used in some text attribute explicitly references a term defined in the term meta-class						

Table F.2 *(3 of 3)*

MMDC_P001	MMDC_P102	MMDC_P004_1. EN	MMDC_P004_1.FR	MMDC_P013	MMDC_P014	MMDC_P014.FR
Property ID	**Requirement**	**Preferred name in English**	**Preferred name in French**	**Version number**	**Revision number**	**Revision number**
Identificateur de propriété	**Exigence**	**Nom préférentiel en anglais**	**Nom préférentiel en français**	**Numéro de version**	**Numéro de révision**	**Numéro de révision**
MDC_P021	KEY	Definition class	Classe de définition	STRING_TYPE	001	001
MDC_P023	OPT [a]	Unit structure	Structure d'unité	STRING_TYPE	001	001
MDC_P023_1	OPT [b]	Unit in text	Unité en texte	STRING_TYPE	001	001
MDC_P023_2	OPT	Unit in SGML	Unité en SGML	STRING_TYPE	001	001
MDC_P112	OPT	Description	Description	001	001	001
MDC_P211	OPT	Segment	Segment	001	001	001
MDC_P231	OPT[c]	Applicable terms	Termes applicables	001	001	001

[a] Mandatory for quantitative data

[b] Mandatory if "MDC_P023_2 Unit in SGML" has a value

[c] necessary only if some term used in some text attribute explicitly references a term defined in the term meta-class

Table F.3 – Meta-properties used by term meta-class *(1 of 3)*

MMDC_P001	MMDC_P102	MMDC_P004_1.EN	MMDC_P004_1.FR	MMDC_P013	MMDC_P014	MMDC_P014.FR
Property ID	**Requirement**	**Preferred name in English**	**Preferred name in French**	**Version number**	**Revision number**	**Revision number**
Identificateur de propriété	**Exigence**	**Nom préférentiel en anglais**	**Nom préférentiel en Français**	**Numéro de version**	**Numéro de révision**	**Numéro de révision**
MDC_P001_11	KEY	Code	Code	001	001	001
MDC_P002_1	KEY	Version number	Numéro de version	001	001	001
MDC_P002_2.<lang>	OPT	Revision number	Numéro de révision	001	001	001
MDC_P003_1	MAND	Date of original definition	Date de la définition originale	001	001	001
MDC_P003_2	MAND	Date of current version	Date de la version actuelle	001	001	001
MDC_P003_3. <lang>	OPT	Date of current revision	Date de la révision actuelle	001	001	001
MDC_P004_1. <lang>	MAND	Preferred name	Nom préférentiel	001	001	001

Table F.3 *(2 of 3)*

MMDC_P001	MMDC_P102	MMDC_P004_1.EN	MMDC_P004_1.FR	MMDC_P013	MMDC_P014	MMDC_P014.FR
Property ID	**Requirement**	**Preferred name in English**	**Preferred name in French**	**Version number**	**Revision number**	**Revision number**
Identificateur de propriété	**Exigence**	**Nom préférentiel en anglais**	**Nom préférentiel en Français**	**Numéro de version**	**Numéro de révision**	**Numéro de révision**
MDC_P004_2	OPT	Synonymous name	Nom synonyme	001	001	001
MDC_P004_3.<lang>	OPT	Short name	Nom abrégé	001	001	001
MDC_P004_4	OPT	Name icon	Icône de nom	001	001	001
MDC_P005. <lang>	MAND	Definition	Définition	001	001	001
MDC_P006_1	OPT	Source document of definition	Document source de définition	001	001	001
MDC_P007_1. <lang>	OPT	Note	Note	001	001	001
MDC_P007_2. <lang>	OPT	Remark	Remarque	001	001	001
MDC_P008_2	OPT	Graphics	Graphisme	001	001	001

Table F.3 *(3 of 3)*

MMDC_P001	MMDC_P102	MMDC_P004_1.EN	MMDC_P004_1.FR	MMDC_P013	MMDC_P014	MMDC_P014.FR
Property ID	**Requirement**	**Preferred name in English**	**Preferred name in French**	**Version number**	**Revision number**	**Revision number**
Identificateur de propriété	**Exigence**	**Nom préférentiel en anglais**	**Nom préférentiel en Français**	**Numéro de version**	**Numéro de révision**	**Numéro de révision**
MDC_P021	MAND	Definition class	Classe de définition	001	001	001
MDC_P022	OPT	Data type	Type de données	001	001	001
MDC_P025_1	OPT [a]	Preferred letter symbol in text	Symbole littéral préférentiel en texte	001	001	001
MDC_P025_2	OPT	Preferred letter symbol in SGML	Symbole littéral préférentiel en SGML	001	001	001
MDC_P025_3	OPT	Synonymous letter symbols	Symboles littéraux synonymes	001	001	001
MDC_P112	OPT	Description	Description	001	001	001
MDC_P113	OPT	Example	Exemple	001	001	001
MDC_P114	OPT	Quantity	Quantité	001	001	001
MDC_P211	OPT	Segment	Segment	001	001	001

[a] Mandatory if “MDC_P025_2 Preferred letter symbol in SGML” has a value

Table F.4 – Meta-properties used by relation meta-class *(1 of 3)*

MMDC_P001	MMDC_P102	MMDC_P004_1.EN	MMDC_P004_1.FR	MMDC_P013	MMDC_P014	MMDC_P014.FR
Property ID	**Requirement**	**Preferred name in English**	**Preferred name in French**	**Version number**	**Revision number**	**Revision number**
Identificateur de propriété	**Exigence**	**Nom préférentiel en anglais**	**Nom préférentiel en Français**	**Numéro de version**	**Numéro de révision**	**Numéro de révision**
MDC_P001_13	KEY	Code	Code	001	001	001
MDC_P002_1	KEY	Version number	Numéro de version	001	001	001
MDC_P002_2.<lang>	MAND	Revision number	Numéro de révision	001	001	001
MDC_P003_1	MAND	Date of original definition	Date de la définition originale	001	001	001
MDC_P003_2	MAND	Date of current version	Date de la version actuelle	001	001	001
MDC_P003_3.<lang>	OPT	Date of current revision	Date de la révision actuelle	001	001	001
MDC_P004_1.<lang>	MAND	Preferred name	Nom préférentiel	001	001	001
MDC_P004_2	OPT	Synonymous name	Nom synonyme	001	001	001
MDC_P004_3.<lang>	OPT	Short name	Nom abrégé	001	001	001
MDC_P004_4	OPT	Name icon	Icône de nom	001	001	001
MDC_P005.<lang>	MAND	Definition	Définition	001	001	001
MDC_P006_1	OPT	Source document of definition	Document source de définition	001	001	001
MDC_P007_1.<lang>	OPT	Note	Note	001	001	001
MDC_P008_2	OPT	Graphics	Graphisme	001	001	001
MDC_P008_3	OPT	Graphic Properties	Propriétés graphiques	001	001	001

Table F.4 *(2 of 3)*

MMDC_P001	MMDC_P102	MMDC_P004_1.EN	MMDC_P004_1.FR	MMDC_P013	MMDC_P014	MMDC_P014.FR
Property ID	**Requirement**	**Preferred name in English**	**Preferred name in French**	**Version number**	**Revision number**	**Revision number**
Identificateur de propriété	**Exigence**	**Nom préférentiel en anglais**	**Nom préférentiel en Français**	**Numéro de version**	**Numéro de révision**	**Numéro de révision**
MDC_P007_2. <lang>	OPT	Remark	Remarque	001	001	001
MDC_P021	KEY	Definition Class	Classe de définition	001	001	001
MDC_P112	OPT	Description	Description	001	001	001
MDC_P113	OPT	Example	Exemple	001	001	001
MDC_P200	MAND	Relation type	Type de relation	001	001	001
MDC_P201	OPT	Domain of the relation	Domaine de la relation	001	001	001
MDC_P202	OPT	Domain of the function	Domaine de la fonction	001	001	001
MDC_P203	OPT	Codomain of the function	Co-domaine de la fonction	001	001	001
MDC_P204	OPT	Formula	Formule	001	001	001

Table F.4 *(3 of 3)*

MMDC_P001	MMDC_P102	MMDC_P004_1.EN	MMDC_P004_1.FR	MMDC_P013	MMDC_P014	MMDC_P014.FR
Property ID	**Requirement**	**Preferred name in English**	**Preferred name in French**	**Version number**	**Revision number**	**Revision number**
Identificateur de propriété	**Exigence**	**Nom préférentiel en anglais**	**Nom préférentiel en Français**	**Numéro de version**	**Numéro de révision**	**Numéro de révision**
MDC_P205	OPT	Language for formula interpretation	Langage pour l'interprétation de la formule	001	001	001
MDC_P206	OPT	External solver for the formula	Résolveur externe pour la formule	001	001	001
MDC_P207	OPT	Trigger event	Déclencheur d'événement	001	001	001
MDC_P208	OPT	Domain element type	Type d'élément de domaine	001	001	001
MDC_P209	OPT	Codomain element type	Type d'élément de co-domaine	001	001	001
MDC_P210	OPT	Role of the relation	Rôle de la relation	001	001	001
MDC_P211	OPT	Segment	Segment	001	001	001
MDC_P212	OPT	Super relation	Super-relation	001	001	001

Annex G
(normative)

Predefined classes and properties in Meta-Ontology

G.1 General

This normative annex comprises three different types of predefined ontological elements concerning Meta-Ontology (MO); one is a list of predefined meta-classes, which are instantiated in the class meta-meta class at MO layer, another is a list of predefined meta-properties instantiated in the property meta-class at MO layer, which are used as schema to define a Domain Ontology (DO) such as a reference dictionary maintained in IEC 61360 CDD database. The data model captured by those meta-classes at MO layer approximately corresponds to the IEC 61360-2/ISO 13584-42 Common Dictionary Model, with some extensions. Note that the list of predefined meta-properties used in each meta-class i.e., each parcel at DO layer, is already explained in Annex E. The definitions instantiated at MO layer are used in the header section of each parcel at DO layer as attributes (meta-data).

Most of the properties defined in meta-meta-classes (at MO layer) are expressed by STRING type. In most of the cases, the length of the codes is not specified in this standard, but if the content originates in a specific standard, the length of the string type properties shall follow the specification stipulated in the respective standard.

In case that the language used for the representation of a property needs to be specified, the extension “<lang>” shall be substituted by a two-letter country code defined by ISO 639-1, possibly followed by a two-letter country code based on ISO 3166-1. The latter code is used to specify a regional variant of the same language.

G.2 Predefined meta-classes in Meta-Ontology

Table G.1 gives a list of predefined classes in MO. These classes are used to model DO. The classes in the listing are the instances defined in the data section of the class meta-meta class at MO.

Table G.1 – List of meta-classes in Meta-Ontology *(1 of 2)*

MMDC_P000	MMDC_P102	MMDC_P004.EN	MMDC_P005.EN	MMDC_P007.EN	MMDC_P010
Class_ ID	**Requirement**	**Preferred name in English**	**Definition in English**	**Note in English**	**MOF modelling layer**
MDC_C001	OPT	Dictionary meta-class	meta-class being characterized by meta-properties that are necessary to identify and specify the information about the owner of the dictionary in a reference dictionary		M3-M2
MDC_C002	MAND	Class meta-class	meta-class being characterized by meta-properties that are necessary to identify and specify each class in a reference dictionary	Parcel for registering meta-classes with an ID and used to describe ontological concept, such as class, property, data type, etc.	M3-M2
MDC_C003	MAND	Property meta-class	meta-class being characterized by meta-properties that are necessary to identify and specify each property in a reference dictionary	Parcel for registering meta-properties with an ID and used to characterize ontological concepts such as class, property, data types, etc., as their attributes	M3-M2
MDC_C004	MAND	Supplier meta-class	meta-class being characterized by meta-properties that are necessary to identify and specify each information supplier in a reference dictionary	Parcel for registering information suppliers of ontological elements with an ID, and used to denote who is responsible for the management of the ontological elements	M3-M2
MDC_C005	OPT	Enumeration meta-class	meta-class being characterized by meta-properties that are necessary to identify and specify an enumeration as a list of terms.	Parcel for registering sets of enumerations with an ID, available for selection in an enumeration type property.	M3-M2
MDC_C006	OPT	Datatype meta-class	meta-class being characterized by meta-properties that are necessary to identify and specify each named data-type in a reference dictionary	Parcel for registering data types with an ID, available for specification as a data type of property.	M3-M2

Table G.1 *(2 of 2)*

MMDC_P000	MMDC_P102	MMDC_P004.EN	MMDC_P005.EN	MMDC_P007.EN	MMDC_P010
Class_ ID	**Requirement**	**Preferred name in English**	**Definition in English**	**Note in English**	**MOF modelling layer**
MDC_C007	OPT	Document meta-class	meta-class being characterized by meta-properties that are necessary to identify and specify each external document in a reference dictionary	Parcel for registering documents with an ID, available for reference in ontological concepts. If a document is referenced, the document meta-class is mandatory.	M3-M2
MDC_C008	OPT	Object meta-class	meta-class being characterized by meta-properties that are necessary to identify and specify each row of instance, as a data object, in the data section of a parcel	Parcel for registering information objects with an ID and some attributes about the object per se. These attributes are not part of the properties of a real world object that the information object is intended to describe	M3-M2
MDC_C009	OPT	UoM meta-class	meta-class being characterized by meta-properties that are necessary to identify and specify each unit of measurement in a reference dictionary	Parcel for registering units of measurements, available for specification with its ID, in a property	M3-M2
MDC_C010	OPT	Term meta-class	meta-class being characterized by meta-properties that are necessary to identify and specify terms used in an enumeration meta-class, or in a header section of parcel of a lower modelling layer	Parcel for registering terms used in enumerations as constants. It may be used to describe alias or synonyms for some key words, used in a lower modelling layer	M3-M2
MDC_C011	OPT	Relation meta-class	meta-class being characterized by meta-properties that are necessary to identify and specify relations other than those provided as defaults, such as class-property relations, class-super class relations	Functions may be defined as a subtype of a relation. The relation meta-class is also used to define some constraint among properties including a grouping of properties within a class.	M3-M2

G.3 Predefined meta-properties in meta-ontology

Table G.2 below gives a list of predefined properties in meta-ontology. The properties in the listing correspond to the instances defined in the data section of the property meta-meta class at MO layer. Thus, they will be used, in their turn, in the header section of several parcels at DO layer to model the ontological elements of a domain dictionary as instances.

Table G.2 – List of meta-properties defined at meta-ontology (MO) layer *(1 of 19)*

MMDC_P001	MMDC_P004_1.EN	MMDC_P005.EN	MMDC_P007.EN	MMDC_P008	MMDC_P013
Property ID	**Preferred name in English**	**Definition in English**	**Note in English**	**Data type**	**Version number**
MDC_P001_1	Dictionary code	code that identifies a dictionary	This attribute may be used to record the dictionary identifier, not based on IRDI	STRING_TYPE	001
MDC_P001_2	Supplier code	globally unique identifier of an information supplier in an IRDI sequence, used in a reference dictionary	This corresponds to RAI part of IRDI.	STRING_TYPE	001
MDC_P001_3[a]	Code	(Currently not used)	(Obsolete)	STRING_TYPE	001
MDC_P001_4[a]	Enumeration code	(Currently not used)	Now, obsolete. Superseded by MDC_P044 “Enumeration code list”	STRING_TYPE	001
MDC_P001_5	Code	globally unique identifier of a class in a reference dictionary in a form of IRDI	The value shall be described by the Class ID.	STRING_TYPE	001
MDC_P001_6	Code	globally unique identifier of a property in a reference dictionary in a form of IRDI	The value shall be described by property code.	STRING_TYPE	001
MDC_P001_7	Code	globally unique identifier of a data type in a reference dictionary in a form of IRDI	The value shall be described by data type code.	STRING_TYPE	001
MDC_P001_8	Code	globally unique identifier of a document in a reference dictionary in a form of IRDI	The value shall be described by the document code.	STRING_TYPE	001
MDC_P001_9	Code	(Currently not used)	(Obsolete)	ICID_STRING	001
MDC_P001_10	Code	globally unique identifier of a unit of measurement in a reference dictionary in a form of IRDI		ICID_STRING	001

Table G.2 *(2 of 19)*

MMDC_P001	MMDC_P004_1.EN	MMDC_P005.EN	MMDC_P007.EN	MMDC_P008	MMDC_P013
Property ID	**Preferred name in English**	**Definition in English**	**Note in English**	**Data type**	**Version number**
MDC_P001_11	Code	globally unique identifier of a term in a form of IRDI	A term is often used as an item of value list in an enumeration.	ICID_STRING	001
MDC_P001_12	Code	globally unique identifier assigned to a list of enumerated values in a form of IRDI	Sequence of the identifier and the enumeration code, combined by a dot “Identifier.code” shall give the global identification for each enumerated value.	ICID_STRING	001
MDC_P001_13	Code	globally unique identifier of a relation in a form of IRDI		ICID_STRING	001
MDC_P001_14	Code	globally unique identifier of a reference dictionary in a form of IRDI	This shall not be confused with dictionary code (MDC_P001_1).	STRING_TYPE	001

Table G.2 *(3 of 19)*

MMDC_P001	MMDC_P004_1.EN	MMDC_P005.EN	MMDC_P007.EN	MMDC_P008	MMDC_P013
Property ID	**Preferred name in English**	**Definition in English**	**Note in English**	**Data type**	**Version number**
MDC_P002_1	Version number	version of an item that is updated when the update should influence the range of instances		STRING_TYPE	001
MDC_P002_2. <lang>	Revision number	revision of the same version of an item	Revision shall not affect the ranges of instances.	STRING_TYPE	001
MDC_P002_3	Content revision	revision that characterises the updating of the information of a content file		STRING_TYPE	001
MDC_P003_1	Date of original definition	date when an item was defined by its library data supplier and thus when it was declared as valid by this supplier	The value shall be in accordance with ISO 8601.	STRING_TYPE	001
MDC_P003_2	Date of current version	date when the current version was defined	The value shall be in accordance with ISO 8601.	STRING_TYPE	001

Table G.2 *(4 of 19)*

MMDC_P001	MMDC_P004_1.EN	MMDC_P005.EN	MMDC_P007.EN	MMDC_P008	MMDC_P013
Property ID	**Preferred name in English**	**Definition in English**	**Note in English**	**Data type**	**Version number**
MDC_P003_3.<lang>	Date of current revision	date of the last revision number change	The value shall be in accordance with ISO 8601.	STRING_TYPE	001
MDC_P004<lang>	Item name	name of an item (in full length whenever possible) used for communication and understanding		TRANSLATABLE_STRING_TYPE	001
MDC_P004_1.<lang>	Preferred name	name of an item (in full length whenever possible) used for communication and understanding		TRANSLATABLE_STRING_TYPE	001
MDC_P004_2	Synonymous name	synonyms to the preferred name provided to facilitate transition from the names used for local or historical reasons	The first element of the list describes a synonymous name and the second one describes its language code based on ISO 639, possibly with an extension.	SET(0,?) OF LIST(2,2) OF STRING_TYPE	001
MDC_P004_3.<lang>	Short name	Shortened representation of the preferred name of an item	This may be used as an identifier of an object in a software application.	TRANSLATABLE_STRING_TYPE	001
MDC_P004_4	Name icon	optional icon which graphically represents the description associated with a name		STRING_TYPE	001
MDC_P005.<lang>	Definition	description of the meaning of an item		TRANSLATABLE_STRING_TYPE	001

Table G.2 *(5 of 19)*

MMDC_P001	MMDC_P004_1.EN	MMDC_P005.EN	MMDC_P007.EN	MMDC_P008	MMDC_P013
Property ID	**Preferred name in English**	**Definition in English**	**Note in English**	**Data type**	**Version number**
MDC_P006_1	Source document of definition	reference to the source document from which the item definition was derived		STRING_TYPE	001
MDC_P006_2	Source document of value	reference to the source document in which the value is originally defined or found		STRING_TYPE	001
MDC_P007_1.<lang>	Note	further information on any part of the terminological record of the definition, that is essential to the understanding of the definition		TRANSLATABLE_STRING_TYPE	001
MDC_P007_2.<lang>	Remark	explanatory text to further clarify the meaning of the usage of the item		TRANSLATABLE_STRING_TYPE	001
MDC_P008_1	Simplified drawing	code of a document file that contains the image of the class to provide a visualisation		STRING_TYPE	001
MDC_P008_2	Graphics	code of a resource file that contains or generates the image of the property to provide a visualisation		STRING_TYPE	001
MDC_P008_3	Graphic properties	List of properties that are used to generate image by graphics	The list is used when Graphics (MDC_P008_2) is used by a function to generate image.	LIST OF ICID_STRING	001

Table G.2 *(6 of 19)*

MMDC_P001	MMDC_P004_1.EN	MMDC_P005.EN	MMDC_P007.EN	MMDC_P008	MMDC_P013
Property ID	**Preferred name in English**	**Definition in English**	**Note in English**	**Data type**	**Version number**
MDC_P010	Superclass	class that is designated as the canonical parent class of the present class		ICID_STRING	001
MDC_P011	Class type	type of class	Possible value shall be either “ITEM_CLASS”, “COMPONENT_CLASS”, “MATERIAL_CLASS”, “FEATURE_CLASS”, “ITEM_CLASS_CASE_OF”, “COMPONENT_CLASS_CASE_OF”, “MATERIAL_CLASS_CASE_OF”, “FEATURE_CLASS_CASE_OF”, or “CATEGORICAL_CLASS”	STRING_TYPE	001
MDC_P012	Supplier	supplier defining this item	The value shall be supplier code.	STRING_TYPE	001
MDC_P013	Is case of	set of referred classes from which some properties, types, and/or documents are imported	The element of the set shall be described by an ICID of class.	SET(0,?) OF STRING_TYPE	001
MDC_P014	Applicable properties	properties that are newly specified as applicable for this class or meta-class and for any of its sub-classes	The element of the set shall be described by an ICID of property.	SET(0,?) OF STRING_TYPE	001

Table G.2 *(7 of 19)*

MMDC_P001	MMDC_P004_1.EN	MMDC_P005.EN	MMDC_P007.EN	MMDC_P008	MMDC_P013
Property ID	**Preferred name in English**	**Definition in English**	**Note in English**	**Data type**	**Version number**
MDC_P015	Applicable types	types that are newly specified as applicable for this class and for any of its sub-classes	The element of the set shall be described by an ICID of data type.	SET(0,?) OF STRING_TYPE	001
MDC_P016	Sub-class selection properties	set of class valued properties which shall be assigned a value	The element of the set shall be described by an ICID of property.	SET(0,?) OF STRING_TYPE	001
MDC_P017	Class value assignment	set of combinations of a class valued property specified for Sub-class Selection and its assigned value in this class	The first element of the list describes an ICID of property and the second element describes its value code.	SET(0,?) OF LIST(2,2) OF STRING_TYPE	001
MDC_P018	Coded name	value domain of the Classifying DET of the superclass		STRING_TYPE	001
MDC_P020	Property data element type	selection among subtypes of property, by dependency on other conditions	Possible values shall be either “NON_DEPENDENT_P_DET”, “DEPENDENT_P_DET” “CONDITION_DET”, or “DEPENDENT_C_DET"	STRING_TYPE	001
MDC_P021	Definition class	class in which the item is defined	The value shall be described by the ICID of the class. If default supplier or default version is defined, supplier code or version may be omitted, respectively.	STRING_TYPE	001

Table G.2 *(8 of 19)*

MMDC_P001	MMDC_P004_1.EN	MMDC_P005.EN	MMDC_P007.EN	MMDC_P008	MMDC_P013
Property ID	**Preferred name in English**	**Definition in English**	**Note in English**	**Data type**	**Version number**
MDC_P022	Data type	data type of the property		STRING_TYPE	001
MDC_P023	Unit structure	unit in which the value of a quantitative property is expressed in structural decomposition		STRING_TYPE	001
MDC_P023_1	Unit in text	unit in which the value of a quantitative property is expressed in text representation		STRING_TYPE	001
MDC_P023_2	Unit in SGML	unit in which the value of a quantitative property is expressed in SGML representation		STRING_TYPE	001
MDC_P024	Value format	specification of the type and length of the representation of the value of a property intended as a maximum value format for communication and database storage	The value format shall be conformant to ISO 9735 and ISO 6093.	STRING_TYPE	001
MDC_P025_1	Preferred letter symbol in text	shorter name of the property in text representation		STRING_TYPE	001
MDC_P025_2	Preferred letter symbol in SGML	shorter name of the property in SGML representation		STRING_TYPE	001
MDC_P025_3	Synonymous letter symbols	set of combinations of the synonymous name and its SGML representation	The first element of the list describes the synonymous letter symbol in text and the second element describes the synonymous letter symbol in SGML.	SET(0,?) OF LIST(2,2) OF STRING_TYPE	001

Table G.2 *(9 of 19)*

MMDC_P001	MMDC_P004_1.EN	MMDC_P005.EN	MMDC_P007.EN	MMDC_P008	MMDC_P013
Property ID	**Preferred name in English**	**Definition in English**	**Note in English**	**Data type**	**Version number**
MDC_P027_1	Formula in text	rule or statement in mathematical form expressing semantics of a quantitative property described in text representation		STRING_TYPE	001
MDC_P027_2	Formula in SGML	rule or statement in mathematical form expressing semantics of a quantitative property described in SGML representation		STRING_TYPE	001
MDC_P028	Condition	set of context parameters on which a context dependent characteristic depends	Each element of the set shall be described by an ICID of the property. If default supplier is defined, supplier code may be omitted.	SET(0,?) OF STRING_TYPE	001
MDC_P029	Definition property	(Originally defining the referencing property of an enumeration in an old version of ISO 13584-35, which is not used in IEC 62656-1)	(Obsolete)	STRING_TYPE	001
MDC_P040	DET classification	classification of the different properties defined in order to make large collections of property definitions more manageable		STRING_TYPE	001
MDC_P041	Code for unit	reference to a UoM defined in the UoM sheet or in another standard		STRING_TYPE	001
MDC_P042	Codes for alternative units	reference by global IDs to alternative UoMs defined in the UoM meta class or elsewhere		SET(0,?) OF STRING_TYPE	001
MDC_P043	Enumerated list of terms	list of terms to be used for resolution of the meaning of the value, assigned to the property	Unique identifiers of the terms defined in a term meta-class must be used for definition of the list.	LIST(1,?) OF STRING_TYPE	001

Table G.2 *(10 of 19)*

MMDC_P001	MMDC_P004_1.EN	MMDC_P005.EN	MMDC_P007.EN	MMDC_P008	MMDC_P013
Property ID	**Preferred name in English**	**Definition in English**	**Note in English**	**Data type**	**Version number**
MDC_P044	Enumeration code list	list of codes to be displayed and assigned as the value for the selected item for the enumeration type property	Number of codes in the list shall correspond to the number of terms in MDC_P042	LIST(1,?) OF STRING_TYPE	001
MDC_P045	Number of selections	minimum and maximum number of elements to be chosen simultaneously	When this attribute is not present, (1,1) is assumed. Note that (0,1) is allowed	LIST (2,2) OF INTEGER	001
MDC_P046	Type of list	designation if the sequence in the list is intended as a ‘permutation’ or a ‘combination’	Values shall be either one of PERMUTATION (PERM for short) or COMBINATION (COMB)	STRING_TYPE	001
MDC_P050_1	Organization id	identifier that distinguishes the organization of the supplier		STRING_TYPE	001
MDC_P050_2	Organization name	label by which the organization of the supplier is known		STRING_TYPE	001
MDC_P050_3	Organization description	text that characterizes the organization of the supplier		STRING_TYPE	001
MDC_P051_1	Internal location	organization-defined address for internal mail delivery		STRING_TYPE	001
MDC_P051_2	Street number	number of a location on a street		STRING_TYPE	001
MDC_P051_3	Street	name of a street		STRING_TYPE	001
MDC_P051_4	Postal box	number of a postal box		STRING_TYPE	001
MDC_P051_5	Town	name of a town		STRING_TYPE	001
MDC_P051_6	Region	name of a region		STRING_TYPE	001
MDC_P051_7	Postal code	code that is used by the country’s postal service		STRING_TYPE	001
MDC_P051_8	Country	name of a country		STRING_TYPE	001

Table G.2 *(11 of 19)*

MMDC_P001	MMDC_P004_1.EN	MMDC_P005.EN	MMDC_P007.EN	MMDC_P008	MMDC_P013
Property ID	**Preferred name in English**	**Definition in English**	**Note in English**	**Data type**	**Version number**
MDC_P051_9	Facsimile number	number at which facsimiles may be received		STRING_TYPE	001
MDC_P051_10	Telephone number	number at which telephone calls may be received		STRING_TYPE	001
MDC_P051_11	E-mail	electronic address at which electronic mail may be received		STRING_TYPE	001
MDC_P051_12	Telex number	number at which telex messages may be received		STRING_TYPE	001
MDC_P061_1	Document organization ID	identifier that designates the organization that publishes the document		STRING_TYPE	001
MDC_P061_2	Document organization name	label by which the organization of the document is known		STRING_TYPE	001
MDC_P061_3	Document organization description	text that characterizes the organization of the document		STRING_TYPE	001
MDC_P062. <lang>	Remote location	absolute URL that specifies the document locator		TRANSLATABLE_ STRING_TYPE	001
MDC_P064. <lang>	Character encoding	particular character encoding used in all the external file that contains characters		TRANSLATABLE_ STRING_TYPE	001
MDC_P065_2. <lang>	Main content file	library external file		TRANSLATABLE_ STRING_TYPE	001
MDC_P065_3. <lang>	Main content encoding	encoding transformation performed on the content of the library external file, if present		TRANSLATABLE_ STRING_TYPE	001
MDC_P065_4. <lang>	Main content mime	MIME type of the http file		TRANSLATABLE_ STRING_TYPE	001
MDC_P065_5. <lang>	Main content exchange format	MIME subtype of the http file		TRANSLATABLE_ STRING_TYPE	001

Table G.2 *(12 of 19)*

MMDC_P001	MMDC_P004_1.EN	MMDC_P005.EN	MMDC_P007.EN	MMDC_P008	MMDC_P013
Property ID	**Preferred name in English**	**Definition in English**	**Note in English**	**Data type**	**Version number**
MDC_P065_5. <lang>	Main content exchange format	MIME subtype of the http file		TRANSLATABLE_ STRING_TYPE	001
MDC_P065_6. <lang>	Main content format RFC	possible IAB RFC that defines the MIME subtype		TRANSLATABLE_ STRING_TYPE	001
MDC_P065_7. <lang>	Main content http file name	file name to be assigned to the http file on the local Internet server		TRANSLATABLE_ STRING_TYPE	001
MDC_P065_8. <lang>	Main content http directory	optional directory to be assigned to the http file on the local Internet server		TRANSLATABLE_ STRING_TYPE	001
MDC_P065_9. <lang>	Main content remote access	possible absolute URL where the http file may be found on an Internet site		TRANSLATABLE_ STRING_TYPE	001
MDC_P066	Data object identifier	globally unique identifier of the data object	The ID shall be kept unchanged while the row of the data object remains unchanged. Once any piece of data is modified, a new ID shall be assigned to the row.	STRING_TYPE	001
MDC_P067	Time stamp	time stamp of the moment when the data object is created		STRING_TYPE	001
MDC_P068	Property constraint	set of constraints that apply to the property	Constraints must be enclosed within a pair of curly braces “{“ and “}”	SET OF STRING	001
MDC_P069	Data type constraint	set of constraints that apply to the property	Constraints must be enclosed within a pair of curly braces “{“ and “}”	SET OF STRING	001
MDC_P071	LIIM source_document_identifier	identifier of the document that contains the data specification		STRING_TYPE	001
MDC_P072	LIIM status	classification of the data specification with respect to its acceptance by the approving body of this International Standard, possibly followed by an integer version	Values are WD’, ‘CD’, ‘DIS’, ‘FDIS’, ‘IS’, ‘TS’, ‘PAS’, ‘ITA’	STRING_TYPE	001

Table G.2 *(13 of 19)*

MMDC_P001	MMDC_P004_1.EN	MMDC_P005.EN	MMDC_P007.EN	MMDC_P008	MMDC_P013
Property ID	**Preferred name in English**	**Definition in English**	**Note in English**	**Data type**	**Version number**
MDC_P073	LIIM name	identifier of the data specification as defined in the corresponding part of ISO 13584		STRING_TYPE	001
MDC_P074	LIIM date	year when the corresponding part of ISO 13584 reached its status		INT_TYPE	001
MDC_P075	LIIM application	identifier possibly defined in the corresponding part of ISO 13584 to characterise an allowed functional subset of the complete data specification		STRING_TYPE	001
MDC_P076	LIIM level	identifier that may be defined in the corresponding part of ISO 13584 that further characterises an allowed subset of the application subset		STRING_TYPE	001
MDC_P080	Global language	language when the dictionary is defined only in one language	It requires language code defined in ISO 639.	STRING_TYPE	001
MDC_P081	Source language	source language when the dictionary is defined in plural languages	It requires language code defined in ISO 639.	STRING_TYPE	001
MDC_P082	Identifier encoding	specification of the global encoding method of identifiers			
MDC_P090	Imported properties	set of properties that are imported from the other class	The element of the set shall be described by an ICID of property. In some cases, supplier code may be omitted.	SET(0,?) OF STRING_TYPE	001
MDC_P091	Imported types	set of types that are imported from the other class	The element of the set shall be described by an ICID of data type. If default supplier is defined, supplier code may be omitted.	SET(0,?) OF STRING_TYPE	001

Table G.2 *(14 of 19)*

MMDC_P001	MMDC_P004_1.EN	MMDC_P005.EN	MMDC_P007.EN	MMDC_P008	MMDC_P013
Property ID	**Preferred name in English**	**Definition in English**	**Note in English**	**Data type**	**Version number**
MDC_P093	Imported documents	set of documents that are imported from the other class	The element of the set shall be described by an ICID of document. If default supplier is defined, supplier code may be omitted.	SET(0,?) OF STRING_TYPE	001
MDC_P094	Applicable documents	documents that are newly specified as applicable for this class and for any of its sub-classes	The element of the set shall be described by an ICID of document.	SET(0,?) OF STRING_TYPE	001
MDC_P095	Property type classification	Categorization of properties according to their physical nature, based on ISO 31 series[b]	equivalent of the data element type classification in IEC 61360-1:2009. In this part of IEC 62656, however, just for archiving old dictionaries, the attribute is maintained. It should not be applied to a newly defined dictionary. Instead, make a reference to a quantity.	STRING_TYPE	001
MDC_P096	Property classification	labelling of properties of a class by an integer value, originally designed to give a security index to each property	First element of the list stores property code, and the second element stores an integer value, and the last element stores the name or meaning of the index value, which is an extension of ISO 13584-24/ISO 13584-25.	SET(0,?) OF LIST(3,3) OF STRING_TYPE	001
MDC_P097	Requirement	indispensability of each property and its value	Represented as a list of ICID of the property and its value. Possible value for each is one of “CONST”, "KEY","MAND","NOT NULL", "OPT", “OBS”, or "" that means null. When the value is not specified it must be taken as "OPT". “OBS” is an abbreviation of “Obsolete” and its value is optional. It also signifies the property is now archaic. Mandatory means the presence of property is required, but its value may be null or unspecified (blank).	SET(0,?) OF LIST(2,2) OF STRING_TYPE	001

Table G.2 *(15 of 19)*

MMDC_P001	MMDC_P004_1.EN	MMDC_P005.EN	MMDC_P007.EN	MMDC_P008	MMDC_P013
Property ID	**Preferred name in English**	**Definition in English**	**Note in English**	**Data type**	**Version number**
MDC_P098	Identification method for parcel	method of global identification for identifiers in the parcel	This is to record the global identification method used for identifiers in the parcel, including data section.	STRING_TYPE	001
MDC_P101	Alternate ID	alternate identifier of the property designated by the property ID	This attribute belongs to a property to record an alternate ID	LIST(0,?) OF STRING_TYPE	001
MDC_P102	Alternate class ID	alternate identifier of the class designated by the class ID		ICID STRING	001

Table G.2 *(16 of 19)*

MMDC_P001	MMDC_P004_1.EN	MMDC_P005.EN	MMDC_P007.EN	MMDC_P008	MMDC_P013
Property ID	**Preferred name in English**	**Definition in English**	**Note in English**	**Data type**	**Version number**
MDC_P110	Super property	property of which the current property is a specialization	When a property is substituted by its super property, the subset where the super-property is true with its value as a parameter includes the subset where the specialized property is true with its value as a parameter.	ICID_STRING	001
MDC_P111	Alternative units	information about other units of measurement that may be used for the property specified by the property ID		STRING_TYPE	001
MDC_P112	Description	textual explanation, but not necessarily definition of an item, according to the principles stipulated in ISO/IEC Directives Part 2	A description may comprise more than one sentence	TRANSLATABLE_ STRING_TYPE	001
MDC_P113	Example	sample values of the property	If more than one example is shown, the examples must be enclosed in a pair of brackets “{“ and “}”.	STRING / LIST OF STRING	001
MDC_P114	Quantity	quantity to which the unit(s) of the property shall belong	The referenced quantity shall be specified by an ICID. It is intended to reference the code defined in IEC/TS 62720	ICID_STRING	001

Table G.2 *(17 of 19)*

MMDC_P001	MMDC_P004_1.EN	MMDC_P005.EN	MMDC_P007.EN	MMDC_P008	MMDC_P013
Property ID	**Preferred name in English**	**Definition in English**	**Note in English**	**Data type**	**Version number**
MDC_P200	Relation type	type of the relation which is either FUNCTION or PREDICATION	If FUNCTION is specified, the codomain (range) of the function must be also specified. FUNC or PRED may be used as an abbreviation for FUNCTION and PREDICATION, respectively.	STRING_TYPE	001
MDC_P201	Domain of the relation	list of IDs of properties or classes that serve as the domain for the relation identified by the relation ID	If there is more than one element, they must be enclosed in a pair of parentheses.	LIST OF ICID_STRING	001
MDC_P202	Domain of the function	list of IDs of properties or classes that serve as the domain for the functional relation identified by the relation	If there is more than one element, they must be enclosed in a pair of parentheses. Either this or MDC_P201 shall be used for function.	LIST OF ICID_STRING	001
MDC_P203	Codomain of the function	ID of the property or class that serves as the codomain (value range) of a function		ICID_STRING	001
MDC_P204	Formula	logical or mathematical formula by which the constraining effect on the domain, and/or the value for the codomain in case of a functional relation, will be computed	The value of the meta-property in a domain or codomain having an ICID = icid shall be addressed as dom[icid] or cod[icid]. Value of other function may be retrieved as func[icid] .	STRING_TYPE	001

Table G.2 *(18 of 19)*

MMDC_P001	MMDC_P004_1.EN	MMDC_P005.EN	MMDC_P007.EN	MMDC_P008	MMDC_P013
Property ID	**Preferred name in English**	**Definition in English**	**Note in English**	**Data type**	**Version number**
MDC_P205	Language for formula interpretation	computer language or logical description system by which the formula can be interpreted		STRING_TYPE	001
MDC_P206	External solver for the formula	reference to an external solver of the formula that is passed to the environment of the parcel system and from which the value of the function will be returned		STRING_TYPE	001
MDC_P207	Trigger event	event that triggers an evaluation of the relation by the formula	A subset of the list of the (meta-) properties specified in the domain (MDC_P201 or MDC_P202)	STRING_TYPE	001
MDC_P208	Domain element type	type(s) of elements expected in the domain	Type(s) shall be specified by a data type or an ICID. For example, if the domain contains only classes, MDC_C002 shall be used.	STRING_TYPE	001
MDC_P209	Codomain element type	type of elements expected in the codomain	Type shall be specified by a data type or an ICID. For example, if the domain contains only classes, MDC_C002 shall be used.	STRING_TYPE	001
MDC_P210	Role of the relation	specific use of the relation	Same relation may be used for different purposes depending on the context. This meta property denotes as what the relation is used.	STRING_TYPE	001

Table G.2 *(19 of 19)*

MMDC_P001	MMDC_P004_1.EN	MMDC_P005.EN	MMDC_P007.EN	MMDC_P008	MMDC_P013
Property ID	**Preferred name in English**	**Definition in English**	**Note in English**	**Data type**	**Version number**
MDC_P211	Segment	grouping of entries within a meta-class or across meta-classes	Entries are in most cases properties or meta-properties but any other ontological elements may be grouped into segments An item may belong to several segments	LIST OF STRING	001
MDC_P212	Super relation	generic type of the current relation, of which the domain includes the domain of the current relation		ICID_STRING	001
MDC_P230	Applicable relations	relations that are applicable to the class or meta-class as constraint among one or several ontological elements	Note that a constraint on a property or properties may be modelled as a relation. Relations are not inherited into subclasses by default.	SET(0,?) OF ICID_STRING	001
MDC_P231	Applicable terms	terms that are newly specified as applicable to this class or meta-class and to any of its sub-classes	The element of the set shall be described by an ICID of property.	SET(0,?) OF ICID_STRING	001

[a] These meta-properties are maintained only for archiving an early version of IEC 61360 CDD. They are out of use for a new release of the dictionary.

[b] Please note that ISO 31 was withdrawn and replaced by ISO 80000. New release of IEC 61360-4 CDD will adopt IEC/TS 62720 for identifying units and quantities.

b

Annex H
(normative)

Predefined meta-relations in meta-ontology

Table H.1 below gives a list of predefined relations in meta-ontology. The relations in the listing correspond to the instances defined in the data section of the relation meta-meta class at MO layer. Thus, they will be used, in their turn, in the header section of several parcels at DO layer to model the ontological elements of a domain dictionary.

Table H.1 – List of meta-relations predefined at MO layer *(1 of 2)*

MMDC_P003	MMDC_P020	MMDC_P004.EN	MMDC_P030	MMDC_P005.EN	MMDC_P007.EN	MMDC_P015.EN
Relation ID	**Relation type**	**Preferred name in English**	**Role of the relation**	**Definition in English**	**Note in English**	**Remark**
MDC_R001	FUNC	Property characterization	conjunction	function that takes a conjunction of the properties of a class specified in the domain to form the intent of the class specified in the codomain	Generic definition of the classifier of instances for meta-meta classes	Separately existing properties are formed into one conjunctive intersection.
MDC_R002	FUNC	Class decomposition	composition	function that relates the classes specified in the domain to a class specified in the codomain as its parts or components	Generic definition of the composer of parts, i.e., has-a, for meta-meta classes	In the current IEC 61360-1, -2, composition need be constructed with class reference type properties.
MDC_R003	FUNC	Boolean decomposition	Boolean logic	function that applies a Boolean operation to the classes specified in the domain with some transformation applied on each node and relates to a class specified in the codomain	A set theoretic operation on geometric primitives is a typical case of this function	Currently, not available in IEC 61360/ISO 13584 common data model
MDC_R004	FUNC	Native Applicable Properties	characterization	function that collects the native properties of the present class	It returns the properties connected to the applicable properties (MDC_P014) of the present class, which are native ones in the class	
MDC_R005	FUNC	Known Applicable properties	characterization	function that collects inherited properties and add them to the set of properties newly defined in the present class	All the properties of the present class come to be known by this function.	

Table H.1 *(2 of 2)*

MMDC_P003	MMDC_P022	MMDC_P023	MMDC_P028	MMDC_P029	MMDC_P024	MMDC_P012.EN
Relation ID	**Domain of the function**	**Codomain of the function**	**Domain element type**	**Codomain element type**	**Formula**	**Short name**
MDC_R001	MDC_C003	MDC_C002	SET of ICID_STRING	SET of ICID_STRING		
MDC_R002	MDC_C002	MDC_C002	SET of ICID_STRING	SET of ICID_STRING		
MDC_R003	MDC_C002	MDC_C002	SET of ICID_STRING	SET of ICID_STRING		
MDC_R004	MDC_C003, MDC_R004	MDC_C002	SET of ICID_STRING	SET of ICID_STRING	(to be formalized)	NAPs() [a]
MDC_R005	MDC_C003, MDC_R004	MDC_C002	SET of ICID_STRING	SET of ICID_STRING	(to be formalized)	KAPs() [a]

[a] NAPs() and KAPs() are the signatures of functions based on shortnames, defined therein where NAPs stands for Native Applicable Properties, and KAPS stands for Known Applicable Properties.

Annex I
(normative)

Axiomatic properties used by each Meta-meta-class

The following Tables I.1 to I.4 give lists of axiomatic (meta-)properties defined as instances in AO and used in some header section of a meta-meta class in MO (parcel sheet used in a dictionary exchange) as schema. Each table explains which meta-property being defined in Axiomatic property meta-class and identified by a code like ‘MMDC_Pxxx’, is used in each meta-meta class.

Table I.1 – Axiomatic properties used by class meta-meta-class *(1 of 2)*

MMDC_P001	MMDC_P102	MMDC_P004_1. EN	MMDC_P004_1.FR	MMDC_P013	MMDC_P014	MMDC_P014.FR
Property ID	**Requirement**	**Preferred name in English**	**Preferred name in French**	**Version number**	**Revision number**	**Revision number**
Identificateur de propriété	**Exigence**	**Nom préférentiel en anglais**	**Nom préférentiel en français**	**Numéro de version**	**Numéro de révision**	**Numéro de révision**
MMDC_P000	CONST[a]	Class ID	Identificateur de classe	001	001	001
MMDC_P001	MAND	Property ID	Identificateur de propriété	001	001	001
MMDC_P002	OPT	Term ID	Identificateur de terme	001	001	001
MMDC_P003	MAND	Relation ID	Identificateur de relation	001	001	001
MMDC_P004_1.<lang>	MAND	Preferred name	Nom préférentiel	001	001	001
MMDC_P004_2.<lang>	MAND	Preferred name of the class	Nom préférentiel de la classe	001	001	001
MMDC_P005. <lang>	MAND	Definition	Définition	001	001	001
MMDC_P006	OPT	Source document for definition	Document source de définition	001	001	001
MMDC_P007. <lang>	OPT	Note	Note	001	001	001
MMDC_P008	MAND	Data type	Type de données	001	001	001
MMDC_P011.<lang>	OPT	Description	Description	001	001	001
MMDC_P013	KEY	Version number	Numéro de version	001	001	001
MMDC_P014	MAND	Revision number	Numéro de révision	001	001	001
MMDC_P015. <lang>	OPT	Remark	Remarque	001	001	001

[a] When the Class ID is used as an attribute (meta-property) of the class designated by the class ID, the value of the attribute is indispensable as a KEY component.

Table I.1 *(2 of 2)*

MMDC_P001	MMDC_P102	MMDC_P004_1. EN	MMDC_P004_1.FR	MMDC_P013	MMDC_P014	MMDC_P014.FR
Property ID	**Requirement**	**Preferred name in English**	**Preferred name in French**	**Version number**	**Revision number**	**Revision number**
Identificateur de propriété	**Exigence**	**Nom préférentiel en anglais**	**Nom préférentiel en français**	**Numéro de version**	**Numéro de révision**	**Numéro de révision**
MMDC_P050	MAND	Applicable properties	Propriétés applicables	001	001	001
MMDC_P051	MAND	Applicable relations	Relations applicables	001	001	001
MMDC_P052	OPT	Applicable terms	Termes applicables	001	001	001
MMDC_P102	OPT	Requirement	Exigence	001	001	001
MMDC_P103_1	OPT	Date of original definition	Date de la définition originale	001	001	001
MMDC_P103_2	MAND	Date of current version	Date de la version actuelle	001	001	001

Table I.2 – Axiomatic properties used by property meta-meta-class *(1 of 2)*

MMDC_P001	MMDC_P102	MMDC_P004_1. EN	MMDC_P004_1.FR	MMDC_P013	MMDC_P014	MMDC_P014.FR
Property ID	**Requirement**	**Preferred name in English**	**Preferred name in French**	**Version number**	**Revision number**	**Revision number**
Identificateur de propriété	**Exigence**	**Nom préférentiel en anglais**	**Nom préférentiel en français**	**Numéro de version**	**Numéro de révision**	**Numéro de révision**
MMDC_P000	CONST	Class ID	Identificateur de classe	001	001	001
MMDC_P001	KEY	Property ID	Identificateur de propriété	001	001	001
MMDC_P002	MAND	Term ID	Identificateur de terme	001	001	001
MMDC_P004_1.<lang>	MAND	Preferred name	Nom préférentiel	001	001	001
MMDC_P004_2.<lang>	MAND	Preferred name of the class	Nom préférentiel de la classe	001	001	001
MMDC_P005. <lang>	MAND	Definition	Définition	001	001	001
MMDC_P006	OPT	Source document for definition	Document source de définition	001	001	001
MMDC_P007. <lang>	OPT	Note	Note	001	001	001
MMDC_P008	MAND	Data type	Type de données	001	001	001
MMDC_P009	MAND	Definition class	Classe de définition	001	001	001

Table I.2 *(2 of 2)*

MMDC_P001	MMDC_P102	MMDC_P004_1. EN	MMDC_P004_1.FR	MMDC_P013	MMDC_P014	MMDC_P014.FR
Property ID	**Requirement**	**Preferred name in English**	**Preferred name in French**	**Version number**	**Revision number**	**Revision number**
Identificateur de propriété	**Exigence**	**Nom préférentiel en anglais**	**Nom préférentiel en français**	**Numéro de version**	**Numéro de révision**	**Numéro de révision**
MMDC_P0011.<lang>	OPT	Description	Description	001	001	001
MMDC_P012.<lang>	OPT	Short name	Nom abrégé	001	001	001
MMDC_P013	KEY	Version number	Numéro de version	001	001	001
MMDC_P014	OPT	Revision number	Numéro de révision	001	001	001
MMDC_P015. <lang>	OPT	Remark	Remarque	001	001	001
MMDC_P102	OPT	Requirement	Exigence	001	001	001
MMDC_P103_1	OPT	Date of original definition	Date de la définition originale	001	001	001
MMDC_P103_2	OPT	Date of current version	Date de la version actuelle	001	001	001

Table I.3 – Axiomatic properties used by term meta-meta-class

MMDC_P001	MMDC_P102	MMDC_P004_1. EN	MMDC_P004_1.FR	MMDC_P013	MMDC_P014	MMDC_P014.FR
Property ID	**Requirement**	**Preferred name in English**	**Preferred name in French**	**Version number**	**Revision number**	**Revision number**
Identificateur de propriété	**Exigence**	**Nom préférentiel en anglais**	**Nom préférentiel en français**	**Numéro de version**	**Numéro de révision**	**Numéro de révision**
MMDC_P000	CONST	Class ID	Identificateur de classe	001	001	001
MMDC_P001	MAND[a]	Property ID	Identificateur de propriété	001	001	001
MMDC_P002	KEY[a]	Term ID	Identificateur de terme	001	001	001
MMDC_P004_1.<lang>	MAND	Preferred name	Nom préférentiel	001	001	001
MMDC_P004_2.<lang>	MAND	Preferred name of the class	Nom préférentiel de la classe	001	001	001
MMDC_P005. <lang>	MAND	Definition	Définition	001	001	001
MMDC_P006	OPT	Source document for definition	Document source de définition	001	001	001
MMDC_P007. <lang>	OPT	Note	Note	001	001	001
MMDC_P008	MAND	Data type	Type de données	001	001	001
MMDC_P009	OPT	Definition class	Classe de définition	001	001	001
MMDC_P012.<lang>	OPT	Short name	Nom abrégé	001	001	001
MMDC_P013	KEY	Version number	Numéro de version	001	001	001
MMDC_P014	MAND	Revision number	Numéro de révision	001	001	001
MMDC_P015. <lang>	OPT	Remark	Remarque	001	001	001
MMDC_P050	OPT	Applicable relations	Relations applicable			
MMDC_P102	OPT	Requirement	Exigence	001	001	001
MMDC_P103_1	OPT	Date of original definition	Date de la définition originale	001	001	001
MMDC_P103_2	OPT	Date of current version	Date de la version actuelle	001	001	001

[a] When Term meta-meta class is used, sometimes, terms are treated just as a degenerated property. In this case, Property ID may be used to list property and terms indiffently.

Table I.4 – Axiomatic properties used by relation meta-meta-class *(1 of 2)*

MMDC_P001	MMDC_P102	MMDC_P004_1. EN	MMDC_P004_1.FR	MMDC_P013	MMDC_P014	MMDC_P014.FR
Property ID	Requirement	Preferred name in English	Preferred name in French	Version number	Revision number	Revision number
Identificateur de propriété	Exigence	Nom préférentiel en anglais	Nom préférentiel en français	Numéro de version	Numéro de révision	Numéro de révision
MMDC_P000	CONST	Class ID	Identificateur de classe	001	001	001
MMDC_P001	MAND	Property ID	Identificateur de propriété	001	001	001
MMDC_P002	OPT	Term ID	Identificateur de terme	001	001	001
MMDC_P003	KEY	Relation ID	Identificateur de relation	001	001	001
MMDC_P004_1.<lang>	MAND	Preferred name	Nom préférentiel	001	001	001
MMDC_P004_2.<lang>	MAND	Preferred name of the class	Nom préférentiel de la classe	001	001	001
MMDC_P005. <lang>	MAND	Definition	Définition	001	001	001
MMDC_P006	OPT	Source document for definition	Document source de définition	001	001	001
MMDC_P007. <lang>	OPT	Note	Note	001	001	001
MMDC_P008	MAND	Data type	Type de données	001	001	001
MMDC_P011.<lang>	OPT	Description	Description	001	001	001
MMDC_P012.<lang>	OPT	Short name	Nom abrégé	001	001	001
MMDC_P013	KEY	Version	Version	001	001	001
MMDC_P014	MAND	Revision number	Numéro de révision	001	001	001
MMDC_P015. <lang>	OPT	Remark	Remarque	001	001	001

Table I.4 *(2 of 2)*

MMDC_P001	MMDC_P102	MMDC_P004_1. EN	MMDC_P004_1.FR	MMDC_P013	MMDC_P014	MMDC_P014.FR
Property ID	**Requirement**	**Preferred name in English**	**Preferred name in French**	**Version number**	**Revision number**	**Revision number**
Identificateur de propriété	**Exigence**	**Nom préférentiel en anglais**	**Nom préférentiel en français**	**Numéro de version**	**Numéro de révision**	**Numéro de révision**
MMDC_P020	MAND	Relation type	Type de relation	001	001	001
MMDC_P021	OPT[a]	Domain of the relation	Domaine de la relation	001	001	001
MMDC_P022	OPT[a]	Domain of the function	Domaine de la fonction	001	001	001
MMDC_P023	OPT[b]	Codomain of the function	Co-domaine de la fonction	001	001	001
MMDC_P024	OPT	Formula	Formule	001	001	001
MMDC_P025	OPT	Language for formula interpretation	Langage pour l'interprétation de la formule	001	001	001
MMDC_P026	OPT	External solver for the formula	Résolveur externe pour la formule	001	001	001
MMDC_P028	OPT	Domain element type	Type d'élément de domaine	001	001	001
MMDC_P029	OPT	Codomain element type	Type d'élément de co-domaine	001	001	001
MMDC_P050	MAND	Applicable properties	Propriétés applicables	001	001	001
MMDC_P051	MAND	Applicable relations	Relations applicables	001	001	001
MMDC_P052	MAND	Applicable terms	Termes applicables	001	001	001
MMDC_P102	OPT	Requirement	Exigence	001	001	001
MMDC_P103_1	OPT	Date of original definition	Date de la définition originale	001	001	001
MMDC_P103_2	MAND	Date of current version	Date de la version actuelle	001	001	001

[a] Either "domain of the relation" or "domain of the function" shall exist for any relation.
[b] If the relation is of function type, Codomain of the function shall exist.

Annex J
(normative)

Predefined classes and properties in Axiomatic Ontology

J.1 General

This normative annex comprises listings of two types of predefined ontological elements of Axiomatic Ontology (AO); one is a list of predefined meta-classes and another is a list of predefined meta-properties. These definitions are used in the header section of each meta-class as attributes (meta-data) in defining each meta-meta class at MO layer.

J.2 Predefined meta-classes in Axiomatic Ontology

The following Table J.1 gives a list of predefined classes in AO.

Table J.1 – Predefined meta-classes in Axiomatic Ontology

MMDC_P000	MMDC_P102	MMDC_P004.EN	MMDC_P005.EN	MMDC_P007.EN	MMDC_P010
Class_ ID	**Requirement**	**Preferred name in English**	**Definition in English**	**Note in English**	**MOF modelling layer**
AMDC_C000	OPT	Nullity	abstract class that signifies emptiness	The class is formed as the intersection of all the classes. This is purely an abstract concept and the identifier is used as the one for nullity.	(All)
AMDC_C001	OPT	Universe	abstract class that signifies the class of all the classes	The class is formed as the union of all the classes. This is purely an abstract concept and the ID is used as the root of all the concepts.	(All)
AMDC_C002	MAND	Axiomatic class meta-class	class that defines meta-classes that collectively define the Axiomatic Ontology (AO),	Usually it is not necessary to provide this explicitly.	M4-M3
AMDC_C003	MAND	Axiomatic property meta-class	class that defines meta-properties used in meta-classes which collectively define the Axiomatic Otology (AO),	Usually it is not necessary to provide this explicitly.	M4-M3
MMDC_C002	MAND	Class meta-meta Class	class that defines meta-classes that collectively define a meta-ontology (MO),	All the meta-classes defined at MO layer and used in DO layer shall be listed as an instance of this meta-meta class.	M3-M2
MMDC_C003	MAND	Property meta-meta-class	class that defines meta-properties used in meta-classes which collectively define a meta otology (MO)	All the meta-properties defined at MO layer and used in DO layer shall be listed as an instance of this meta-meta class.	M3-M2
MMDC_C010	MAND	Term meta-meta-class	class that defines constant terms used in meta-classes and meta-properties at meta-ontology (MO) layer	This meta-meta class is used to add functionality to an ontology model.	M3-M2
MMDC_C011	MAND	Relation meta-meta class	class that defines relations among meta-classes and meta-properties at meta-ontology (MO) layer	This meta-meta-class is used to define or clarify what are the given constants data types, and keywords as terms.	M3-M2

NOTE 1 For ease of comprehension, IDs of the meta-meta classes in the AO layer are not issued in a consecutive order, but rather in correspondence to the IDs of the similar type of meta-class in the MO layer.

NOTE 2 Identifiers, AMDC_C004 through AMDC_C0011 are not used by domain dictionaries, for they are reserved for future extensions of the model.

NOTE 3 AMDC_C000 and AMDC_C001 can be used at all modelling layers for identifying absolute entirety and nullity. They need not be instantiated, but exist only as identifiers for pure concepts.

J.3 Predefined meta-properties in Axiomatic ontology

The following Table J.2 gives a list of all the predefined properties in AO.

Table J.2 – List of axiomatic meta-properties defined at Axiomatic Ontology (AO) layer *(1 of 6)*

MMDC_P001	MMDC_P004.EN	MMDC_P005.EN	MMDC_P007.EN	MMDC_P008	MMDC_P013
Property ID	**Preferred name in English**	**Definition in English**	**Note in English**	**Data type**	**Version number**
MMDC_P000	Class ID	global Identifier based on ICID of a class of ontological modeling elements		STRING_TYPE	001
MMDC_P001	Property ID	global identifier of a meta-property		STRING_TYPE	001
MMDC_P002	Term ID	global identifier of a term	This 'requirement' is for ontological elements at AO and MO.	STRING_TYPE	001
MMDC_P003	Relation ID	global identifier of a meta-relation		STRING_TYPE	001
MMDC_P004. <lang>	Item name	name that is preferred for use in referencing an ontological element in general	It must be used only when the kind of ontological element is unkown or insignificant.	TRANSLATABLE_ STRING_TYPE	001
MMDC_P004_1. <lang>	Preferred name	name that is preferred for use in referencing the preferred name of an ontological element in Axiomatic Ontology or Meta-Ontology	This is a specialization of the item name (MMDC_P004).	TRANSLATABLE_ STRING_TYPE	001
MMDC_P004_2. <lang>	Preferred name of the class	name that is preferred for use in referencing a class in Axiomatic Ontology and Meta-Ontology	This is a specialization of item name (MMDC_P004) and used at MO and AO.	TRANSLATABLE_ STRING_TYPE	001
MMDC_P004_3. <lang>	Preferred name of the term	name that is preferred for use in referencing a term defined in MO	This is a specialization of item name (MMDC_P004).	TRANSLATABLE_ STRING_TYPE	001
MMDC_P004_4. <lang>	Preferred name of the relation	name that is preferred for use in referencing a relation defined in MO	This is a specialization of item name (MMDC_P004).	TRANSLATABLE_ STRING_TYPE	001

Table J.2 *(2 of 6)*

MMDC_P001	MMDC_P004.EN	MMDC_P005.EN	MMDC_P007.EN	MMDC_P008	MMDC_P013
Property ID	**Preferred name in English**	**Definition in English**	**Note in English**	**Data type**	**Version number**
MMDC_P005.<lang>	Definition	statement about the meaning of the concept	Language for description may be specified by dot notation.	TRANSLATABLE_STRING_TYPE	001
MMDC_P006	Source document for definition	document from which the definition derives		STRING_TYPE	001
MMDC_P007.<lang>	Note	additional statement about the definition of the concept	Language for description may be specified by dot notation.	TRANSLATABLE_STRING_TYPE	001
MMDC_P008	Data type	data type of the meta property	This applies to Axiomatic Ontology and Meta-Ontology.	STRING_TYPE	001
MMDC_P009	Definition class	class under which the definition applies		STRING_TYPE	001
MMDC_P010	MOF modelling layer	modeling layer according to MOF definition	Since a parcel consists of two layers, description is in the form of M-N, where M for the header section and N for the instance section	STRING_TYPE	001
MMDC_P011.<lang>	Description	textual explanation, but not necessarily definition of an item, according to the principles stipulated in ISO/IEC Directives Part 2	A description may comprise more than one sentence. For description of something at Domain Ontology layer (DO), MDC_P0112 shall be used.	TRANSLATABLE_STRING_TYPE	001
MMDC_P012.<lang>	Short name	shortened representation of the preferred name of an item	This may be used as a keyword in a software application. When it is used in the instruction column of a parcel sheet, it must be preceded by a “#” character.	LIST OF TRANSLATABLE_STRING_TYPE	001

Table J.2 *(3 of 6)*

MMDC_P001	MMDC_P004.EN	MMDC_P005.EN	MMDC_P007.EN	MMDC_P008	MMDC_P013
Property ID	**Preferred name in English**	**Definition in English**	**Note in English**	**Data type**	**Version number**
MMDC_P013	Version number	version of an item that is updated when the update should influence the range of instances	This applies to Axiomatic Ontology and Meta-Ontology.	STRING_TYPE	001
MMDC_P014. <lang>	Revision number	revision of an item that is updated when the update does not influence the range of instances	This is used to record the version of a translation with respect to a version in the source language.	STRING_TYPE	001
MMDC_P015. <lang>	Remark	additional information about the definition or note of an item, that shall not affect the meaning of it	Additional information about the definition or note of the item, that shall not affect the meaning.	TRANSLATABLE_ STRING_TYPE	001
MMDC_P016	Letter symbol	preferred letter symbol of an ontological entity	When used for functions, there shall be no space character between words and all the characters shall be alphanumeric.	STRING_TYPE	001
MMDC_P020	Relation type	type of the relation which is either FUNCTION or PREDICATION	If FUNCTION is specified, the codomain of the function must be also specified. FUNC or PRED may be used as an abbreviation for FUNCTION and PREDICATION, respectively.	STRING_TYPE	001
MMDC_P021	Domain of the relation	list of IDs of properties or classes that serve as the domain for the relation identified by the relation ID	If there is more than one element, they must be enclosed in a pair of parentheses.	LIST OF STRING_TYPE	001
MMDC_P022	Domain of the function	list of IDs of properties or classes that serve as the domain for the functional relation identified by the relation ID	If there is more than one element, they must be enclosed in a pair of parentheses. Either this or MDC_P201 shall be used for function.	LIST OF STRING_TYPE	001
MMDC_P023	Codomain of the function	ID of the property or class that serves as the codomain (value range) of a function		STRING_TYPE	001

Table J.2 *(4 of 6)*

MMDC_P001	MMDC_P004.EN	MMDC_P005.EN	MMDC_P007.EN	MMDC_P008	MMDC_P013
Property ID	**Preferred name in English**	**Definition in English**	**Note in English**	**Data type**	**Version number**
MMDC_P024	Formula	logical or mathematical formula by which the constraining effect on the domain, and/or the value for the codomain in case of a functional relation, will be computed	The value of the meta-property in a domain or codomain having an ICID = icid shall be addressed as dom[icid] or cod[icid]. Value of other function may be retrieved as func[icid] .	STRING_TYPE	001
MMDC_P025	Language for formula interpretation	computer language or logical description system by which the formula can be interpreted		STRING_TYPE	001
MMDC_P026	External solver for the formula	reference to an external solver of the formula that is passed to the environment of the parcel system and from which the value of the function will be returned		STRING_TYPE	001
MMDC_P028	Domain element type	type of the elements expected in the domain, when they are of a homogeneous type	Type shall be specified by a data type or an ICID. For example, if the domain contains only classes, MDC_C002 shall be used.	ICID_STRING	001
MMDC_P029	Codomain element type	type of ontological elements expected in the codomain, when they are of a homogeneous type	Type shall be specified by a data type or an ICID. For example, if the domain contains only meta-classes, MMDC_C002 shall be used.	STRING_TYPE	001

Table J.2 *(5 of 6)*

MMDC_P001	MMDC_P004.EN	MMDC_P005.EN	MMDC_P007.EN	MMDC_P008	MMDC_P013
Property ID	**Preferred name in English**	**Definition in English**	**Note in English**	**Data type**	**Version number**
MMDC_P030	Role of the relation	specific use of the relation	Same relation may be used for different purposes depending on the context.	STRING_TYPE	001
MMDC_P031	Segment	grouping of entries within a meta-class or across meta-classes	Entries are in most cases properties or meta-properties but any other ontological elements may be grouped into segments. An item may belong to several segments.	LIST OF STRING	001
MMDC_P032	Super relation	generic type of the current relation, of which the domain includes the domain of the current relation		ICID_STRING	001
MMDC_P050	Applicable properties	set of properties that can be newly applied to this (meta-)class and its sub-classes	Class can be any class in AO and MO.	SET (0,?) OF ICID_STRING	001
MMDC_P051	Applicable relations	set of relations that can be newly applied to this (meta-)class and its sub-classes	Class can be any class in AO and MO.	SET (0,?) OF ICID_STRING	001
MMDC_P052	Applicable terms	set of terms that can be newly applied to this (meta-)class and its sub-classes	Class can be any class in AO and MO.	SET (0,?) OF ICID_STRING	001
MMDC_P053	Component classes	set of classes that are parts of this class	An element of the set forms a component of this class, but not necessarily a subset of this class.	SET (0,?) OF ICID_STRING	001

Table J.2 *(6 of 6)*

MMDC_P001	MMDC_P004.EN	MMDC_P005.EN	MMDC_P007.EN	MMDC_P008	MMDC_P013
Property ID	**Preferred name in English**	**Definition in English**	**Note in English**	**Data type**	**Version number**
MMDC_P100	Value format	specification of the type and length of the representation of the value of a property intended as a maximum value format for communication and database storage	Currently not used in IEC 62656-1, but is used in other parts of IEC 62656, for comparing attributes of different standards or specifying mappings between them.	STRING_TYPE	001
MMDC_P101	Map to EXPRESS	mapping to the EXPRESS code, used to define the common schema defined in IEC 61360-2/ISO 13584 Common Dictionary Schema		STRING_TYPE	001
MMDC_P102	Requirement	level of requirement of an ontological element and its value	This 'requirement' is for ontological elements at AO and MO.	STRING_TYPE	001
MMDC_P103_1	Date of original definition	date when the item is originally made	The value shall be in accordance with ISO 8601.	STRING_TYPE	001
MMDC_P103_2	Date of current version	date when the current version was made	The value shall be in accordance with ISO 8601.	STRING_TYPE	001
MMDC_P103_3	Date of current revision	date when the current revision was made	The value shall be in accordance with ISO 8601.	STRING_TYPE	001

Annex K
(informative)

Mapping of meta-properties to EXPRESS

K.1 EXPRESS mapping for mandatory meta-classes

The following Tables K.1 to K.6 give the mapping of meta-properties of mandatory meta-classes to the common dictionary model defined in EXPRESS language given in IEC 61360-2 and ISO 13584-42. Note that not all meta-properties are mappable to the EXPRESS language model, because some meta-properties are original in Parcellized Ontology Model (POM), and added in this part of IEC 62656.

The mapping will be helpful in developing a validator for IEC 61360-ISO 13584 compatible dictionary definitions represented by POM. In other words, it must be completely meaningful within the range of ISO/TS 13584-35 that is a PLIB-binding subset of this standard.

The EXPRESS language itself is defined in ISO 10303-11, and for the detailed explanation about the semantics and syntax of the language, please refer to the latest reference manual of the language.

Table K.1 – Mapping to EXPRESS modelling languages for meta-properties of dictionary meta-class *(1 of 3)*

MMDC_P001	MMDC_P004_1.EN	MMDC_P101
Property ID	**Preferred name in English**	**Map to EXPRESS**
MDC_P001_1	Dictionary code	dictionary.identified_by.code
MDC_P002_1	Version number	dictionary.identified_by.version
MDC_P002_2.<lang>	Revision number	dictionary.identified_by.revision
MDC_P004_1.<lang>	Preferred name	dictionary.names.preferred_name
MDC_P004_2	Synonymous name	dictionary.names.synonymous_names
MDC_P004_3.<lang>	Short name	dictionary.names.short_name
MDC_P004_4	Name icon	dictionary.names.icon\referenced_graphics.graphics_reference
MDC_P007_1.<lang>	Note	dictionary.note
MDC_P007_2.<lang>	Remark	dictionary.remark
MDC_P012	Supplier	dictionary.identified_by.defined_by
MDC_P071	LIIM source_document_identifier	dictionary.library_structure.source_document_identifier
MDC_P072	LIIM status	dictionary.library_structure.status
MDC_P073	LIIM name	dictionary.library_structure.name
MDC_P074	LIIM date	dictionary.library_structure.date
MDC_P075	LIIM application	dictionary.library_structure.application
MDC_P076	LIIM level	dictionary.library_structure.level
MDC_P080	Global language	global_language_assignment.language
MDC_P081	Source language	Currently not mapped. It will be mapped in the future edition of IEC 61360-2 or ISO 13584-42.

Table K.1 *(2 of 3)*

MMDC_P001	MMDC_P004_1.EN	MMDC_P101
Property ID	**Preferred name in English**	**Map to EXPRESS**
MDC_P001_5	Code	class_bsu.code
MDC_P002_1	Version number	class_bsu.version
MDC_P002_2.<lang>	Revision number	item_class.revision
MDC_P003_1	Date of original definition	item_class.time_stamps.date_of_original_definition
MDC_P003_2	Date of current version	item_class.time_stamps.date_of_current_version
MDC_P003_3.<lang>	Date of current revision	item_class.time_stamps.date_of_current_revision
MDC_P004_1.<lang>	Preferred name	item_class.names.preferred_name
MDC_P004_2	Synonymous name	item_class.names.synonymous_names
MDC_P004_3.<lang>	Short name	item_class.names.short_name
MDC_P004_4	Name icon	item_class.names.icon.graphics_reference
MDC_P005.<lang>	Definition	item_class.definition
MDC_P006_1	Source document of definition	item_class.source_doc_of_definition.document_identifier
MDC_P007_1.<lang>	Note	item_class.note
MDC_P007_2.<lang>	Remark	item_class.remark
MDC_P008_1	Simplified drawing	item_class.simplified_drawing.graphics_reference
MDC_P010	Superclass	item_class.its_superclass

Table K.1 *(3 of 3)*

MMDC_P001	MMDC_P004_1.EN	MMDC_P101
Property ID	**Preferred name in English**	**Map to EXPRESS**
MDC_P011	Class type	TYPEOF(class)
MDC_P012	Supplier	class_bsu.defined_by
MDC_P013	Is case of	item_class_case_of.is_case_of
MDC_P014	Applicable properties	item_class.described_by
MDC_P015	Applicable types	item_class.defined_types
MDC_P094	Applicable documents	class_document_relationship.related_tokens
MDC_P016	Sub-class selection properties	item_class.sub_class_properties
MDC_P017	Class value assignment	item_class.class_constant_values
MDC_P090	Imported properties	item_class_case_of.imported_properties
MDC_P091	Imported types	item_class_case_of.imported_types
MDC_P093	Imported documents	item_class_case_of.imported_documents
MDC_P018	Coded name	item_class.coded_name
MDC_P096	Property classification	Property_classification.classification_value
MDC_P097	Requirement	Not mapped
MDC_P098	Identification method for parcel	Not mapped

Table K.2 – Mapping to EXPRESS modelling languages for meta-properties of property meta-class *(1 of 2)*

MMDC_P001	MMDC_P004_1.EN	MMDC_P101
Property ID	**Preferred name in English**	**Map to EXPRESS**
MDC_P001_6	Code	property_bsu.code
MDC_P002_1	Version number	property_bsu.version
MDC_P002_2.<lang>	Revision number	property_det.revision
MDC_P003_1	Date of original definition	property_det.time_stamps.date_of_original_definition
MDC_P003_2	Date of current version	property_det.time_stamps.date_of_current_version
MDC_P003_3.<lang>	Date of current revision	property_det.time_stamps.date_of_current_revision
MDC_P004_1.<lang>	Preferred name	property_det.names.preferred_name
MDC_P004_2	Synonymous name	property_det.names.synonymous_names
MDC_P004_3.<lang>	Short name	property_det.names.short_name
MDC_P004_4	Name icon	property_det.names.icon.graphics_reference
MDC_P005.<lang>	Definition	property_det.definition
MDC_P006_1	Source document of definition	property_det.source_doc_of_definition.document_identifier
MDC_P007_1.<lang>	Note	property_det.note
MDC_P007_2.<lang>	Remark	property_det.remark
MDC_P008_2	Graphics	property_det.figure\referenced_graphics.graphics_reference
MDC_P020	Property data element type	TYPEOF(property_det)
MDC_P021	Definition class	property_bsu.name_scope

Table K.2 *(2 of 2)*

MMDC_P001	MMDC_P004_1.EN	MMDC_P101
Property ID	**Preferred name in English**	**Map to EXPRESS**
MDC_P022	Data type	property_det.domain
MDC_P023	Unit structure	property_det.domain.unit.structured_representation
MDC_P023_1	Unit in text	property_det.domain.unit.string_representation.text_representation
MDC_P023_2	Unit in SGML	property_det.domain.unit.string_representation.sgml_representation
MDC_P024	Value format	property_det.domain.value_format
MDC_P025_1	Preferred letter symbol in text	property_det.preferred_symbol.text_representation
MDC_P025_2	Preferred letter symbol in SGML	property_det.preferred_symbol.sgml_representation
MDC_P025_3	Synonymous letter symbols	property_det.synonymous_symbols
MDC_P027_1	Formula in text	property_det.formula.text_representation
MDC_P027_2	Formula in SGML	property_det.formula.sgml_representation
MDC_P028	Condition	dependent_p_det.depends_on
MDC_P040	DET classification	property_det.det_classification
MDC_P041	Code for unit	Currently not mapped
MDC_P042	Codes for alternative units	Currently not mapped

Table K.3 – Mapping to EXPRESS modelling languages for meta-properties of supplier meta-class

MMDC_P001	MMDC_P004_1.EN	MMDC_P101
Property ID	**Preferred name in English**	**Map to EXPRESS**
MDC_P001_2	Supplier code	In case it is mapped to BSU, then, it must be mapped to the following: supplier_bsu.code
MDC_P002_2.<lang>	Revision number	supplier_element.revision
MDC_P003_1	Date of original definition	supplier_element.time_stamps.date_of_original_definition
MDC_P003_2	Date of current version	supplier_element.time_stamps.date_of_current_version
MDC_P003_3.<lang>	Date of current revision	supplier_element.time_stamps.date_of_current_revision
MDC_P050_1	Organization id	supplier_element.org.id
MDC_P050_2	Organization name	supplier_element.org.name
MDC_P050_3	Organization description	supplier_element.org.description
MDC_P051_1	Internal location	supplier_element.addr.internal_location
MDC_P051_2	Street number	supplier_element.addr.street_number
MDC_P051_3	Street	supplier_element.addr.street
MDC_P051_4	Postal box	supplier_element.addr.postal_box
MDC_P051_5	Town	supplier_element.addr.town
MDC_P051_6	Region	supplier_element.addr.region
MDC_P051_7	Postal code	supplier_element.addr.postal_code
MDC_P051_8	Country	supplier_element.addr.country
MDC_P051_9	Facsimile number	supplier_element.addr.facsimile_number
MDC_P051_10	Telephone number	supplier_element.addr.telephone_number
MDC_P051_11	E-mail	supplier_element.addr.electronic_mail_address
MDC_P051_12	Telex number	supplier_element.addr.telex_number

Table K.4 – Mapping to EXPRESS modelling languages for meta-properties of enumeration meta-class

MMDC_P001	MMDC_P004_1.EN	MMDC_P101
Property ID	**Preferred name in English**	**Map to EXPRESS**
MDC_P001_12	Code	Currently not mapped
MDC_P043	Enumerated list of terms	value_domain.terms value_domain.its_value. meaning
MDC_P044	Enumeration code list	value_domain.its_value.value_ code
MDC_P045	Number of selections	Currently not mapped
MDC_P046	Type of list	Currently not mapped
MDC_P002_1	Version number	Currently not mapped
MDC_P002_2.<lang>	Revision number	Currently not mapped
MDC_P003_1	Date of original definition	Currently not mapped
MDC_P003_2	Date of current version	Currently not mapped
MDC_P003_3.<lang>	Date of current revision	Currently not mapped
MDC_P004_1.<lang>	Preferred name	Currently not mapped
MDC_P004_2	Synonymous name	Currently not mapped
MDC_P004_3.<lang>	Short name	Currently not mapped
MDC_P004_4	Name icon	Currently not mapped
MDC_P005.<lang>	Definition	Currently not mapped
MDC_P006_1	Source document of definition	value_domain.source_doc_of_value_domain
MDC_P007_1.<lang>	Note	Currently not mapped
MDC_P007_2.<lang>	Remark	Currently not mapped
MDC_P021	Definition class	Currently not mapped

Table K.5 – Mapping to EXPRESS modelling languages for meta-properties of datatype meta-class

MMDC_P001	MMDC_P004_1.EN	MMDC_P101
Property ID	**Preferred name in English**	**Map to EXPRESS**
MDC_P001_7	Code	data_type_bsu.code
MDC_P002_1	Version number	data_type_bsu.version
MDC_P002_2.<lang>	Revision number	data_type_element.revision
MDC_P003_1	Date of original definition	data_type_element.time_stamps.date_of_original_definition
MDC_P003_2	Date of current version	data_type_element.time_stamps.date_of_current_version
MDC_P003_3.<lang>	Date of current revision	data_type_element.time_stamps.date_of_current_revision
MDC_P004_1.<lang>	Preferred name	data_type_element.names.preferred_name
MDC_P004_2	Synonymous name	data_type_element.names.synonymous_names
MDC_P004_3.<lang>	Short name	data_type_element.names.short_name
MDC_P004_4	Name icon	data_type_element.names.icon.graphics_reference
MDC_P021	Definition class	data_type_bsu.name_scope
MDC_P022	Data type	data_type_element.type_definition
MDC_P023	Unit structure	data_type_element.type_definition.unit.structured_representation
MDC_P023_1	Unit in text	data_type_element.type_definition.unit.string_representation.text_representation
MDC_P023_2	Unit in SGML	data_type_element.type_definition.unit.string_representation.sgml_representation
MDC_P024	Value format	data_type_element.type_definition.value_format
MDC_P041	Code for unit	Currently not mapped

Table K.6 – Mapping to EXPRESS modelling languages for meta-properties of document meta-class *(1 of 2)*

MMDC_P001	MMDC_P004_1.EN	MMDC_P101
Property ID	**Preferred name in English**	**Map to EXPRESS**
MDC_P001_8	Code	document_bsu.code
MDC_P002_1	Version number	document_bsu.version
MDC_P002_2.<lang>	Revision number	document_element.revision
MDC_P002_3	Content revision	document_content.revision
MDC_P003_1	Date of original definition	document_element.time_stamps.date_of_original_definition
MDC_P003_2	Date of current version	document_element.time_stamps.date_of_current_version
MDC_P003_3.<lang>	Date of current revision	document_element.time_stamps.date_of_current_revision
MDC_P004_1.<lang>	Preferred name	document_element.names.preferred_name
MDC_P004_2	Synonymous name	document_element.names.synonymous_names
MDC_P004_3.<lang>	Short name	document_element.names.short_name
MDC_P004_4	Name icon	document_element.names.icon.graphics_reference
MDC_P005.<lang>	Definition	document_element.definition
MDC_P007_1.<lang>	Note	document_element.note
MDC_P007_2.<lang>	Remark	document_element.remark
MDC_P021	Definition class	document_bsu.name_scope
MDC_P061_1	Document organization ID	document_element.publishing_organisation.id
MDC_P061_2	Document organization name	document_element.publishing_organisation.name

Table K.6 *(2 of 2)*

MMDC_P001	MMDC_P004_1.EN	MMDC_P101
Property ID	**Preferred name in English**	**Map to EXPRESS**
MDC_P061_3	Document organization description	document_element.publishing_organisation.description
MDC_P062.<lang>	Remote location	document_element_with_translated_http_access.remote_locations
MDC_P064.<lang>	Character encoding	document_content.content.consists_of[i].character_encoding
MDC_P065_2.<lang>	Main content file	document_content.content.consists_of[i].content_file[1].file
MDC_P065_3.<lang>	Main content encoding	document_content.content.consists_of[i].content_file[1].content_encoding
MDC_P065_4.<lang>	Main content mime	document_content.content.consists_of[i].content_file[1].mime
MDC_P065_5.<lang>	Main content exchange format	document_content.content.consists_of[i].content_file[1].exchange_format
MDC_P065_6.<lang>	Main content format RFC	document_content.content.consists_of[i].content_file[1].format_rfc
MDC_P065_7.<lang>	Main content http file name	document_content.content.consists_of[i].content_file[1].http_file_name
MDC_P065_8.<lang>	Main content http directory	document_content.content.consists_of[i].content_file[1].http_directory.name
MDC_P065_9.<lang>	Main content remote access	document_content.content.consists_of[i].content_file[1].remote_access

K.2 EXPRESS mapping for optional meta-classes

The following Tables K.7 and K.8 give the mapping of meta-properties of optional meta-classes to the common dictionary model defined in EXPRESS language given in IEC 61360-2 and ISO 13584-42. Note that not all meta-properties are mappable to the EXPRESS language model, because some meta-properties are original in Parcellized Ontology Model (POM), and added in this part of IEC 62656.

The mapping will be helpful in developing a validator for IEC 61360-ISO 13584 compatible dictionary definitions represented by POM. In other words, it must be completely meaningful within the range of ISO/TS 13584-35 that is a PLIB-binding subset of this standard.

The EXPRESS language itself is defined in ISO 10303-11, and for the detailed explanation about the semantics and syntax of the language, please refer to the latest reference manual of the language.

Table K.7 – Mapping to EXPRESS modelling languages for meta-properties of object meta-class

MMDC_P001	MMDC_P004_1.EN	MMDC_P101
Property ID	**Preferred name in English**	**Map to EXPRESS**
MDC_P066	Data object identifier	Currently not mapped
MDC_P067	Time stamp	Currently not mapped
MDC_P001_10	Code	Currently not mapped
MDC_P002_1	Version number	Currently not mapped
MDC_P002_2.<lang>	Revision number	Currently not mapped
MDC_P003_1	Date of original definition	Currently not mapped
MDC_P003_2	Date of current version	Currently not mapped
MDC_P003_3. <lang>	Date of current revision	Currently not mapped
MDC_P004_1. <lang>	Preferred name	Currently not mapped
MDC_P004_2	Synonymous name	Currently not mapped
MDC_P004_3.<lang>	Short name	Currently not mapped
MDC_P005.<lang>	Definition	Currently not mapped
MDC_P006_1	Source document of definition	Currently not mapped
MDC_P007_1.<lang>	Note	Currently not mapped
MDC_P007_2.<lang>	Remark	Currently not mapped
MDC_P021	Definition class	Currently not mapped
MDC_P023	Unit structure	int_measure_type.unit.structured_representation
MDC_P023_1	Unit in text	int_measure_type.unit.string_representation.text_representation
MDC_P023_2	Unit in SGML	int_measure_type.unit.string_representation.sgml_representation

Table K.8 – Mapping to EXPRESS modelling languages for meta-properties of terminology meta-class *(1 of 2)*

MMDC_P001	MMDC_P004_1.EN	MMDC_P101
Property ID	**Preferred name in English**	**Map to EXPRESS**
MDC_P001_11	Code	Currently not mapped
MDC_P002_1	Version number	Currently not mapped
MDC_P002_2.<lang>	Revision number	Currently not mapped
MDC_P003_1	Date of original definition	Currently not mapped
MDC_P003_2	Date of current version	Currently not mapped
MDC_P003_3.<lang>	Date of current revision	Currently not mapped
MDC_P004_1.<lang>	Preferred name	If reference as value code, it is mapped to the following. value_domain.its_values[i].meaning.preferred_name
MDC_P004_2	Synonymous name	Currently not mapped For value code, it is mapped to the following. value_domain.its_values[i].meaning.synonymous_names
MDC_P004_3.<lang>	Short name	Currently not mapped For value code, it is mapped to the following. value_domain.its_values[i].meaning.short_name
MDC_P005.<lang>	Definition	Currently not mapped

Table K.8 *(2 of 2)*

MMDC_P001	MMDC_ P004_1.EN	MMDC_P101
Property ID	**Preferred name in English**	**Map to EXPRESS**
MDC_P006_1	Source document of definition	Currently not mapped For value code, it is mapped to the following. value_domain.its_values[i].source_doc_of_value
MDC_P007_1.<lang>	Note	Currently not mapped
MDC_P007_2.<lang>	Remark	Currently not mapped
MDC_P021	Definition class	Currently not mapped
MDC_P025_1	Preferred letter symbol in text	Currently not mapped If it is referenced as value code in Enumeration meta-class, it is mapped to the following. value_domain.its_values[i]. value_code
MDC_P025_2	Preferred letter symbol in SGML	Currently not mapped

Annex L
(informative)

Meta-class properties mapped with DIN 4002

Meta-properties, i.e., the properties of meta-classes, used as the constructs for the definition of classes and properties are mapped with the dictionary attribute identifiers described in DIN 4002. The aim of comparing the properties and attributes of those standards is to provide a detailed guide for those who develop a formal and automated mapping between those standards, however this annex per se does not to intend to standardize or automate it.

The following Tables L.1 to L.4 contain a list of meta-properties for the definition of a class, a property, or an enumeration mapped with DIN 4002.

Table L.1 – Meta-properties for the definition of a class or a property, mapped with DIN 4002 *(1 of 6)*

IEC 62656-1/ISO 13584-35		DIN 4002		Note
Property ID	**Preferred Name**	**ID**	**Benennung**	
MDC_P001_5	Code	C01	Kennung	
MDC_P002_1	Version number	C02	Versionsnummer	
MDC_P002_2.<lang>	Revision number	C03	Revisionsnummer	
MDC_P003_1	Date of original definition			It is not mapped to DIN 4002.
MDC_P003_2	Date of current version			It is not mapped to DIN 4002.
MDC_P003_3. <lang>	Date of current revision			It is not mapped to DIN 4002.
MDC_P004_1.<lang>	Preferred name	A01(DE)	Bevorzugte Benennung	
MDC_P004_2	Synonymous name	A02(DE)	Synonym(e)	
MDC_P004_3.<lang>	Short name	A08(DE)	Kurzbezeichnung	
MDC_P004_4	Name icon			It is not mapped to DIN 4002.
MDC_P005.<lang>	Definition	A04(DE)	Definition	
MDC_P006_1	Source document of definition	A05(DE)	Quelle der Definition	
MDC_P007_1.<lang>	Note	A06(DE)	Anmerkung	
MDC_P007_2.<lang>	Remark	A07(DE)	Kommentar	
MDC_P008	Drawing	B05	Bildidentifikator	B05 is a list of icon identifications (list of ID like strings).
MDC_P010	Superclass	C09	Zuordnung zum übergeordneten Strukturelement	

Table L.1 *(2 of 6)*

IEC 62656-1/ISO 13584-35		DIN 4002		Note
Property ID	**Preferred Name**	**ID**	**Benennung**	
MDC_P011	Class type	A13	Art des Strukturelements	The values for A13 shall be I - item, M - material, C - component, F - feature, A - classification class, I-RE - Anforderung/Requirement, I-DD - Gerätebeschreibung/Device Description, I-LO - Logistics
MDC_P012	Supplier	C08	Kurzbezeichnung des Herausgebers	MDC_P012 represents the supplier code (i.e. 112/2///61360_4_1) while C08 is supplier short name (i.e. IEC)
MDC_P013	Is case of			It is not mapped to DIN 4002.
MDC_P014	Applicable properties	A19	Liste der verwendeten Merkmale	
MDC_P015	Applicable types			It is not mapped to DIN 4002.
MDC_P016	Sub-class selection properties	A20	Liste der klassenbestimmenden Merkmale	
MDC_P017	Class value assignment	A21	Liste der Merkmale mit klassenbestimmender konstanter Ausprägung	
MDC_P018	Coded name			It is not mapped to DIN 4002.

Table L.1 *(3 of 6)*

IEC 62656-1/ISO 13584-35		DIN 4002		Note
Property ID	**Preferred Name**	**ID**	**Preferred Name**	
MDC_P090	Imported properties	A22	Liste der importierten Merkmale	
MDC_P091	Imported types			It is not mapped to DIN 4002.
MDC_P093	Imported documents			It is not mapped to DIN 4002.
MDC_P094	Applicable documents			It is not mapped to DIN 4002.
		A03(DE)	Schlagwörter	It means "keyword" in English. It is not mapped to ISO 13584-35.
		C04	Identifikator	It means “identifier” in English, and it is derived from MDC_P012, MDC_P001_6 and MDC_P002_1.
		C10	Zuordnung zu ICS	It means "ICS numbers" in English. It is not mapped to ISO 13584-35.

Table L.1 *(4 of 6)*

IEC 62656-1/ISO 13584-35		DIN 4002		Note
Property ID	**Preferred Name**	**ID**	**Benennung**	
MDC_P001_6	Code	C01	Kennung	
MDC_P002_1	Version number	C02	Versionsnummer	
MDC_P002_2.<lang>	Revision number	C03	Revisionsnummer	
MDC_P003_1	Date of original definition			Dates are not mapped because DIN 4002 was not built to exchange dictionaries but to add data into an existing dictionary.
MDC_P003_2	Date of current version			It is not mapped to DIN 4002. See the note of MDC_P003_1.
MDC_P003_3. <lang>	Date of current revision			It is not mapped to DIN 4002. See the note of MDC_P003_1.
MDC_P004_1.<lang>	Preferred name	A01(DE)	Bevorzugte Benennung	
MDC_P004_2	Synonymous name	A02(DE)	Synonym(e)	
MDC_P004_3.<lang>	Short name	A08(DE)	Kurzbezeichnung	
MDC_P004_4	Name Icon			It is not mapped to DIN 4002.
MDC_P005.<lang>	Definition	A04(DE)	Definition	
MDC_P006_1	Source document of definition	A05(DE)	Quelle der Definition	
MDC_P007_1.<lang>	Note	A06(DE)	Anmerkung	
MDC_P007_2.<lang>	Remark	A07(DE)	Kommentar	
MDC_P008	Drawing	B05	Bildidentifikator	B05 is a list of icon identifications (list of ID like strings).
MDC_P020	Property data element type	A13	Art des Strukturelements	
MDC_P021	Definition class	A14	Zuordnung zum Geltungsbereich	
MDC_P022	Data type	A15	Merkmaldatentyp	

Table L.1 *(5 of 6)*

IEC 62656-1/ISO 13584-35		DIN 4002		Note
Property ID	**Preferred Name**	**ID**	**Benennung**	
MDC_P023	Unit structure	A10	Einheit	DIN 4002 exports Iec_61360. If not present, Iso_1000 else Un_ece_code. For local units (e.g. IEC dictionary) DIN 4002 uses textRepresentation (corresponding to P023_1).
MDC_P023_1	Unit in text			See the note for MDC_P023.
MDC_P023_2	Unit in SGML			See the note for MDC_P023.
MDC_P024	Value format	A09	Werteformat	
MDC_P025_1	Preferred letter symbol in text	B01	Bevorzugtes Symbol	
MDC_P025_2	Synonymous letter symbol	B02	Ersatzsymbol	
MDC_P025_3	Preferred letter symbol in SGML			It is not mapped to DIN 4002.
MDC_P027_1	Formula in text	B03	Formel	In DIN 4002 formula is in MATHML format
MDC_P027_2	Formula in SGML			It is possible to map it to B03.
MDC_P028	Condition	B04	Abhängigkeit von Bedingungen	
MDC_P040	DET classification	B07	IEC-Klassifikation des Merkmals	
MDC_P041	Code for unit	A11	Code für Einheit	
		A03(DE)	Schlagwörter	It means "keyword" in English. It is not mapped to ISO 13584-35.
		A12	Identifikator eines globalen Merkmaldatentyps	It means "global domain Identification" in English. It is described with its data type "named_type" in MDC_P022.
		A16	Wertkodierung	It means “value Specification Type” in English. It is not mapped to ISO 13584-35.
		A17	Werteliste	It means “list of values” in English. It is described with its data type “non_quantitative_code/int_type” in MDC_P022.

Table L.1 *(6 of 6)*

IEC 62656-1/ISO 13584-35		DIN 4002		Note
Property ID	**Preferred Name**	**ID**	**Benennung**	
		B06	Identifikator für den Datentyp class_instance	It means “identification class_instance_type” in English. It is described with its data type “Class_instance_type” in MDC_P022.
		C04	Identifikator	It means “identifier” in English. It is derived from MDC_P021, MDC_P001_6 and MDC_P002_1.
		C08	Kurzbezeichnung des Herausgebers	It means "shortname of supplier" in English. It is included in MDC_P021.
		C09	Zuordnung zum übergeordneten Strukturelement	It means "superelement" in English. It is not mapped to ISO 13584-35.
		C10	Zuordnung zu ICS	It means "ICS numbers" in English. It is not mapped to ISO 13584-35.

Table L.2 – Meta-properties for the definition of an enumeration, mapped with DIN 4002 *(1 of 2)*

IEC 62656-1/ISO 13584-35		DIN 4002		Note
Property ID	**Preferred Name**	**ID**	**Benennung**	
MDC_P001_4	Enumeration code	C04	Identifikator	
MDC_P004_1.<lang>	Preferred name	A01(DE)	Bevorzugte Benennung	
MDC_P004_2	Synonymous name	A02(DE)	Synonym(e)	
MDC_P004_3.<lang>	Short name	A08(DE)	Kurzbezeichnung	
MDC_P004_4	Name icon			
MDC_P006_2	Source document of value	A05(DE)	Quelle der Definition	
MDC_P029	Definition property	A14	Zuordnung zum Geltungsbereich	MDC_P029 is now obsolete.
		A03(DE)	Schlagwörter	It means "keyword" in English. It may be mapped to MDC_P004_1 or MDC_P004_2 of terminology parcel in ISO 13584-35.
		A04(DE)	Definition	It means "definition" in English. It may be mapped to MDC_P005 of terminology parcel in ISO 13584-35.
		A06(DE)	Anmerkung	It means "note to definition" in English. It may be mapped to MDC_P007_1 of terminology parcel in ISO 13584-35.
		A07(DE)	Kommentar	It means "remark" in English. It may be mapped to MDC_P007_2 of terminology parcel in ISO 13584-35.

Table L.2 *(2 of 2)*

IEC 62656-1/ISO 13584-35		DIN 4002		Note
Property ID	**Preferred Name**	**ID**	**Benennung**	
		A18	Wert	It means "value" in English. Furhter information is needed to map into ISO 13584-35.
		B01	Bevorzugtes Symbol	It means “preferred symbol” in English. Furhter information is needed to map into ISO 13584-35.
		B02	Ersatzsymbol	It means "substitute symbol" in English. Furhter information is needed to map into ISO 13584-35.
		B03	Formel	It means "formula" in English. Furhter information is needed to map into ISO 13584-35.
		B05	Bildidentifikator	It means "Drawing" in English. Furhter information is needed to map into ISO 13584-35.
		C09	Zuordnung zum übergeordneten Strukturelement	It means "superelement" in English. Furhter information is needed to map into ISO 13584-35.

Table L.3 – Meta-properties for the definition of a data type, mapped with DIN 4002 *(1 of 2)*

IEC 62656/ISO 13584-35		DIN 4002		Note
Property ID	**Preferred Name**	**ID**	**Benennung**	
MDC_P001_7	Code	C01	Kennung	
MDC_P002_1	Version number	C02	Versionsnummer	
MDC_P002_2.<lang>	Revision number	C03	Revisionsnummer	
MDC_P003_1	Date of original definition			Dates are not mapped because DIN 4002 was not built to exchange dictionaries but to add data into an existing dictionary.
MDC_P003_2	Date of current version			It is not mapped to DIN 4002. See the note of MDC_P003_1.
MDC_P003_3.<lang>	Date of current revision			It is not mapped to DIN 4002. See the note of MDC_P003_1.
MDC_P004_1.<lang>	Preferred name	A01(DE)	Bevorzugte Benennung	
MDC_P004_2	Synonymous name	A02(DE)	Synonym(e)	
MDC_P004_3.<lang>	Short name	A08(DE)	Kurzbezeichnung	
MDC_P004_4	Name icon			
MDC_P021	Definition class	A14	Zuordnung zum Geltungsbereich	
MDC_P022	Data type	A15	Merkmaldatentyp	
MDC_P023	Unit structure	A10	Einheit	
MDC_P023_1	Unit in text			It is not mapped to DIN 4002.
MDC_P023_2	Unit in SGML			It is not mapped to DIN 4002.
MDC_P024	Value format			It is not mapped to DIN 4002.
MDC_P041	Code for unit	A11	Code für Einheit	
		A03(DE)	Schlagwörter	It means "keyword" in English. It is not mapped to ISO 13584-35.

Table L.3 *(2 of 2)*

IEC 62656/ISO 13584-35		DIN 4002		Note
Property ID	**Preferred Name**	**ID**	**Benennung**	
		A04(DE)	Definition	It means “definition” in English. It is not mapped to IEC 62656-1/ISO 13584-35.
		A05(DE)	Quelle der Definition	It means “source of definition” in English. It is not mapped to IEC 62656-1/ISO 13584-35.
		A06(DE)	Anmerkung	It is translated into “note to definition” in English. It is not mapped to IEC 62656-1/ISO 13584-35.
		A07(DE)	Kommentar	It means “remark” in English. It is not mapped to IEC 62656-1/ISO 13584-35.
		A16	Wertkodierung	It means “value specification type” in English. It is not mapped to IEC 62656-1/ISO 13584-35.
		A17	Werteliste	It means “list of values” in English. It is described with its data type “non_quantitative_code/int_type” in MDC_P022.
		B01	Bevorzugtes Symbol	It means “preferred symbol” in English. It is not mapped to IEC 62656-1/ISO 13584-35.
		B02	Ersatzsymbol	It means “substitute symbol” in English. It is not mapped to IEC 62656-1/ISO 13584-35.
		B03	Formel	It means “formula” in English. It is not mapped to IEC 62656-1/ISO 13584-35.
		C04	Identifikator	It means “identification” in English. It is derived from MDC_P021, MDC_P001_6 and MDC_P002_1.
		B05	Bildidentifikator	It means “drawing” in English. It is not mapped to IEC 62656-1/ISO 13584-35.
		C08	Kurzbezeichnung des Herausgebers	It means “shortname of supplier” in English. It is included in MDC_P021.
		C09	Zuordnung zum übergeordneten Strukturelement	It means “superelement” in English. It is included in MDC_P021.
		C10	Zuordnung zu ICS	It means "ICS numbers" in English. It is not mapped to IEC 62656-1/ISO 13584-35.

Table L.4 – Meta-properties for the definition of a UoM, mapped with DIN 4002 *(1 of 2)*

IEC 62656/ISO 13584-35		DIN 4002		Note
Property ID	**Preferred Name**	**ID**	**Benennung**	
MDC_P001_10	UoM Code	P10	Einheitenidentifikation	
MDC_P004_1.<lang>	Preferred name			It may be mapped to one of P03, P13, P14.
MDC_P004_2	Synonymous Name			It may be mapped to some of P03, P13, P14.
MDC_P004_3.<lang>	Short name	P06(DE)	Kurzbezeichnung	
MDC_P004_4	Name Icon			It is not mapped to DIN 4002.
MDC_P006_1	Source document of definition	P08(DE)	Quelle	
MDC_P023	Unit structure			It may be mapped to P01 or P02.
MDC_P023_1	Unit in text	P12	Primäreinheit	
MDC_P023_2	Unit in SGML			It may be mapped to P01 or P02.
MDC_P002_1	Version number			It is not mapped to DIN 4002.
MDC_P002_2.<lang>	Revision number			It is not mapped to DIN 4002.
MDC_P003_1	Date of original definition			Dates are not mapped because DIN 4002 was not built to exchange dictionaries but to add data into an existing dictionary.
MDC_P003_2	Date of current version			It is not mapped to DIN 4002. See the note of MDC_P003_1.
MDC_P003_3. <lang>	Date of current revision			It is not mapped to DIN 4002. See the note of MDC_P003_1.
MDC_P005.<lang>	Definition	P07(DE)	Erklärung	
MDC_P007_1.<lang>	Note	P09(DE)	Kommentar	
MDC_P007_2.<lang>	Remark			It is not mapped to DIN 4002.
MDC_P021	Definition class			It is not mapped to DIN 4002.
		P03	ECE Name	It is possible to map it to MDC_P004_1.

Table L.4 *(2 of 2)*

IEC 62656/ISO 13584-35		DIN 4002		Note
Property ID	**Preferred Name**	**ID**	**Benennung**	
		P05(DE)	strukturierte Benennung	It means “structured designation” in English. It is not mapped to IEC 62656-1/ISO 13584-35.
		P01	SI-Schreibweise	It means “SI way of writing” in English. It may be mapped to MDC_P023_x.
		P02	DIN-Schreibweise	It means “DIN way of writing” in English. It may be mapped to MDC_P023_x.
		P04	ECE Code	It means “ECE code” in English. It is not mapped to IEC 62656-1/ISO 13584-35.
		P13	SI-name	It is possible to map it to MDC_P004_1.
		P14	NIST	It is possible to map it to MDC_P004_1.
		P15	Geltungsbereich	It means “area of application” in English. It is not mapped to IEC 62656-1/ISO 13584-35.

Annex M
(informative)

Use case of relation for units and quantities

The relations and their subtypes, i.e., predication and function are the structures to build a relationship or grouping among already exisiting ontological entities. Apart from some graphic or display related properties, such as symbols or graphic functions, the relations themselves have no characterisitic properties, nor they are allowed to have properties to influence the basic nature of the items listed in the domain or codomain. Thus the structure is quite useful for inserting a relationship among existing ontological entities, such as classes, properties, data types, and units of measurement.

The following figures are designed to provide an intuitive understanding about how to use the relation meta-class together with a UoM meta-class to represent the concepts of quantity, system of units of measurement, and units of measurement in POM. It is not designed, however, to formally specify what are the mandatory attributes and what are not.

For the limitations of space in the figures, not all the units nor attributes are shown for the length quantity. Moreover all the units belonging to the mass quantity are omitted for simplifying the figures. Approximately, Figure M.3 corresponds to the UoM parcel (meta-class) in Figure M.1 plus the relation parcel (meta-class) in Figure M.2.

In Figure M.2, a quantity may be modelled as a function or a predication. However, when the “SI coherent derived unit” for the quantity is considered, it is recommended to use a function. The ID of the coherent derived unit must be mentioned in the codomain of the function. When it is not immediately available, it may be kept open (null) as shown in Figure M.2.

#PROPERTY_ID	MDC_P001_10	MDC_P004_1.en	MDC_P023_1
#PROPERTY_NAME	Code	Preferred name	Unit in text
	UAA726	metre	m
	UAA862	milimetre	mm
	UAA090	micrometre	µm
	UAA912	nanometre	nm
	UAB030	yard	yd
	UAA440	foot	ft
	UAA539	inch	in

Figure M.1 – Example of UoM meta-class for defining units for length

#PROPERTY_ID	MDC_P001_13	MDC_P004_1.en	MDC_P200	MDC_P201	MDC_P202	MDC_P203	MDC_P210	MDC_P212
#PROPERTY_NAME	Code	Preferred name	Relation type	Domain of the relation	Domain of the function	Codomain of the function	Role of the relation	Super relation
	USXX1	System of units of measurement	PRED	USXX2, USXX3			System of Units	UNIVERSE
	USXX2	SI System of units	PRED	UAD104, UAD072			System of Units	
	USXX3	Imperial System of Units	PRED	UQXX4,UQXX5			System of Units	
	UQXX1	Quantity	PRED	....			Quantity	UNIVERSE
	UQXX2	Length Measure	PRED	....			Quantity	UQXX1
	UQXX3	Mass Measure	PRED	....			Quantity	UQXX1
	UAD104	SI Mass	FUNC		....	....	Quantity	UQXX3
	UAD072	SI Length	FUNC		UAA726, UAA862, UAA090, UAA912	UAA726	Quantity	UQXX2
	UQXX4	Imperial Mass Measure	FUNC		....	(null)	Quantity	UQXX3
	UQXX5	Imperial Length Measure	FUNC [a]		UAB030, UAA440, UAA539	(null)	Quantity	UQXX2

[a] In Figure M.3, all the units that belong to the same quantity shall be listed in the domain of a function as a quantity. The coherent derived unit for imperial length measure is not considered (It can be a foot or metre). Thus, in lieu of a function, a predication could be used. However for the ease of implementation, use of a function is recommended. In this case, the codomain may be kept open (NULL).

Figure M.2 – Sample specification of the relation meta-class for quantity and system of units of measurement

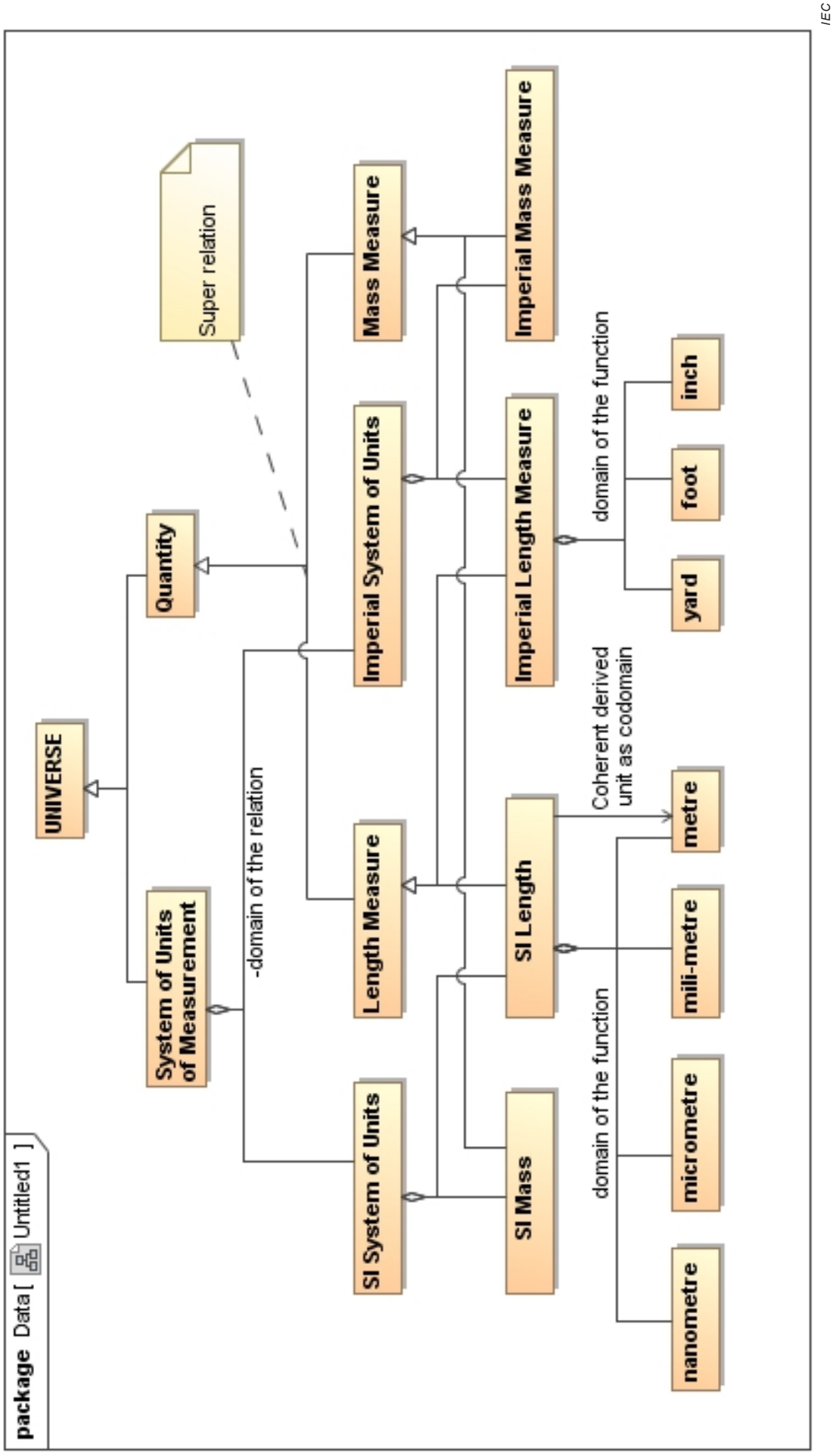


Figure M.3 – Quantity and system of units of measurement expressed as relations

Annex N
(informative)

Guide for the use of placement data types

N.1 Primitive coordinates

In the case of ISO 10303-42 standard, where the placement data types are originally defined, AXIS1_PLACEMENT_2D, AXIS1_PLACEMENT_3D, AXIS2_PLACEMENT_2D and AXIS2_PLACEMENT_3D may inherit an attribute named “location” that is an attribute of PLACEMENT type in 2D or 3D space, from a generic geometric entity named “placement”. The attribute is useful and even indispensable in defining an object shape resulting from a small number of Boolean operations (set-theoretic Boolean operations) of primitive shapes. This is achieved simply by making those placement type properties dependent on a Cartesian point that is modelled as another placement type property, usually with 3 coordinate values in real numbers.

For example, a cylinder whose top and bottom faces are covered respectively by a cone and a semisphere, can be generated from Boolean operations of a cylinder, a cone and a sphere. The cone must be displaced along the longitudinal direction (i.e., Z’-direction) of the cylinder and placed at the centre of the top face.

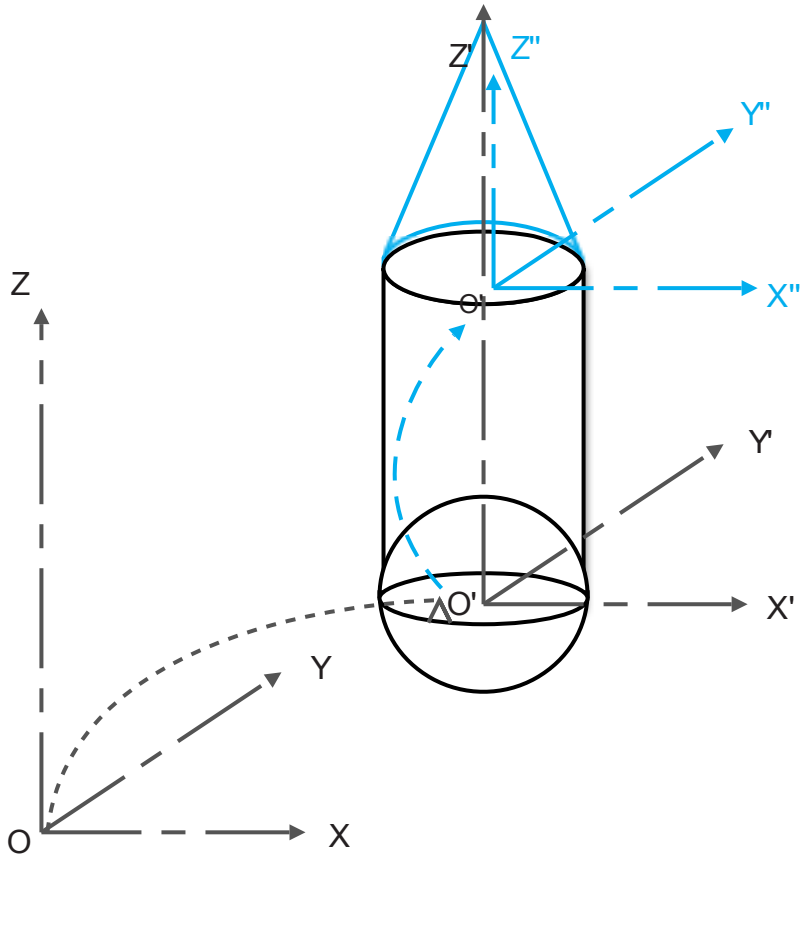


Figure N.1 – Local coordinate system and the primitive coordinates

To place a cone at the centre of the top face, it is indispensable to move the cone along Z'-direction in the local coordinate system (X'Y'Z'), as shown in Figure N.1. To achieve this effect, the cone needs to be defined in the primitive coordinates (X''Y''Z'') that are translated upward from the local coordinate system (X'Y'Z'). The resulting shape of Boolean operation (in this case a union operation is applied), can be placed anywhere in the global coordinate system (XYZ), by applying a placement (again) with Axis2_placement3D. However, in general, this last translation need not be recorded for the object shapes to be registered in IEC CDD. But this should not be confused with the local translational move of a primitive shape that is necessary for defining an object shape.

N.2 EXPRESS language codes

The following lists are the extraction from ISO 10303-42 of the EXPRESS codes that define the placement types and some of the CSG primitives. The lists are for explanation purposes only.

```
ENTITY placement
   SUPERTYPE OF (ONEOF(axis1_placement,axis2_placement_2d,axis2_placement_3d))
   SUBTYPE OF (geometric_representation_item);
   location : cartesian_point;
END_ENTITY;

ENTITY axis1_placement
   SUBTYPE OF (placement);
   axis : OPTIONAL direction;
   DERIVE
   z : direction := NVL(normalise(axis), dummy_gri ||
   direction([0.0,0.0,1.0]));
   WHERE
   WR1: SELF\geometric_representation_item.dim = 3;
END_ENTITY

ENTITY axis2_placement_3d
   SUBTYPE OF (placement);
   axis : OPTIONAL direction;
   ref_direction : OPTIONAL direction;
   DERIVE
   p : LIST [3:3] OF direction := build_axes(axis,ref_direction);
   WHERE
   WR1: SELF\placement.location.dim = 3;
   WR2: (NOT (EXISTS (axis))) OR (axis.dim = 3);
   WR3: (NOT (EXISTS (ref_direction))) OR (ref_direction.dim = 3);
   WR4: (NOT (EXISTS (axis))) OR (NOT (EXISTS (ref_direction))) OR
   (cross_product(axis,ref_direction).magnitude > 0.0);
END_ENTITY;

ENTITY cartesian_point
SUPERTYPE OF (ONEOF(cylindrical_point, polar_point, spherical_point))
SUBTYPE OF (point);
coordinates : LIST [1:3] OF length_measure;
END_ENTITY;
```

Figure N.2 – Extracts of EXPRESS codes for placement types

```
ENTITY right_circular_cylinder
   SUBTYPE OF (geometric_representation_item);
   position : axis1_placement;
   height : positive_length_measure;
   radius : positive_length_measure;
END_ENTITY;

ENTITY sphere
SUBTYPE OF (geometric_representation_item);
radius : positive_length_measure;
centre : point;
END_ENTITY;

ENTITY right_circular_cone
SUBTYPE OF (geometric_representation_item);
position : axis1_placement;
height : positive_length_measure;
radius : length_measure;
semi_angle : plane_angle_measure;
WHERE
WR1: radius >= 0.0;
END_ENTITY;
```

Figure N.3 – Extracts of EXPRESS codes for CSG primitives

Annex O
(informative)

Foundation in mathematical-logic

O.1 Class and property as sets

From a math-logical point of view, a class is a concept about a collection defined by the intention for its members to be gathered, such that,

$$C = \{x \mid P(x)\},$$

where C is a class, x is an element of the class, and $P(x)$ is a logical statement about x, called “intent” or “intention” of C, that is either true or false. The above formula signifies that only x that makes $P(x)$ true shall be gathered as a member of the class C. A class and a set only differ in that a set is assured to have member(s) (though an empty set may be regarded as having a member “nil”), while a class may not have any. If the $P(x)$ is given in a following manner, then class C is a set:

$$C = \{x \mid P(x) \wedge x \in S\}$$

Namely, $x \in S$ assures that x is already a member of a set S.

The following formula gives a definition of a class but it is not regarded as a definition of a set:

$$C = \{x \mid P(x) \wedge \neg P(x)\},$$

for such a logical statement always gives a falsitude. Namely, $P(x) \wedge \neg P(x)$ is a contradiction.

A property is a kind of class, yet always used to subdivide the members of another set into subsets (subclasses). A property as a characteristic function maps the elements of one class called “domain” into another class called “codomain”(traditionally, it has been called “range”).

Thus we can write this as $y = p(x)$ where x is an element in the domain and y is an element in the codomain, as illustrated in the following Figure O.1.

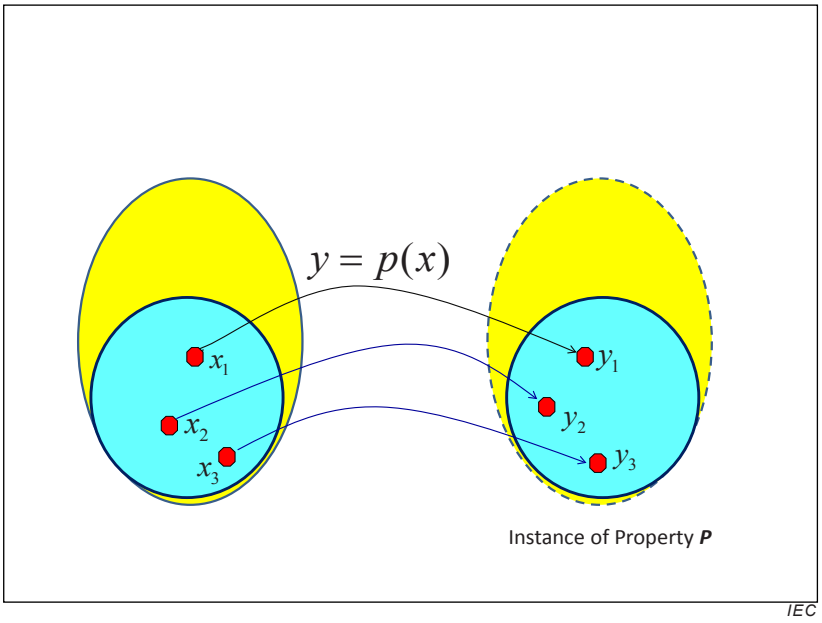


Figure O.1 – Class, property and property-value function

However this mapping from a class to a property entails a characteristic function such that $P(x, y) \rightarrow \{truth, false\}$ (i.e., $P(x, y) = T$ or $P(x, y) = F$) where the combination of x and y returns true only if x is in the subset (subclass), as illustrated in Figure O.2.

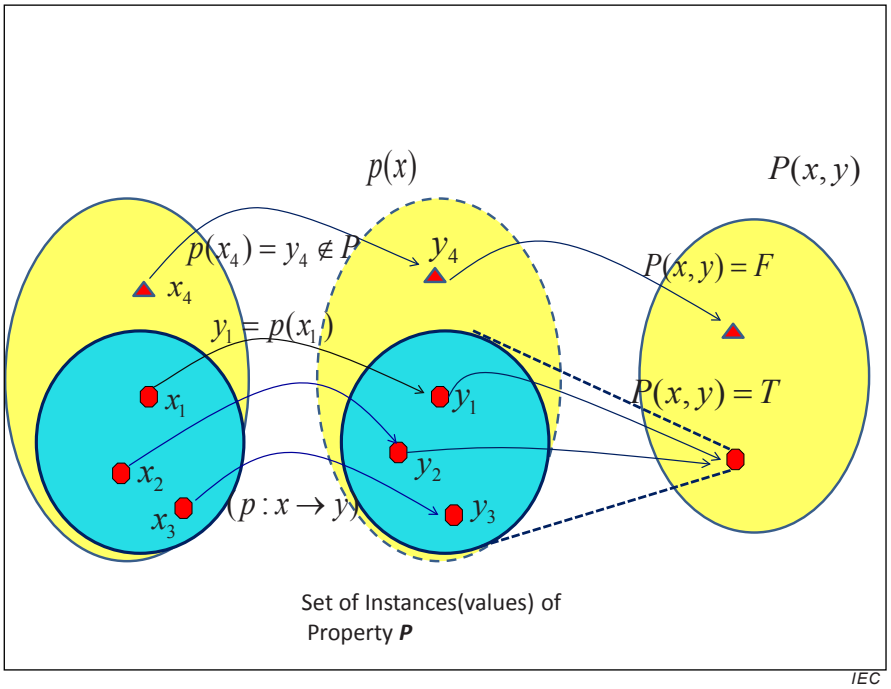


Figure O.2 – Class and Property and its characteristic function

In other words, a property is a kind of class that serves as a classifier of the elements of another class into a true or a false value according to whether it belongs to a certain subset of the latter class or not, through a process called “instantiation” of its variable (parameter).

O.2 Property specialization explained by set theory

Now we explain the notion of property specialization, using the notion of set and subset in relation to the property as a set. A property may be specialized when we can restrict the domain of the property-value function, as in Figure O.3. In the figure, the range of variable x is restricted to C_2 from C_1. As a result, the range of values of properties is restricted to P_2.

Methods of restricting the domain of a property-value function are multiple. For example, changing the definition class (i.e., the name scope) of a property to a subclass of the current class effectively restricts the domain to a subset of the original one. Adding conditions to a property is another way of limiting the candidate elements in the domain to a subset.

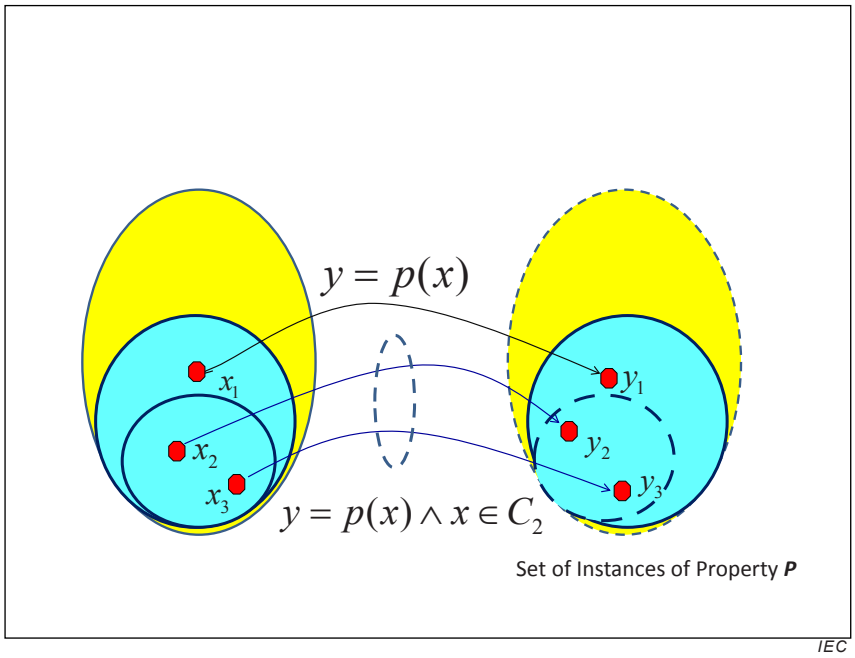


Figure O.3 – Property specialization by restriction of the domain

Note that a similar effect of specialization is achieved by directly narrowing the codomain instead of the domain, as shown in Figure O.4.

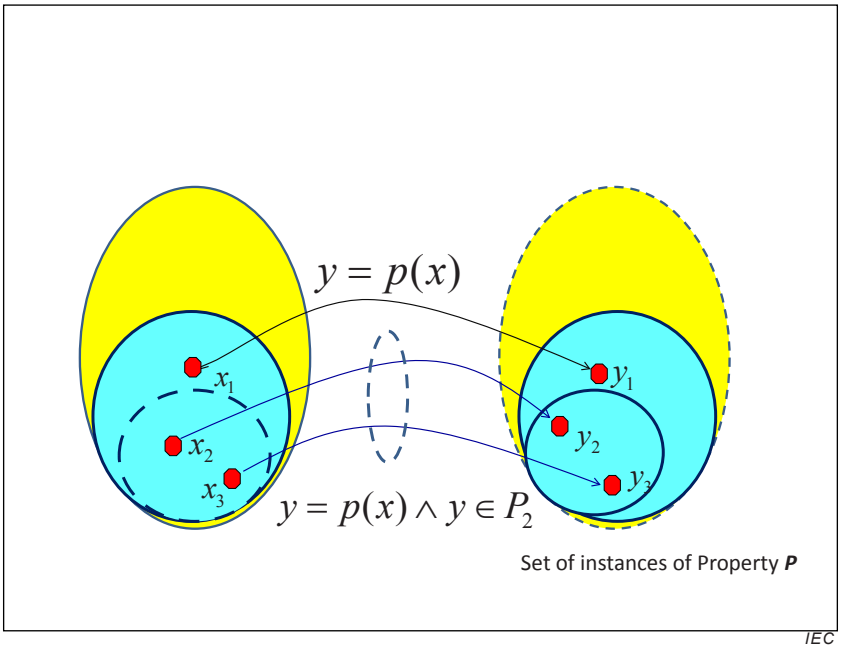


Figure O.4 – Property specialization by restriction of the codomain

It is also worthwhile to note that a property-value function is considered as a combination between some element(s) of the domain and an element in the codomain. Thus, when we regard each property-value function as an ordered set of discreet mappings between the elements, such as $p_1 = \langle x_1, y_1 \rangle$, $p_2 = \langle x_2, y_2 \rangle$, $p_3 = \langle x_3, y_3 \rangle$, we may also restrict the number of selectable property-value functions directly, as illustrated in the following Figure O.5.

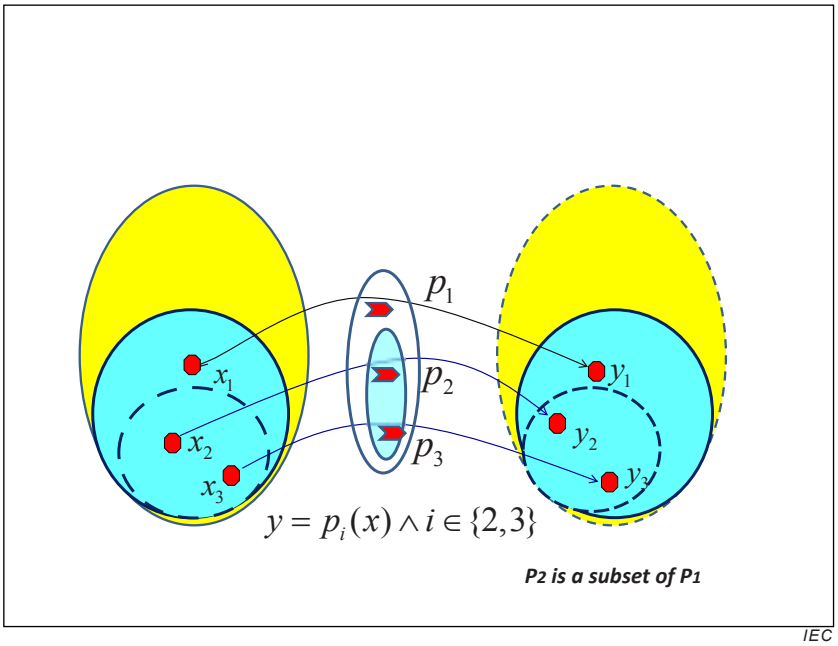


Figure O.5 – Property specialization by limiting the selectable function set

O.3 Mathematical basis of POM

The Parcellized Ontology Model or POM in short is an application of a well-known theorem in mathematical logic, i.e. the reduction theorem that plays a pivotal role in the proof of Completeness Theorem of Kurt Gödel. As we see in the body of the standard, the POM has a 4-layer ontology architecture, where each layer is based on the same model structure named parcel, starting from the Axiomatic Ontology layer at the top, then the Meta-Ontology layer (MO), being followed by the Domain Ontology layer, ending at the Domain Library layer at the bottom. Each layer is abbreviated as AO, MO, DO, and DL, respectively (see Figure O.6). Although the properties in industrial data modelling are often taking the form of informal propositions, the following sub-clauses demonstrate how deep the POM structure is modelled after the reduction theorem, and is related to the Completeness Theorem.

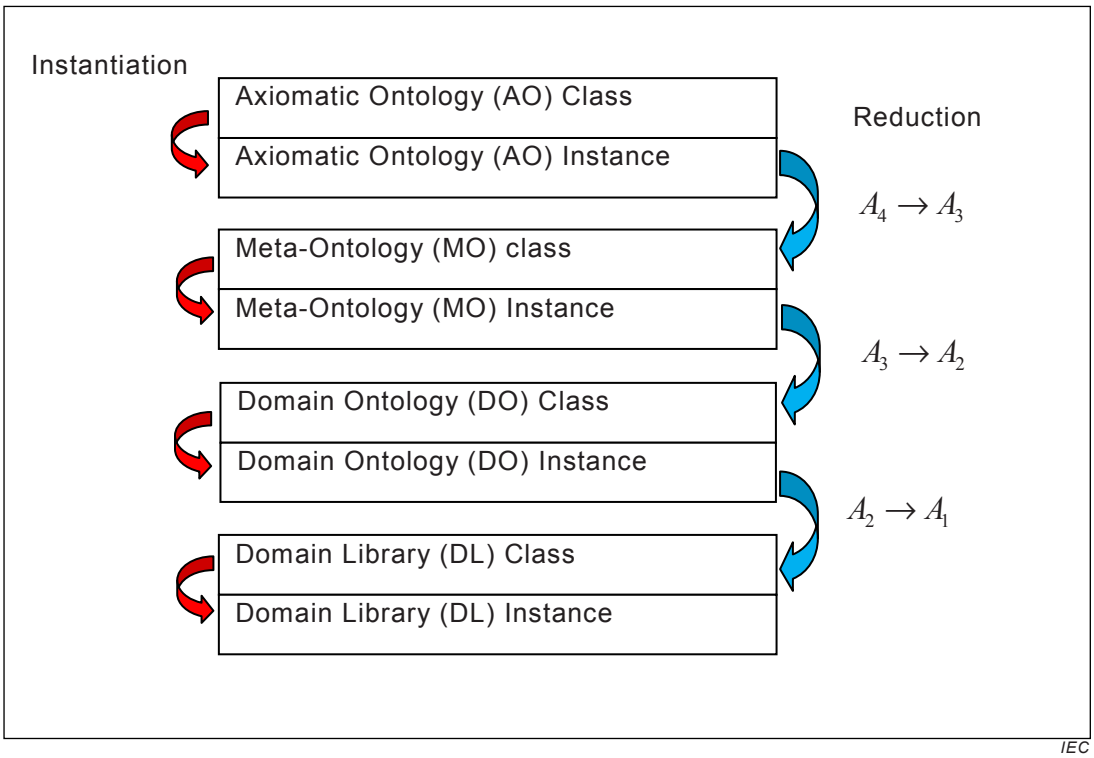


Figure O.6 – Architecture of POM

Let's say A is a logical formula containing a free variable in mathematical logic, the following theorem is known as the substitution theorem:

$$A \rightarrow \exists a A\,,$$

where A means A is assumed to be true.

Namely, if A is assumed true then there must be at least one instance a such that replacing x by a in A makes A true.

Now let a theory A^j be a logical formula, with k free variables $x=[x_1, x_2, \cdots, x_k]$. Then the formula may be rewritten as follows:

$$A^j = A^j[x_1, x_2, \cdots, x_k]\,.$$

If there is $a=[a_1,a_2,\cdots,a_k]$ such that assignment of a to x makes A^j true, then we may rewrite this as $a\ A^j$.

Applying the substitution theorem one by one to k free variables we obtain:

$$A^j \rightarrow \exists a A^j,$$

Where $\exists a$ means $\exists a_1 \exists a_2 \cdots \exists a_k$.

Now we further assume A^{j-1} is a logical formula that is made up of a finite number of logical conjunctions or disjunctions of terms where no negation is applied:

$$a_1, a_2, \cdots, a_k.$$

Note that if $a_{i\in[1,k]}$ is true, then the result of any number of logical conjunctions of a_i is true and the result of any number of logical disjunctions of a_i is true, thus A^{j-1} is true. This means

$$A^j \rightarrow \exists a A^j \longrightarrow A^{j-1}$$

Now, what we intend to assimilate to the above formula is a process to make classes and properties from the set of meta-classes of the layer immediately above.

Assume that A^{j-1} still contains some other free variables a'. Then we apply the above process to A^{j-1} and we have:

$$A^{j-1} \rightarrow \exists a' A^{j-1}$$

In a repeated application of the above process, we will have the following chain:

$$A^j \rightarrow \exists a A^j \longrightarrow \exists a' A^{j-1} \longrightarrow \cdots \longrightarrow \exists e A^1,$$

where $e=[e_1,e_2,\cdots,e_k]$ signifies a list of the values of real world objects, or constants, such as rows of instance found in DL.

#PROPERTY_ID	0140/TOPAS# P000001	0140/TOPAS# P001089.MAX	0140/TOPAS# P001089.MIN	0140/TOPAS# P000894.EN	0140/TOPAS# P000894.FR
#DATATYPE	STRING	INTEGER	INTEGER	STRING	STRING
	JIS	***1000***	***2000***	***JAPAN Corporation***	***JAPON SA***
	CEN	***20***	***23***	***FRENCH Ltd.***	***FRANÇAIS SA***

Figure O.7 – Examples of instances at DL layer

From Figure O.7, it is easy to see that the instances are made up of constants (individuals) and are no more reducible. It is also evident that all these properties are defined at the DO

layer, notably in the property meta-class. Mandatory properties are regarded as conjunctions while optional ones may be understood as disjunctions.

This chain of reductions is known by the name of Completeness Theorem of Kurt Gödel, and is not our invention. Or replacing a' with a^{j-1} in the above, we can express the same formula for $j=4$ as:

$$A^4 \rightarrow \exists a^3 A^4 \rightarrow \exists a^2 A^3 \rightarrow \exists a^1 A^2 \rightarrow \exists e A^1$$

And the users will be satisfied when they see that $\exists a^{j-1} A^j$ corresponds to either one of four layers of POM, namely:

$$\text{AO} \Leftrightarrow \exists a^3 A^4, \text{MO} \Leftrightarrow \exists a^2 A^3, \text{DO} \Leftrightarrow \exists a^1 A^2, \text{DL} \Leftrightarrow \exists e^0 A^1$$

The above process of explanation about the foundation of POM literally signifies that despite its informal presentation as a set of spreadsheets, the POM has a solid foundation in mathematical logic, and that is a reason why it has a high capacity to accommodate various ontology modelling frameworks within.

Bibliography

[1] *OMG Unified Modeling Language (OMG UML), Infrastructure,* V2.1.2, 2007-11-4, OMG (Object Management Group Inc.) , http://www.omg.org/spec/UML/2.1.2/Infrastructure/PDF

[2] *Meta Object Facility (MOF) Core Specification, OMG Available Specification Version 2.0,* OMG (Object Management Group Inc.), 2006-01-01, http://www.omg.org/spec/MOF/

[3] *Mathematical Logic, Joseph R. Shoenfield, 1967, Association for Symbolic Logic (by AK Press)*

[4] *TOPOI The categorical analysis of logic, Robert Goldblatt, 2006 Dover Publications Inc.*

[5] *Model Theory: An Introduction, David Marker, 2002, Springer*

[6] *RFC 5646: Tags for Identifying languages, 2009, The Internet Engineering Task Force (IETF)* http://tools.ietf.org/html/rfc5646#section-3.1.4

[7] IEC 61360-4, *Standard data element types with associated classification scheme for electric components – Part 4: IEC reference collection of standard data element types, and component classes.* Available from: http://std.iec.ch/iec61360

[8] IEC 61968 (all parts), *Application integration at electric utilities – System interfaces for distribution management*

[9] IEC 61970 (all parts), *Energy management system application program interface (EMS-API)*

[10] ISO ICS:2005*, International Classification for Standards (ICS)*

[11] ISO 10303 (all parts), *Industrial automation systems and integration – Product data representation and exchange*

[12] ISO 13584 (all parts), *Industrial automation systems and integration – Parts library*

[13] ISO 29002 (all parts), *Industrial automation systems and integration – Exchange of characteristic data*

[14] ISO 80000 (all parts), *Quantities and units*

[15] ISO/IEC 11179 (all parts), *Information technology – Metadata registries (MDR)*

[16] ISO/IEC 29500 (all parts), *Information technology — Document description and processing languages — Office Open XML File Formats*

[17] DIN 4002 (all parts), *Properties and their scopes for product data exchange* (available in German only)

———————

This page deliberately left blank

This page deliberately left blank

British Standards Institution (BSI)

BSI is the national body responsible for preparing British Standards and other standards-related publications, information and services.

BSI is incorporated by Royal Charter. British Standards and other standardization products are published by BSI Standards Limited.

About us

We bring together business, industry, government, consumers, innovators and others to shape their combined experience and expertise into standards -based solutions.

The knowledge embodied in our standards has been carefully assembled in a dependable format and refined through our open consultation process. Organizations of all sizes and across all sectors choose standards to help them achieve their goals.

Information on standards

We can provide you with the knowledge that your organization needs to succeed. Find out more about British Standards by visiting our website at bsigroup.com/standards or contacting our Customer Services team or Knowledge Centre.

Buying standards

You can buy and download PDF versions of BSI publications, including British and adopted European and international standards, through our website at bsigroup.com/shop, where hard copies can also be purchased.

If you need international and foreign standards from other Standards Development Organizations, hard copies can be ordered from our Customer Services team.

Subscriptions

Our range of subscription services are designed to make using standards easier for you. For further information on our subscription products go to bsigroup.com/subscriptions.

With **British Standards Online (BSOL)** you'll have instant access to over 55,000 British and adopted European and international standards from your desktop. It's available 24/7 and is refreshed daily so you'll always be up to date.

You can keep in touch with standards developments and receive substantial discounts on the purchase price of standards, both in single copy and subscription format, by becoming a **BSI Subscribing Member**.

PLUS is an updating service exclusive to BSI Subscribing Members. You will automatically receive the latest hard copy of your standards when they're revised or replaced.

To find out more about becoming a BSI Subscribing Member and the benefits of membership, please visit bsigroup.com/shop.

With a **Multi-User Network Licence (MUNL)** you are able to host standards publications on your intranet. Licences can cover as few or as many users as you wish. With updates supplied as soon as they're available, you can be sure your documentation is current. For further information, email bsmusales@bsigroup.com.

BSI Group Headquarters

389 Chiswick High Road London W4 4AL UK

Revisions

Our British Standards and other publications are updated by amendment or revision.

We continually improve the quality of our products and services to benefit your business. If you find an inaccuracy or ambiguity within a British Standard or other BSI publication please inform the Knowledge Centre.

Copyright

Useful Contacts:

Customer Services
Tel: +44 845 086 9001
Email (orders): orders@bsigroup.com
Email (enquiries): cservices@bsigroup.com

Subscriptions
Tel: +44 845 086 9001
Email: subscriptions@bsigroup.com

Knowledge Centre
Tel: +44 20 8996 7004
Email: knowledgecentre@bsigroup.com

Copyright & Licensing
Tel: +44 20 8996 7070
Email: copyright@bsigroup.com

bsi.

...making excellence a habit.™