

INTERNATIONAL STANDARD

NORME INTERNATIONALE



**Standardized product ontology register and transfer by spreadsheets –
Part 3: Interface for Common Information Model**

**Enregistrement d'ontologie de produits normalisés et transfert par tableurs –
Partie 3: Interface pour un modèle d'information commun**

IECNORM.COM : Click to view the full PDF of IEC 62656-3:2015



THIS PUBLICATION IS COPYRIGHT PROTECTED

Copyright © 2015 IEC, Geneva, Switzerland

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from either IEC or IEC's member National Committee in the country of the requester. If you have any questions about IEC copyright or have an enquiry about obtaining additional rights to this publication, please contact the address below or your local IEC member National Committee for further information.

Droits de reproduction réservés. Sauf indication contraire, aucune partie de cette publication ne peut être reproduite ni utilisée sous quelque forme que ce soit et par aucun procédé, électronique ou mécanique, y compris la photocopie et les microfilms, sans l'accord écrit de l'IEC ou du Comité national de l'IEC du pays du demandeur. Si vous avez des questions sur le copyright de l'IEC ou si vous désirez obtenir des droits supplémentaires sur cette publication, utilisez les coordonnées ci-après ou contactez le Comité national de l'IEC de votre pays de résidence.

IEC Central Office
3, rue de Varembe
CH-1211 Geneva 20
Switzerland

Tel.: +41 22 919 02 11
Fax: +41 22 919 03 00
info@iec.ch
www.iec.ch

About the IEC

The International Electrotechnical Commission (IEC) is the leading global organization that prepares and publishes International Standards for all electrical, electronic and related technologies.

About IEC publications

The technical content of IEC publications is kept under constant review by the IEC. Please make sure that you have the latest edition, a corrigenda or an amendment might have been published.

IEC Catalogue - webstore.iec.ch/catalogue

The stand-alone application for consulting the entire bibliographical information on IEC International Standards, Technical Specifications, Technical Reports and other documents. Available for PC, Mac OS, Android Tablets and iPad.

IEC publications search - www.iec.ch/searchpub

The advanced search enables to find IEC publications by a variety of criteria (reference number, text, technical committee,...). It also gives information on projects, replaced and withdrawn publications.

IEC Just Published - webstore.iec.ch/justpublished

Stay up to date on all new IEC publications. Just Published details all new publications released. Available online and also once a month by email.

Electropedia - www.electropedia.org

The world's leading online dictionary of electronic and electrical terms containing more than 30 000 terms and definitions in English and French, with equivalent terms in 15 additional languages. Also known as the International Electrotechnical Vocabulary (IEV) online.

IEC Glossary - std.iec.ch/glossary

More than 60 000 electrotechnical terminology entries in English and French extracted from the Terms and Definitions clause of IEC publications issued since 2002. Some entries have been collected from earlier publications of IEC TC 37, 77, 86 and CISPR.

IEC Customer Service Centre - webstore.iec.ch/csc

If you wish to give us your feedback on this publication or need further assistance, please contact the Customer Service Centre: csc@iec.ch.

A propos de l'IEC

La Commission Electrotechnique Internationale (IEC) est la première organisation mondiale qui élabore et publie des Normes internationales pour tout ce qui a trait à l'électricité, à l'électronique et aux technologies apparentées.

A propos des publications IEC

Le contenu technique des publications IEC est constamment revu. Veuillez vous assurer que vous possédez l'édition la plus récente, un corrigendum ou amendement peut avoir été publié.

Catalogue IEC - webstore.iec.ch/catalogue

Application autonome pour consulter tous les renseignements bibliographiques sur les Normes internationales, Spécifications techniques, Rapports techniques et autres documents de l'IEC. Disponible pour PC, Mac OS, tablettes Android et iPad.

Recherche de publications IEC - www.iec.ch/searchpub

La recherche avancée permet de trouver des publications IEC en utilisant différents critères (numéro de référence, texte, comité d'études,...). Elle donne aussi des informations sur les projets et les publications remplacées ou retirées.

IEC Just Published - webstore.iec.ch/justpublished

Restez informé sur les nouvelles publications IEC. Just Published détaille les nouvelles publications parues. Disponible en ligne et aussi une fois par mois par email.

Electropedia - www.electropedia.org

Le premier dictionnaire en ligne de termes électroniques et électriques. Il contient plus de 30 000 termes et définitions en anglais et en français, ainsi que les termes équivalents dans 15 langues additionnelles. Egalement appelé Vocabulaire Electrotechnique International (IEV) en ligne.

Glossaire IEC - std.iec.ch/glossary

Plus de 60 000 entrées terminologiques électrotechniques, en anglais et en français, extraites des articles Termes et Définitions des publications IEC parues depuis 2002. Plus certaines entrées antérieures extraites des publications des CE 37, 77, 86 et CISPR de l'IEC.

Service Clients - webstore.iec.ch/csc

Si vous désirez nous donner des commentaires sur cette publication ou si vous avez des questions contactez-nous: csc@iec.ch.

INTERNATIONAL STANDARD

NORME INTERNATIONALE



**Standardized product ontology register and transfer by spreadsheets –
Part 3: Interface for Common Information Model**

**Enregistrement d'ontologie de produits normalisés et transfert par tableurs –
Partie 3: Interface pour un modèle d'information commun**

INTERNATIONAL
ELECTROTECHNICAL
COMMISSION

COMMISSION
ELECTROTECHNIQUE
INTERNATIONALE

ICS 01.040.01; 01.110

ISBN 978-2-8322-2258-4

**Warning! Make sure that you obtained this publication from an authorized distributor.
Attention! Veuillez vous assurer que vous avez obtenu cette publication via un distributeur agréé.**

CONTENTS

FOREWORD	7
INTRODUCTION	9
1 Scope	10
2 Normative references	10
3 Terms, definitions and abbreviations	11
3.1 Terms and definitions	11
3.2 Abbreviations	13
4 Generic rules and principles for the design of CIM interface	13
4.1 Comparative analysis of the structures and functionalities of POM and CIM	13
4.2 CIM standard series	14
5 Rules of transformation from CIM to POM	15
5.1 General rules	15
5.2 CIM Class	16
5.2.1 General	16
5.2.2 Name	17
5.2.3 Description	17
5.2.4 Generalization-specialization	17
5.3 CIM class attributes	18
5.3.1 General	18
5.3.2 Name	18
5.3.3 Native attribute	18
5.3.4 Type	18
5.3.5 Description	18
5.4 CIM association	19
5.4.1 General	19
5.4.2 CIM association end structure	19
5.4.3 General transformation rule for association	20
5.4.4 General transformation rule for association end	20
5.5 CIM Enumeration class	24
5.5.1 General	24
5.5.2 Name	25
5.5.3 Description	25
5.6 CIM Enumeration Attribute	25
5.6.1 General	25
5.6.2 Name	25
5.6.3 Description	25
5.7 Data type	26
5.7.1 CIM basic datatype	26
5.7.2 CIM named data types	26
5.8 Package	28
5.8.1 General	28
5.8.2 POM relation object	30
5.8.3 CIM packages described in POM relation	31
6 CIM description in IEC 62656 format	33
6.1 General	33
6.2 Class	34

6.2.1	General	34
6.2.2	Code	34
6.2.3	Version number	34
6.2.4	Revision number	35
6.2.5	Date of original definition	35
6.2.6	Date of current version	35
6.2.7	Preferred name	35
6.2.8	Short name	36
6.2.9	Definition	36
6.2.10	Note	36
6.2.11	Superclass	36
6.2.12	Class type	37
6.2.13	Alternate ID	37
6.2.14	CIM package	37
6.2.15	Applicable properties	38
6.3	Property	38
6.3.1	General	38
6.3.2	Property data element type	38
6.3.3	Definition class	38
6.3.4	Data type	39
6.3.5	Applicable relations	39
6.3.6	Condition	39
6.3.7	Unit in text	39
6.3.8	Alternative units	40
6.4	Data type	40
6.4.1	General	40
6.4.2	Definition class	40
6.4.3	Unit in text	40
6.4.4	Alternative units	41
6.4.5	Data type	41
6.5	Enumeration	41
6.5.1	General	41
6.5.2	Definition class	41
6.5.3	Enumerated list of terms	41
6.5.4	Enumerate code list	42
6.6	Term meta-class	42
6.6.1	General	42
6.6.2	Preferred letter symbol in text	42
6.7	Relation	42
6.7.1	General	42
6.7.2	Relation type	43
6.7.3	Domain of the relation	43
6.7.4	Domain of the function	43
6.7.5	Codomain of the function	43
6.7.6	Domain element type	44
6.7.7	Role of the relation	44
7	XML schema for updated CIM instances in IEC CDD	44
7.1	General	44
7.2	Principles	45

8	Version control mechanism in POM for the parcellized CIM	46
Annex A (normative)	Mapping rules from CIM to POM – Mapping rules described in POM relation meta-class	47
Annex B (normative)	CIM in IEC 62656 format – Normative meta-properties of the CIM in IEC 62656 model	56
Annex C (informative)	XML Schema for parcellized CIM	63
C.1	XML schema for POM models	63
C.2	XML schema for a class	63
C.2.1	General	63
C.2.2	class.xsd	63
C.2.3	class1.xsd	64
C.2.4	class2.xsd	64
C.2.5	class3.xsd	64
C.2.6	class4.xsd	65
C.3	XML schema for a property	65
C.3.1	General	65
C.3.2	property.xsd	65
C.3.3	property1.xsd	66
C.3.4	property2.xsd	66
C.3.5	property3.xsd	67
C.3.6	property4.xsd	67
C.4	XML schema for a property derived from an association end	68
C.4.1	General	68
C.4.2	association.xsd	68
C.4.3	association1.xsd	68
C.4.4	association2.xsd	69
C.4.5	association3.xsd	70
C.4.6	association4.xsd	70
C.5	XML schema for a data type	70
C.5.1	General	70
C.5.2	datatype.xsd	71
C.5.3	datatype1.xsd	71
C.5.4	datatype2.xsd	71
C.5.5	datatype3.xsd	72
C.5.6	datatype4.xsd	72
C.6	XML schema for an enumeration and its enumerators	73
C.6.1	General	73
C.6.2	enum_term.xsd	73
C.6.3	enum_term1.xsd	73
C.6.4	enum_term2.xsd	74
C.6.5	enum_term3.xsd	74
C.6.6	enum_term4.xsd	75
C.7	XML schema for a CIM package	75
C.7.1	General	75
C.7.2	package.xsd example	75
C.7.3	package1.xsd example	76
C.7.4	package2.xsd example	76
C.7.5	package3.xsd example	76
C.7.6	package4.xsd example	77

Annex D (informative) XML examples for parcellized CIM	78
D.1 XML example for a CIM class with extended attributes	78
D.2 XML example for a property with extended attributes	78
D.3 XML example for a CIM association and aggregation with extended attributes	79
D.4 XML example for a CIM data type with extended attributes	81
D.5 XML example for a CIM enumeration and its enumerators with extended attributes	81
D.6 XML example for a CIM package with extended attributes	82
Annex E (informative) Property reference between CIM and IEC CDD – “Power transformer” definitions in both IEC CDD and CIM	84
Annex F (informative) Four Layer architecture of UML and POM – Four layer architecture of MOF (meta object facility)	87
Annex G (informative) POM Relation usage in IEC 62656-3	88
G.1 General	88
G.2 Meta-model mapping rules between UML and POM(M2)	88
G.3 UML Association(M1)	88
G.4 Model level relation definition(M1)	88
Annex H (informative) Parcellized CIM files access – URL for the parcellized CIM files	89
Bibliography	90
Figure 1 – CIM generalization example (from Figure 2 of IEC 61970-301:2013)	17
Figure 2 – Simple association example (from Figure 3 of IEC 61970-301:2013)	21
Figure 3 – Aggregation example (from Figure 4 of IEC 61970-301:2013)	21
Figure 4 – Property data element type definition	23
Figure 5 – CIM package example (from Figure 1 of IEC 61970-301:2013)	30
Figure 6 – POM relation model explanation in UML	31
Figure 7 – Class instances of parcellized CIM	45
Figure 8 – Conceptual reference mechanism with ID in POM	46
Table 1 – Mapping of CIM objects and POM objects	15
Table 2 – Correspondence between a CIM class in UML and a POM class	17
Table 3 – Mapping of CIM class attribute and POM property	18
Table 4 – CIM association end structure	19
Table 5 – CIM associations defined as POM relations	20
Table 6 – Mapping of CIM association end and POM property	21
Table 7 – CIM association ends described as POM properties	22
Table 8 – Mapping of CIM enumeration and POM enumeration	25
Table 9 – Mapping of CIM enumeration attribute and POM term	25
Table 10 – Mapping of CIM basic data type and POM simple data type	26
Table 11 – Mapping of CIM data type and a named data type in POM	27
Table 12 – Mapping of CIM package and POM relation	29
Table 13 – CIM package described as POM relation	32
Table A.1 – Mapping rule from CIM class to POM class	48
Table A.2 – Mapping rule from CIM class attribute to POM property	49
Table A.3 – Mapping rule from CIM association to POM relation	50

Table A.4 – Mapping rule from CIM association end to POM property	51
Table A.5 – Mapping rule between CIM enumeration and POM enumeration.....	52
Table A.6 – Mapping rule between CIM enumeration attribute and POM term	52
Table A.7 – Mapping rule between CIM data type and POM data type	53
Table A.8 – Mapping rule between CIM Package and POM relation	54
Table A.9 – Mapping rule from POM property to CIM aggregation/association/class attribute	55
Table B.1 – Meta-properties for class meta-class	57
Table B.2 – Meta-properties for property meta-class	58
Table B.3 –Meta-properties for datatype meta-class	59
Table B.4 – Meta-properties for enumeration meta-class	60
Table B.5 – Meta-properties for term meta-class	61
Table B.6 – Meta-properties for relation meta-class	62
Table C.1 – POM models and XML schemas.....	63
Table E.1 – Native properties of Power transformer class in the IEC ODD	85
Table E.2 – Native properties of Power transformer class in IEC 61970-301:2013	86

IECNORM.COM : Click to view the full PDF of IEC 62656-3:2015

INTERNATIONAL ELECTROTECHNICAL COMMISSION

**STANDARDIZED PRODUCT ONTOLOGY REGISTER
AND TRANSFER BY SPREADSHEETS –****Part 3: Interface for Common Information Model**

FOREWORD

- 1) The International Electrotechnical Commission (IEC) is a worldwide organization for standardization comprising all national electrotechnical committees (IEC National Committees). The object of IEC is to promote international co-operation on all questions concerning standardization in the electrical and electronic fields. To this end and in addition to other activities, IEC publishes International Standards, Technical Specifications, Technical Reports, Publicly Available Specifications (PAS) and Guides (hereafter referred to as "IEC Publication(s)"). Their preparation is entrusted to technical committees; any IEC National Committee interested in the subject dealt with may participate in this preparatory work. International governmental and non-governmental organizations liaising with the IEC also participate in this preparation. IEC collaborates closely with the International Organization for Standardization (ISO) in accordance with conditions determined by agreement between the two organizations.
- 2) The formal decisions or agreements of IEC on technical matters express, as nearly as possible, an international consensus of opinion on the relevant subjects since each technical committee has representation from all interested IEC National Committees.
- 3) IEC Publications have the form of recommendations for international use and are accepted by IEC National Committees in that sense. While all reasonable efforts are made to ensure that the technical content of IEC Publications is accurate, IEC cannot be held responsible for the way in which they are used or for any misinterpretation by any end user.
- 4) In order to promote international uniformity, IEC National Committees undertake to apply IEC Publications transparently to the maximum extent possible in their national and regional publications. Any divergence between any IEC Publication and the corresponding national or regional publication shall be clearly indicated in the latter.
- 5) IEC itself does not provide any attestation of conformity. Independent certification bodies provide conformity assessment services and, in some areas, access to IEC marks of conformity. IEC is not responsible for any services carried out by independent certification bodies.
- 6) All users should ensure that they have the latest edition of this publication.
- 7) No liability shall attach to IEC or its directors, employees, servants or agents including individual experts and members of its technical committees and IEC National Committees for any personal injury, property damage or other damage of any nature whatsoever, whether direct or indirect, or for costs (including legal fees) and expenses arising out of the publication, use of, or reliance upon, this IEC Publication or any other IEC Publications.
- 8) Attention is drawn to the Normative references cited in this publication. Use of the referenced publications is indispensable for the correct application of this publication.
- 9) Attention is drawn to the possibility that some of the elements of this IEC Publication may be the subject of patent rights. IEC shall not be held responsible for identifying any or all such patent rights.

International Standard IEC 62656-3 has been prepared by subcommittee 3D, Product properties and classes and their identification, of IEC technical committee 3: Information structures, documentation and graphical symbols.

The text of this standard is based on the following documents:

FDIS	Report on voting
3D/234/FDIS	3D/245/RVD

Full information on the voting for the approval of this standard can be found in the report on voting indicated in the above table.

This publication has been drafted in accordance with the ISO/IEC Directives, Part 2.

A list of all parts in the IEC 62656 series, published under the general title *Standardized product ontology register and transfer by spreadsheets*, can be found on the IEC website.

The committee has decided that the contents of this publication will remain unchanged until the stability date indicated on the IEC website under "<http://webstore.iec.ch>" in the data related to the specific publication. At this date, the publication will be

- reconfirmed,
- withdrawn,
- replaced by a revised edition, or
- amended.

IMPORTANT – The 'colour inside' logo on the cover page of this publication indicates that it contains colours which are considered to be useful for the correct understanding of its contents. Users should therefore print this document using a colour printer.

IECNORM.COM : Click to view the full PDF of IEC 62656-3:2015

INTRODUCTION

The IEC 62656 series, *Standardized product ontology register and transfer by spreadsheets* is a series of International standards that collectively define the methods for transferring and registering the ontologies of various products and services to and from the ontology registries and applications based on IEC 61360 / ISO13584 common data dictionary model. The IEC common data dictionary, or IEC CDD for short, is one of such registries maintained online as an IEC 61360-4 International Standard based on IEC database procedure stipulated in ISO/IEC Directives Supplement – Procedures specific to IEC. The IEC CDD is a cross-domain data dictionary covering all electro-technical products and services, maintained and updated through a database administered by IEC Central Office.

The common information model originally defined in IEC 61968 and IEC 61970 series of standards, often called by its short name “CIM” provides a standard way to represent all the major objects in an electric utility enterprise typically needed to model the operational aspects of a utility. This model includes public classes and attributes for these objects, as well as the relationships between them. It is known as an information model for energy management system (EMS) of power grids and currently is recognized as a standard ontology model for smart grids. An ontology specification conformant to the CIM data model is available in UML format according to IEC 61970-301, and in RDF format according to IEC 61970-501.

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

- Part 1: Logical structure for data parcels;
- Part 2: Application guide for use with the IEC common data dictionary (CDD);
- Part 3: Interface for common information model.

STANDARDIZED PRODUCT ONTOLOGY REGISTER AND TRANSFER BY SPREADSHEETS –

Part 3: Interface for Common Information Model

1 Scope

This part of IEC 62656 specifies an interface between IEC 62656 series and meta-model for CIM originally defined in IEC 61968 and IEC 61970 series of standards. The current CIM includes IEC 62325 series and the interface specified in this part of IEC 62656 also applies to the model defined in IEC 62325-301:2014. More specifically, this standard defines a formal mapping between the IEC 62656 and meta-model for CIM in order to import the CIM ontology into the IEC CDD, and to ensure the interoperability of ontologies of two standards, or even among a wider spectrum of standards. For the basis of the mapping from CIM to the data model defined in IEC 62656-1, the UML representation of CIM is referenced.

As a result of the interface specification available from this part of IEC 62656, the smart grid ontology defined in CIM becomes accessible and interoperable in the midst of ontology pieces originating in other ontology standards, encompassing material, environmental, and mechanical, and logistic domains of information. In addition, the specification may also cover the lifecycle of products.

This part of IEC 62656 also defines methods for transforming the IEC CDD content into the format defined in IEC 61968 and IEC 61970 series. However, this standard does not intend any standardization of the elements defined in IEC CDD as part of the CIM ontology, for all the definitions of IEC CDD are already part of an acknowledged horizontal International Standard known as IEC 61360-4-DB. Rather it makes the pieces of information stored in IEC CDD available for further standardization or customization in power electric domains or more specific user communities as a basic resource, in particular for the purpose of extending the CIM series of standards.

The data model of this part is based on IEC 62656-1 which specifies the logical structure for a data parcel, or just simply “parcel,” that is a short name for the medium of registering or transferring of product ontology. Such a parcel may be typically implemented in a leaf of a spreadsheet. The data model built on a set of parcels is called “parcellized ontology model”, which is often abbreviated as “POM” in the remainder of this document.

This part of IEC 62656 specifies:

- the mapping rules and principles for importing CIM UML objects into POM;
- the syntactic and semantic requirements on the parcellized CIM objects;
- the piecewise version control mechanism in POM for the parcellized CIM objects.

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

- definition of the interface between CIM UML and CIM RDF;
- specification of syntactic and semantic requirements on CIM UML;
- specification of syntactic and semantic requirements on CIM RDF.

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 62656-1:2014, *Standardized product ontology register and transfer by spreadsheets – Part 1: Logical structure for data parcels*

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

IEC 61360-4, *Standard data element types with associated classification scheme for electric components – Part 4: IEC reference collection of standard data element types, component classes and terms*, available from <<http://std.iec.ch/iec61360>>

IEC 61968-1:2003, *System interfaces for distribution management – Part 1: Interface architecture and general requirements*

IEC 61968-11:2010, *System interfaces for distribution management – Part 11: Common information model (CIM) extensions for distribution*

IEC 61970-1:2005, *Energy management system application program interface (EMS-API) – Part 1: Guidelines and general requirements*

IEC 61970-2:2004, *Energy management system application program interface (EMS-API) – Part 2: Glossary*

IEC 61970-301:2013, *Energy management system application program interface (EMS-API) – Part 301: Common Information Model(CIM) base*

IEC 61970-501:2006, *Energy management system application program interface(EMS-API) – Part 501: Common Information Model Resource Description Framework(CIM RDF) schema*

IEC 62325-301:2014, *Framework for energy market communications – Part 301: Common information model (CIM) extensions for markets*

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

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

ISO/IEC Directives Supplement:2013, *Procedures specific to IEC*

3 Terms, definitions and abbreviations

3.1 Terms and definitions

For the purpose of this document, the terms and definitions given in IEC 61970-2 and IEC 62656-1, as well as the following apply.

3.1.1 attribute

- a) attribute defined in IEC 62656-1, equivalent to a meta-property
- b) attribute of class used in IEC 61970-301 corresponding to a property in POM
- c) attribute of enumeration used in IEC 61970-301 corresponding to a term in POM

Note 1 to entry: In case of b), it is more specifically called as “CIM attribute” or “CIM class attribute” in this standard

3.1.2

aggregationKind

kind of the UML association comprising normal association, aggregation and composition.

Note 1 to entry: Possible value of aggregationKind is “none” for normal association, “shared” for aggregation and “composition” for a composition, the same as ISO/IEC 19505-2:2012 definitions.

3.1.3

CIM multiplicity at the source

multiplicity of the instances of the class located at the source side of the two classes connected by an association

3.1.4

CIM basic data type

basic data type defined in the CIM comprising String, Boolean, Float and Integer

Note 1 to entry: In CIM edition, there are other basic data types available for use

3.1.5

CIM object

class, attribute, package, or relationship defined in series of IEC 61970, IEC 61968 and IEC 62325 series to represent power system resources

3.1.6

CIM package

general-purpose means of grouping related model elements of CIM

Note 1 to entry: CIM Package is a synonymous name assigned to the Segment(MDC_P211) attribute of IEC 62656-1:2014

3.1.7

CIM UML ID

ID of each CIM object, assigned by a UML tool by which the object is designed

3.1.8

INVERSIBLE_P_DET INVERSIBLE_PROPERTY

kind of property data element type that can depend on the same type of property

Note 1 to entry: A UML association with unspecified navigability is defined with INVERSIBLE_P_DET because the navigability may be assigned in the future

3.1.9

navigability

state of accessibility of the instances of a class at one end of an association from the instances of the class at the other end of the association.

Note 1 to entry: In this standard, the navigability of bidirectional, unidirectional and unspecified is distinguished.

3.1.10

parcellized CIM

collection of CIM objects translated into and represented by a set of spreadsheets conformant to the POM objects defined in IEC 62656-1, with an extended list of attributes added in this part of the standard

3.1.11

POM object

set of objects defined by a meta-class of IEC 62656-1:2014 and used in this part of IEC 62656.

EXAMPLE Class, property, datatype, enumeration, unit, term and relation.

3.1.12

property

- a) instance of property meta-class, or the meta-class per se as schema; defined in IEC 62656-1
- b) property defined in IEC 61970-2 that represents a specific aspect, characteristic, attribute or relation used to describe a network resource

Note 1 to entry: To clearly distinguish between these two cases, in case of a), it is called “POM property” or “property in POM”. In case of b), it is called “CIM property” in this part of IEC 62656; whenever necessary.

3.2 Abbreviations

For the purposes of this standing document, the following abbreviations apply.

CIM	common information model
IEC CDD	IEC common data dictionary
POM	parcellized ontology model
RDF	resource description framework
RDFS	resource description framework schema
UML	unified modeling language
XML	extensible markup language

4 Generic rules and principles for the design of CIM interface

4.1 Comparative analysis of the structures and functionalities of POM and CIM

The current CIM is modelled with the UML which is widely accepted among IT engineers for designing concepts of products and services. Thus UML is used as a convenient design tool to model the CIM. However, from the users' perspective, the current CIM modelled in UML is totally devoid of mechanism to locally update or extend the entities of the model on an entity-by-entity basis, while the POM is adequately equipped to record the history of changes in a continuous manner. In other words, the current CIM standard is designed to update the model as a whole in a file at once. However, this poses a significant obstacle to the maintenance of the model as an ontology in a database and to the updates of the elements of the database on an individual basis. Since every update shall entail a holistic change of the model, and such a change will disrupt the operation of the objects already installed and used in various applications and systems.

To address this concern, an advanced framework for ontology maintenance and change management for the elements of the ontology in a database form is required, which shall be equipped with:

- a) ontology evolution management
- b) ontology extension mechanism
- c) hierarchy management for both the specialisation (is-a) and composition (has-a).

The above three requirements are fully met by POM, while the CIM in UML or in RDF is designed for other requirements and objectives.

Moreover, by scope, the CIM concentrates on information exchange among electric-power related systems and products, thus the information covered by IEC 61968, IEC 61970 and IEC 62325 series is not complete for describing the whole range and life-cycle of the products and services used in the power grids: Namely, manufacturing, installation, maintenance, transport, and disposal related characteristics of the products and services are not covered. Although some parts of IEC 61968 and IEC 62325 series are intended to relate pieces of information of different domains, they do not and cannot standardize the entire domains of electro-technical ontology, simply because such is not within the scope of one domain TC/SC.

It requires almost all the pieces of information of all the technical committees of IEC: That is why the generic description of product classes, properties and identification covering all technical domains is assigned to a horizontal committee. With the IEC 61360 CDD managed by such a committee, integration and maintenance of all the range of electro-technical ontologies is made possible.

Hence, this part of IEC 62656 provides the means and methods for importing the CIM defined ontology into IEC CDD, and thereby allows the interoperable use of the ontologies of both IEC CDD and CIM in a harmonious manner for the benefit of all electro-technical systems and applications. The case of “Transformer” class is listed in Annex E as a typical example to show how CIM and IEC CDD definitions complement each other.

With this perspective in mind, this part of IEC 62656 defines in the first place the rules and principles of mapping between CIM and POM meta-models with an aim to import the objects of the former into the IEC CDD, in a form of the latter. It also defines additional attributes to the current CIM model that are indispensable for managing the content in the IEC CDD database. In terms of the choice of source data representation, this standard directly maps the objects of CIM UML into POM, since CIM is defined originally in UML. Consequently, CIM profiles, which are subsets of the CIM model, can also be transformed into POM because they are required to be consistent with the CIM UML definitions.

Secondly, this standard intends to specify rules and principle for transforming the objects conforming to IEC 61360 represented in IEC 62656-1, into a format that conforms to the CIM standard, namely, the CIM RDFS format. This virtually means that objects in IEC 61360 CDD are transformable into pieces of ontology that can be regarded as a possible extension of CIM.

The choice of this format, RDFS, is for practical reasons: This part of IEC 62656 does not intend to fully automate the generation of CIM objects from the IEC CDD. A converter based on this standard may create CIM-like objects in RDFS, and the information contained in the objects shall be useful for updating or extending the CIM ontology of the next generation. However, the selection and validation from both the syntactic and semantic perspectives need to be performed by experts in the electric power domain, and the techno-semantic standardization in this perspective is out of the scope of this part of IEC 62656: It is a horizontal sub-committee responsible for development and maintenance of the model and content of the IEC CDD across the domains of IEC, but it does not necessarily have the expertise of each specific technical domain.

Nevertheless, the steps of integration of the CIM objects into IEC CDD are significantly facilitated by the methods described in this standard. For this standard fully describes the rules for syntactic transformation of CIM objects into IEC CDD compatible forms through the projection of the former constructs onto the parcel data structures. This standard additionally provides a set of basic rules and methods useful for semantic validation of the parcellized CIM content in IEC CDD, with an aim to assure the interoperability of the ontologies originating in different standard committees.

While the detailed semantic validation shall lie in the hands of human experts, the IEC CDD change management rules for the content stored in the IEC CDD in combination with the quality guide manual will provide a practical guide and help for this purpose.

In Clause 5, the rules and principles of the mapping between CIM and POM are defined as normative content. The detailed description of the mappings is summarized in the Annex A. Reverse transformation of the objects of the IEC CDD into CIM RDF and its relationship with the original CIM-RDF are listed in Annex C as informative.

4.2 CIM standard series

The common information model (CIM) specifies the semantics of the application programming interface (API) for energy management systems (EMS). Since the CIM is an abstract model

that represents the semantics of all the typical objects in an electric utility enterprise, it comprises multiple standard parts each of which addresses a specific field and topic concerning the power grid ontology. IEC 61970-301 is the basic one of them, and it defines a base set of CIM, as a set of packages, which provides a logical view of the functional aspects of EMS including SCADA. IEC 61970-302 is another part of the series, and it defines financial and energy scheduling aspects. IEC 61968 is another series of CIM that deals with other logical views, such as utility operations including assets, location, activities, documentation, and work management. In addition, IEC 62325-301 is a standard part which defines CIM extensions for energy market communications. There is a single and unified information model that comprises all the parts of CIM, even though they have different individual standard documents.

The methodology and model defined in IEC 61970-301:2013 is selected as the logical basis for this part of standard.

5 Rules of transformation from CIM to POM

5.1 General rules

As shown in Table 1, UML meta-model available in CIM shall be projected onto the POM meta-model. First of all, each primary class in CIM shall be mapped to a class in POM. Each attribute of CIM class shall be transformed into a corresponding property in POM. An enumeration in CIM shall be transformed to an enumeration in POM and the attributes of an enumeration in CIM shall be transformed into the corresponding terms in POM. An association between classes in CIM, including aggregation, shall be also transformed into a property in POM. Each data type in CIM shall be mapped into an instance of datatype meta-class, thus it is treated as a user defined data type that augments built-in data types of POM, and a package of CIM shall be transformed into a relation in POM.

Annex A lists the detail of the transformation rules exemplified in Table 1 in a form of a set of relations in POM: Note that the POM has a built-in capacity to describe various mapping rules between ontologies as relations. However, it is important to be aware of the distinction between the mapping table and the parcellized CIM model, which is produced as a result of the transformation rules described in Table 1.

Table 1 – Mapping of CIM objects and POM objects

UML meta-model			POM meta-model
Meta object	(1)	Class	Class
	(2)	Enumeration class	Enumeration
	(3)	Datatype class	Datatype
	(4)	Attributes of primary class	Property
	(5)	Attributes of enumeration	Term
	(6)	Package	Relation
NOTE Primary class means a class defined in the UML except classes for enumeration and data type.			

Details of the transformation rules for each object are explained in the following subclauses 5.2 to 5.8.

5.2 CIM Class

5.2.1 General

From a data modelling point of view, a class in UML is an abstraction of an information structure that represents a static component of an information system. The components are drawn as classes with their explicit attributes, and the relationships among the classes are shown as associations. A class may have a hierarchical relationship with a generalized class or a specialized class, referred to as a “parent class” or a “child class” respectively, where inheritance transcends the attributes of the parent class to its offspring. In short, a class in UML is an element of generic software design tool or methodology for object-oriented programming than a structure for representing a concept of products or services in the real world. Thus it fits for generally guiding the overall software design. However it is not necessarily the best for maintaining and updating the product and service specifications as one information entity called ontology that is separate from the body of procedural software description, for its content grows in volume and whose specification of elements evolves individually over time.

In contrast, a class in POM is an abstraction of a product concept with a collection of properties each of which is designed as an abstraction of “product property” that has in general a unit of measure and a corresponding data type, when the type is quantifiable. Moreover, a class is generated as an instance of a class meta-class, i.e., a “factory” of classes or sometimes short-named as “class parcel” and is designed to be defined and updated within a database. Therefore, a class is supposed to have, a preferred name, synonymous names, definitions, multiple language translations, globally unique IDs, versions and revisions, a set of characteristics, etc., as part of a default and pre-defined structure for representing a concept of products or services. Needless to say, inheritance and part-whole relationships are built-in as part of the model. In short, a class in POM is more specifically designed as part of a database structure for representing a concept of products and services in the real world.

With this understanding, CIM classes can be transformed into POM classes, as specified by a set of mapping rules summarized in Table 2.

In the table, the column of “UML meta-model in CIM” is for the original UML meta-model applied in the CIM. Here are two sub-columns: the “UML object” is for the UML object name, and the “CIM definition” is for the corresponding CIM definition of the UML object on the left side.

The column of “POM meta-model” is for the meta-model from POM. Here are three sub-columns: the “POM object” is for the POM object name corresponding to the CIM UML concept, “Attribute name” is for the POM attribute name and “Attribute ID” is for the ID defined in POM for the POM attribute name on the left.

The Description column is for noting a human interpretable instruction for guidance in transforming the UML meta-model in CIM into the corresponding meta-model of POM.

Table 2 – Correspondence between a CIM class in UML and a POM class

UML meta-model in CIM		POM meta-model			Description
UML object	CIM definition	POM object	Attribute name	Attribute ID	
Class	Name	Class	Short name	MDC_P004_3.<lang>	<lang> in POM shall be a two-letter ISO 639 code. As CIM only gives name in English, lang here shall be “en”.
	Description		Note	MDC_P007_1.<lang>	<lang> in POM shall be a two-letter ISO 639 code. As CIM only gives note in English, lang here shall be “en”.
	Generalization (inverse of Specializes)		Superclass	MDC_P010	

5.2.2 Name

The name of a CIM class shall be transformed to the “short name” (MDC_P004_3.<lang>) attribute of the POM class. At the same time, as the POM requires a “Preferred name”(MDC_P004_1.<lang>) for each POM class as a mandatory item, the original CIM name shall be copied to the “Preferred name” of the same class as an initial tentative name when transforming from CIM to POM for the first time. The “Preferred name” of a class in the POM may be updated according to the maintenance procedure for database standards specified in the ISO/IEC Directives Supplement.

5.2.3 Description

The “Description” attribute of a CIM Class is to explain or describe a class. Therefore, the “Description” of a CIM class shall be mapped to the “Note”(MDC_P007_1.<lang>) in a POM class. At the same time, as the POM requires a “Definition”(MDC_P005.<lang>) for each POM class as a mandatory item, the original CIM note shall be copied to the “Definition” of the same POM class as an initial tentative definition when transforming from CIM to POM for the first time. The “Definition” in the POM may be updated according to the maintenance procedure for database standards specified in the ISO/IEC Directives Supplement.

5.2.4 Generalization-specialization

UML has a capacity to describe the generalization-specialization relationship between two classes. In case of the example shown in Figure 1, the class “PowerSystemResource” is a generalization of the class “Equipment”. Specialization is the inverse of the generalization. The generalization is also known as an “is-a” relationship which can be described by a “Superclass” attribute of a class in POM. With this example, class “Equipment” has a superclass named “PowerSystemResource” in POM.

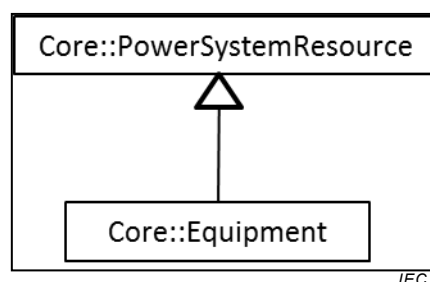


Figure 1 – CIM generalization example
(from Figure 2 of IEC 61970-301:2013)

5.3 CIM class attributes

5.3.1 General

Attributes of a UML class are the descriptive elements of a class, while a property of POM describes one characteristic aspect of a class. Thus, the UML class attributes can be mapped into POM properties with due specification of data type, etc.

Table 3 – Mapping of CIM class attribute and POM property

UML meta-model in CIM		POM meta-model			Description
UML object	CIM definition	POM object	Attribute name	Attribute ID	
Attribute of class	Name	Property	Short name	MDC_P004_3.<lang>	<lang> in POM shall be a two-letter ISO 639 code. As CIM only gives name in English, lang here shall be "en".
	(for native attribute)		Definition class	MDC_P021	for native attribute, its definition class shall be described here
	Type		Data type	MDC_P022	
	Description		Note	MDC_P007_1.<lang>	<lang> in POM shall be a two-letter ISO 639 code. As CIM only gives note in English, lang here shall be "en".

5.3.2 Name

The name of a CIM class attribute shall be transformed to the "Short name" (MDC_P004_3.<lang>) attribute of the POM property. At the same time, the original name of the CIM class attribute shall be copied to the "Preferred name" of the same POM property as an initial tentative name when transforming from CIM to POM for the first time. The "Preferred name" of a property in the POM may be updated according to the maintenance procedure for database standards specified in the ISO/IEC Directives Supplement.

5.3.3 Native attribute

In CIM, there are two types of class attributes: one type is a set of native attributes, which are defined in the class itself. The other type is the set of inherited attributes, which are inherited from the super class or ancestor classes.

POM provides the following structure for differentiating the native attributes and the inherited attributes:

- Each native attribute shall be transformed into a POM property which has its "Definition class" in the class in which the property is defined. At the same time, the property shall be added to the attribute of "Applicable properties" (explained at 6.2.15.) of the POM class.
- Inherited attributes of a class shall be collected from the "Applicable properties" of its upper classes.

5.3.4 Type

The type of a CIM attribute shall be transformed into the data type of a POM property which defines the possible data type of the instance values.

5.3.5 Description

The "Description" of the CIM class attribute shall be transformed into the "Note" attribute of a POM property. The note shall be copied to the "Definition" attribute of the same POM

property as a tentative definition, when the conversion from CIM to POM is performed for the first time. The “Note” in POM will not be changed to keep the originality of CIM, and the “Definition” in POM will be updated according to the maintenance procedure for database standards specified in the ISO/IEC Directives Supplement.

5.4 CIM association

5.4.1 General

An association is a conceptual connection among classes. In UML, an association may have a “name” and have more than two “association ends” members. For each association end, attributes of “aggregationKind”, “navigability” and “multiplicity” are defined. An aggregation in UML is a specialized case of an association. Compared to the normal association, it describes a part-whole relationship between classes. Furthermore, there is a special type of aggregation defined as “composition”, which restricts that the composite object shall have responsibility for the existence and storage of the composed objects, i.e., both objects should have the same life cycle according to the composition. As the aggregation and composition are specified types of association, three kinds of them shall be described with the same model in the POM. In this subclause 5.4, how to represent an association including aggregation and composition in POM is explained.

In the CIM, an association name is not specified normally. Each association end contains the information including multiplicity shown in Table 4. Attributes of “aggregationKind” and “navigability” are not explained directly in the association end structure in the CIM, but they can be accessed from the UML class diagram or the corresponding XMI (XML Metadata Interchange) file for the CIM.

In this part of the standard, each association end with its aggregationKind, navigability and multiplicity is mapped to a POM property. Details of the mapping rule are explained in 5.4.3.

Even though composition has not yet been applied in the CIM models, the method for representing composition in POM is introduced in this standard for a possible use of the construct in a future edition of CIM.

5.4.2 CIM association end structure

Each association end contains the information shown in Table 4, which is editorially adapted from the original defined in IEC 61970-301:2013.

As described in the CIM standard, the first column describes the multiplicity of instances of the class (per one instance of the class connected at the named end) from which the association having the end name is extended (i.e., the multiplicity of the class for which the association ends are being described). In the second column, the multiplicity of the association end, namely the range of instances of the class connected to the association end with a specific association end name in plain text is enclosed within a pair of brackets placed at the head of the name. The name of the class connected to the association end is listed in the “type” column. A multiplicity of zero indicates an optional association. A multiplicity of “*” indicates any number is allowed. For example, a multiplicity of [1..*] indicates that a range of instances from 1 to any number is allowed.

Table 4 – CIM association end structure

Multi from	[Mult to] name	Type	description
[0..*]	[1..1]BaseVoltage	BaseVoltage	The base voltage used for all equipment within the VoltageLevel

5.4.3 General transformation rule for association

In the UML, each association should have no less than two association ends. Each association end is defined as a member of the association. As only binary association which includes two association ends is available in the current CIM, the representation of a binary association is expressed in this standard.

In the IEC 62656-3, each association end is transformed to an instance of a POM property and each association is transformed to an instance of the POM relation. The association ends owned by an association are defined in the “Domain of the Relation” of the relation derived from the UML association.

For each association explained in the POM relation, values of following attributes shall be assigned as:

- Relation type(MDC_P200) has a value of “Predication”
- Role of the relation(MDC_P210) has a value of “UML Association”
- Domain of the relation(MDC_P201) has a value of a list of property ID representing association end owned by this association

For an association without a given name, the code assigned to the corresponding relation can be copied to the “Preferred name”(MDC_P004_1) attribute.

Table 5 shows samples of how to describe an association in the POM relation.

In Table 5, “Asso1” represents the association shown in Figure 2 with its association ends definition in Table 7 in the first two records. “Asso2” represents the association shown in Figure 3 with its association ends definition in Table 7 in the third and fourth records.

Table 5 –CIM associations defined as POM relations

MDC_P001_13	MDC_P004_1.<lang>	MDC_P007_1.<lang>	MDC_P200	MDC_P201	MDC_P202	MDC_P203	MDC_P208	MDC_P210
Code	Preferred name	Note	Relation type	Domain of the relation	Domain of the function	CoDomain of the function	Domain element type	Role of the relation
Asso1			Predication	{Property_code1,Property_code2}				UML Association
Asso2			Predication	{Property_code3,Property_code4}				UML Association

5.4.4 General transformation rule for association end

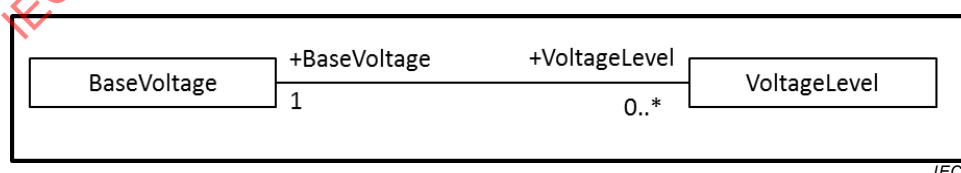
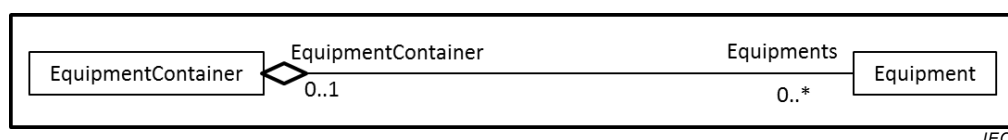
5.4.4.1 General

The CIM association end shall be mapped to a POM property. In this property, attribute of “applicable relations” should list the ID of the relation which represents the association containing this association end. The navigability of an association end is distinguished with the “property data element type” and “condition” attributes of the property. The aggregationKind is distinguished from the “remark” attribute of the property. Table 6 summarises the representation of association end of CIM in POM.

Table 6 – Mapping of CIM association end and POM property

UML meta-model in CIM		POM meta-model			Description
UML object	CIM definition	POM object	Attribute name	Attribute ID	
Association end	Association end name	Property	Short name	MDC_P004_3. <lang>	<lang> in POM shall be a two-letter ISO 639 code. As CIM only gives name in English, lang here shall be “en”.
	Description		Note	MDC_P007_1. <lang>	<lang> in POM shall be a two-letter ISO 639 code. As CIM gives note only in English, lang here shall be “en”.
			Definition class	MDC_P021	For an association end, its definition class shall be set to the class which is connected to.
			Property data element type	MDC_P020	For a property derived from an association end, the value of this attribute shall be set to “INVERSIBLE_P_DET”, for other properties from CIM, the value of this attribute shall be set to “NON_DEPENDENT_P_DET”
			Condition	MDC_P028	For a property derived from an association end with navigability, the ID of the property representing the association end at the other side shall be described in this attribute.
			Applicable relations	MDC_P230	This attribute is to describe the association containing this association end.
			Remark	MDC_P007_2	This attribute is to describe the aggregationKind which has value of “none”, “shared” or “composition”.
	[Mult to] name		Data type	MDC_P022	For a property derived from an association end, it is required to have the datatype as “class reference type”, with a reference to its connected class. For the association end example shown in Table 4, it shall be defined as a property with the following datatype: CLASS_REFERENCE_TYPE(ICID of the class BaseVoltage).
	Type				

Let us use a simple association shown in Figure 2, an aggregation shown in Figure 3 as samples to explain the transformation rules.

**Figure 2 – Simple association example (from Figure 3 of IEC 61970-301:2013)****Figure 3 – Aggregation example (from Figure 4 of IEC 61970-301:2013)**

For these two samples, Table 7 shows their corresponding POM descriptions. Each property in Table 7 represents an association end with their multiplicities.

Table 7 – CIM association ends described as POM properties

MDC_P001_6	MDC_P004_1. en	MDC_P005. en	MDC_P007_2. en	MDC_P021	MDC_P230	MDC_P020	MDC_P028	MDC_P022
Code	Preferred name	Definition	Remark	Definition class	Applicable relations	Property data element type	Condition	Data type
Property_code1	VoltageLevel	The VoltageLevel is having this BaseVoltage.	#aggregationKind=none	ICID of BaseVoltage	Asso1	INVERSIBLE_P_DET		List(0,n) of class_reference_type(ICID of class VoltageLevel)
Property_code2	BaseVoltage	The base voltage used for all equipment within the VoltageLevel.	#aggregationKind=none	ICID of class VoltageLevel	Asso1	INVERSIBLE_P_DET		class_reference_type of(ICID of class BaseVoltage)
Property_code3	Equipments	The association is used in the naming hierarchy.	#aggregationKind=shared	ICID of Equipment Container	Asso2	INVERSIBLE_P_DET		List(0,n) of class_reference_type(ICID of class Equipment)
Property_code4	Equipment Container		#aggregationKind=none	ICID of Equipment	Asso2	INVERSIBLE_P_DET		List(0,1) of class_reference_type(ICID of class EquipmentContainer)

5.4.4.2 Name of an association end

As CIM does not name an association but gives a name to each end of an association, the name of the association end, for example, “VoltageLevel” is transformed to the “short name” of a POM property which is held in a class with the preferred name “BaseVoltage”. At the same time, the name of the association end shall be copied to the “preferred name” of the same POM property as an initial tentative name, when transforming from CIM to POM for the first time.

The multiplicity of an association end at the target class, which is defined as [mult to] in the CIM, shall be described in the datatype as List(0,n) of class_reference_type. For example, the association end “VoltageLevel” described in the first record of Table 7, the multiplicity at the target class, i.e., at “BaseVoltage” class is specified as “(0,n)” in the datatype as List(0,n) of class_reference_type. For the second record of Table 7, the multiplicity at the target class, i.e., at “VoltageLevel” class is omitted as the multiplicity is “1”. Details are explained in 5.4.4.8.

5.4.4.3 Description

A “description” which explains an association shall be transformed to a “Note” attribute (MDC_P007_1.<lang>) of the POM property where <lang> is assumed to be “en”. At the same time, since the POM requires a “definition”(MDC_P005.<lang>) for each property, the original CIM note shall be copied to the “definition” attribute of the same POM property when transforming from CIM to the POM for the first time. The “definition” in the POM will be updated according to the maintenance procedure for database standards specified in the ISO/IEC Directives Supplement.

5.4.4.4 Remark

In this part of the standard, “Remark” is used to describe the “aggregationKind” of each association end. The “aggregationKind” which is the same as that of the UML, may have value of “none” for normal association, or “shared” for an aggregation, or “composition” for a composition.

In this part of the standard, in order to distinct the aggregationKind definition from other Remark, the aggregationKind shall be described with the following format:

#AggregationKind=none|shared|composition

Samples are shown in the “Remark” column of Table 7.

5.4.4.5 Property data element type and condition

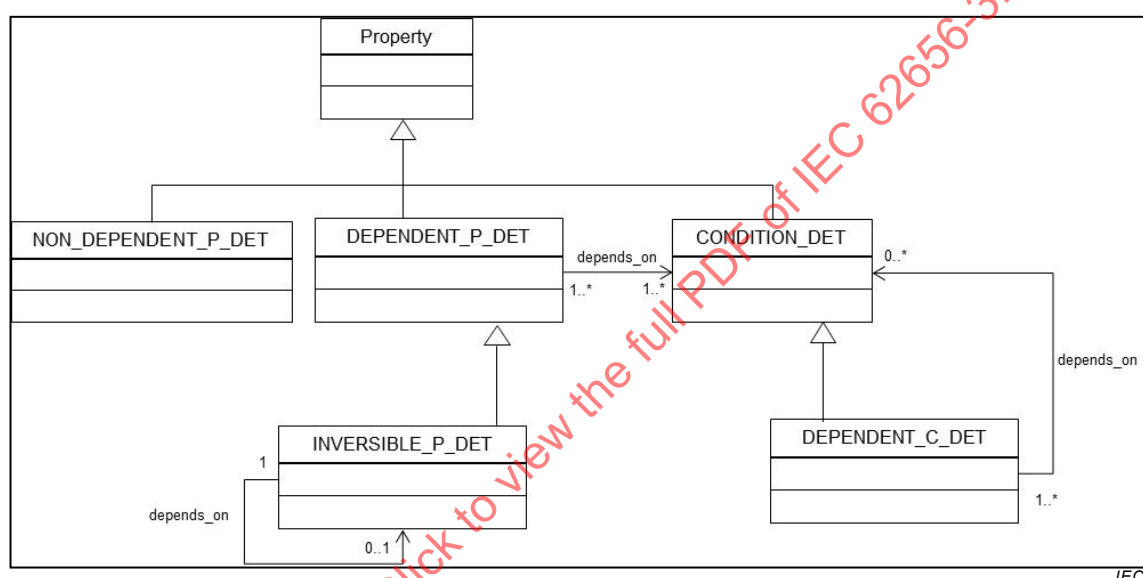


Figure 4 – Property data element type definition

In this standard, “INVERSIBLE_P_DET” is added as a new “property data element type” in order to describe the association end which may have navigability. Figure 4 shows the relationship among property and their subtypes and their dependent conditions, with the addition of “INVERSIBLE_P_DET”. In the IEC 61360 series and IEC 62656-1, a property of “DEPENDENT_P_DET” is assumed to depend on a property of “CONDITION_DET” type. In this standard, a property of “INVERSIBLE_P_DET” can depend on another property of “INVERSIBLE_P_DET”. Thus, a property derived from an association end can depend on a property derived from the association end on the opposite side. For other usages of the “INVERSIBLE_P_DET”, the cardinality of “depends_on” of the “INVERSIBLE_P_DET” does not need to be 0 or 1, but can be set to any number by necessity.

In this standard, the “INVERSIBLE_P_DET” combined with the attribute of “Condition” is applied to record the navigability of an association end.

In the UML, there are three types of navigability for an association end:

- Non-navigable

NOTE 1 In the UML class diagram, an association end with non-navigability is drawn with a X mark.

- Navigable

NOTE 2 In the UML class diagram, an association end with navigability is drawn with an arrow.

– Unspecified

NOTE 3 In the UML class diagram, an association end with unspecified navigability is drawn with a simple line. In the CIM, most association ends are defined with unspecified navigability.

Each association end shall be transformed into a POM property with the “property data element type” as “INVERSBILE_P_DET”. For an association with two navigable association ends, “Condition” of the property (e.g. Property_A) representing the association end shall have the value of the property ID (e.g. Property_B) representing the association end on the opposite side.

5.4.4.6 Applicable relations

For a POM property which is derived from an association end of CIM, this attribute is used to describe the association which contains this association end. As the association itself is described as an instance of POM relation, the corresponding ICID of the relation shall be described in this attribute. Examples are shown in Table 7.

5.4.4.7 Native and inherited association ends

In the CIM, there are two kinds of association ends for a class: native association ends and inherited association ends from its upper classes.

- Each native association end shall be transformed into a POM property which has its “definition class” in the class in which the property is defined. At the same time, the property shall be added to the “applicable properties (explained in 6.2.15)” of the POM class.
- Inherited association ends of a class shall be collected from the “applicable properties” of its upper class and ancestor classes.

5.4.4.8 Data type

POM is equipped with a property of “class_reference_type” which is capable of describing an association between two classes. Therefore, a property transformed from a CIM association end into POM shall be of “class_reference_type”. Furthermore, to describe the multiplicity of instances of a class to which the association end is connected, POM may make use of an ordered set of properties of “class_reference_type”, i.e. the data type of the property shall be described as “List(0,n) of class_reference_type (ICID)” where the ICID signifies the global ID of the class to which the association end is connected. When the multiplicity is 1, the data type does not need to be defined as “List(1) of class_reference_type(ICID)” but only as “class_reference_type(ICID)”.

For an association of “Composition” type, it is recommended to use data type of “class_instance_type” instead of “class_reference_type” because the former one restricts that the composed objects must have the same lifecycles of the composite objects.

5.5 CIM Enumeration class

5.5.1 General

In the CIM, UML stereotypes of “primitive”, “enumeration” and “datatype” are applied. A class with primitive type of the stereotype standards for a common class, a class of an enumeration type of the stereotype stands for an enumeration class and a class with datatype of the stereotype is for a datatype class. An enumeration class defines all possible values for this enumeration and enumeration class has no hierarchical structures.

POM defines an object as enumeration which is provided to describe the possible values for a property. Therefore, the CIM enumeration shall be described as a POM enumeration.

POM defines an object as datatype which shall be provided to describe the datatype of the CIM. The detail of the datatype is explained in 5.7.

Table 8 – Mapping of CIM enumeration and POM enumeration

UML meta-model in CIM		POM meta-model		
UML object	CIM definition	POM object	Attribute name	Attribute ID
Enumeration	Name	Enumeration	Short name	MDC_P004_3.<lang>
	Description		Note	MDC_P007.<lang>

5.5.2 Name

CIM enumeration class name shall be transformed to the “short name” of the POM enumeration. At the same time, the name of the enumeration class shall be copied to the “Preferred name” of the same POM enumeration as an initial tentative name, when transforming from CIM to POM for the first time. And the “Preferred name” in POM will be updated according to the maintenance procedure for database standards specified in the ISO/IEC Directives Supplement.

5.5.3 Description

The “Description” of the CIM enumeration class attribute shall be transformed to “Note” attribute of a POM enumeration. And when it is the first time to transform from CIM to POM, the “Note” shall be copied to the POM “Definition” attribute too. The “Note” in POM will not be changed to keep the originality of CIM, and the “Definition” in POM will be updated according to the maintenance procedure for database standards specified in the ISO/IEC Directives Supplement.

5.6 CIM Enumeration Attribute

5.6.1 General

In CIM, enumeration is defined as a class and its possible values are defined as its native attributes. POM defines an object as term to provide a means to define repeatedly used terms, constants or individual values used in a data model. While transforming from CIM to the POM, each possible enumeration value of the CIM shall be defined as one instance of the POM term. As shown in Table 9, attributes of an enumeration shall be transformed to instances of the POM term.

Table 9 – Mapping of CIM enumeration attribute and POM term

UML meta-model in CIM		POM meta-model		
UML object	CIM definition	POM object	Attribute name	Attribute ID
Attribute of enumeration	Name	Term	Short name	MDC_P004_3.<lang>
	Description		Note	MDC_P007.<lang>

5.6.2 Name

Name of an enumeration attribute shall be transformed to the “Short name” of the POM term. At the same time, the name of the CIM enumeration attribute shall be copied to the “Preferred name” of the same POM term as an initial tentative name, when transforming from CIM to POM for the first time. The “Preferred name” in POM will be updated according to the maintenance procedure for database standards specified in the ISO/IEC Directives Supplement.

5.6.3 Description

“Description” of an enumeration attribute shall be transformed to the “Note” of the POM term. And when it is the first time to transform from CIM to POM, the “Note” shall be copied to the

POM “Definition” attribute. The “Note” in POM shall not be changed to keep the originality of CIM, and the “Definition” in POM will be updated according to the maintenance procedure for database standards specified in the ISO/IEC Directives Supplement.

5.7 Data type

5.7.1 CIM basic datatype

In the CIM model, there are two kinds of data types: the basic data type of String, Integer, Float and Boolean, which can be used to describe other data types. The other kind is named data type which is defined with the basic data type. Named data type is defined as a class with a stereotype of data type which has a triple of attributes {value, unit, multiplier}.

Table 10 – Mapping of CIM basic data type and POM simple data type

CIM basic data type		POM simple data type (MDC_P022)
Name	Definition in the CIM	
Integer	An integer number. The range is unspecified and not limited.	INT_MEASURE_TYPE
Boolean	A type with the value space “true” and “false”	BOOLEAN_TYPE
Float	A floating point number. The range is unspecified and not limited.	REAL_MEASURE_TYPE
String	A string consisting of a sequence of 8 bit characters. The character encoding is UTF-8. The string length is unspecified and unlimited.	STRING_TYPE

In POM, there are also two kinds of data types: simple data type and named data type. The simple data type can be used to define a named datatype.

While transforming from CIM to POM, the basic data type shall be mapped to the corresponding POM simple data type. Table 10 gives the mapping rule between the CIM basic data types and the POM simple data types.

A prominent difference between the CIM basic data type and the POM data type is that POM distinguishes the data type with units from the data type without units. For example, there are both REAL_TYPE for those without units and REAL_MEASURE_TYPE for those with unit. On the contrary, the basic data type of CIM is not related to a unit. As the unit in POM is set to “1” by default, therefore, the basic data type about number in CIM shall be transformed to the POM simple data type.

5.7.2 CIM named data types

5.7.2.1 General

In CIM, a data type except the basic ones is described as a class. It shall be transformed to a named data type in POM with the rules shown in Table 11.

5.7.2.2 Name

Name of the datatype shall be transformed to the “short name” of the POM datatype. At the same time, the name of the CIM data type shall be copied to the “Preferred name” of the same POM data type as an initial tentative name, when transforming from CIM to POM for the first time. The “preferred name” in POM will be updated according to the maintenance procedure for database standards specified in the ISO/IEC Directives Supplement.

5.7.2.3 Description

Description of the CIM data type shall be transformed to the “Note” of the POM datatype. And when it is the first time to transform from CIM to POM, the “Note” shall be copied to the “Definition” attribute of the same POM datatype. The “Note” in POM will not be changed to keep the originality of CIM, and the “Definition” in POM will be updated according to the maintenance procedure for database standards specified in the ISO/IEC Directives Supplement.

Table 11 – Mapping of CIM data type and a named data type in POM

UML meta-model in CIM		POM meta-model			Description
UML object	CIM definition	POM object	Attribute name	Attribute ID	
Datatype	Name	Datatype	Short name	MDC_P004_3.<lang>	Name of a CIM data type shall be described with this attribute.
	Description		Note	MDC_P007_1.<lang>	<lang> in POM shall be a two-letter ISO 639 code. As CIM only gives note in English, lang here shall be “en”.
			Data type	MDC_P022	A corresponding POM simple data type is described here.
	Unit		Unit in text	MDC_P023_1	This attribute is used to describe the unit of a data type.
	Multiplier		Alternative units	MDC_P111	This attribute is used to describe the alternative units of a datatype or a property.

5.7.2.4 Data type

This attribute shall describe the POM simple data type corresponding to the CIM basic data type.

5.7.2.5 Unit

CIM defines a unit as part of a datatype whenever necessary. POM can associate an SI unit to each datatype. Therefore, the unit of CIM datatype shall be transformed to the unit of POM datatype.

5.7.2.6 Alternative units

CIM defines multiplier for a unit which intends to restrict possible prefix. In the CIM of IEC 61970-301:2013, the unit multiplier is an enumeration named as unitMultiplier with possible enumerators as {p,n,micro,m,c,d,k,M,G,T,none} which is part of the SI prefixes. In each named datatype, a unit multiplier as UnitMultiplier [0..1] or “none” is assigned. Here, “none” standards for that no prefix for a unit is available for a datatype. UnitMultiplier[0..1] means that no more than one enumerator can be selected from the unitMultiplier for a unit. However, there is no clear constraint on which unitMultipliers are selectable and which are not for a unit defined in a data type.

In order to make the definition of a unit and its prefix more clear, in this part of the standard, attribute of “Alternative units”(MDC_P111) shall be used to represent all selectable units for a SI unit.

5.8 Package

5.8.1 General

The CIM package diagram defines the name scope for each object in CIM. And UML packages may have part-whole relationships among each other. In POM, a relation object is used to describe the package information. Table 12 shows the general rule of how to transform a CIM package object into a POM relation object. To explain the transformation rules, a CIM package definition example shown in Figure 5 is applied. It is corresponding POM relation definitions are shown in Table 13.

IECNORM.COM : Click to view the full PDF of IEC 62656-3:2015

Table 12 – Mapping of CIM package and POM relation

UML meta-model in CIM		POM meta-model		Description
UML object	CIM definition	POM object	Attribute name	Attribute ID
Package	Name	Relation	Short name	MDC_P004_3. <lang>
	Description		Note	MDC_P007_1. <lang>
	Version		Version	MDC_P002_1
	Date		Date of current definition	MDC_P003_2
			Relation type	MDC_P200
			Domain of the relation	MDC_P201
			Domain of the function	MDC_P202
	Package including elements		Codomain of the function	MDC_P203
	Package relation		Domain element type	MDC_P208
			Role of the relation	MDC_P210

Name of a package shall be described in this POM attribute.

The explanation of a package shall be described in this POM attribute.
<lang> in POM shall be a two-letter ISO 639 code. As CIM only gives note in English, lang here shall be "en".

Version of a package shall be described in this attribute.

Date of a package shall be described in this attribute

CIM package shall be transformed to the POM relation with the relation type as "function". Otherwise, E.g., for a relation of class collections, the value of this attribute shall be set to "Predication"

This attribute is used to describe elements of a relation when the relation type(MDC_P200) is "Predication"

Elements of the package shall be listed here. Collection of each kind of elements shall be defined as an instance of relation. And the value of this attribute is only valid when the relation type(MDC_P200) is "Function"

The ICID of the parent package shall be described here.

This attribute is to describe the "element type" available in the MDC_P201.
Possible values for this attribute are the meta-property for the object type. E.g.:

(1)MDC_C002, for classes

(2)MDC_C003, for properties

(3)MDC_C006, for data type

The value of this attribute shall be set to;

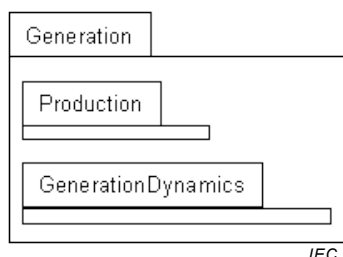
(1) "CIM package" while this relation is derived from a CIM package,

(2) "CIM association" while it's derived from a CIM association.

Otherwise the value of this attribute shall be null.

5.8.2 POM relation object

For the example shown in Figure 5, the package “Generation” includes the packages of “Production” and “GenerationDynamics”. For each package, there are objects such as class, property, data type and etc. belonging to it.



**Figure 5 – CIM package example
(from Figure 1 of IEC 61970-301:2013)**

In the POM, the relation is a kind of set. For example, if “A and B are related by relation R1”, that means A and B are belonging to the same set of R1.

Thus, for the packages of CIM, the following relation instances are defined for classes, properties and data types of a package accordingly.

- An instance of POM relation is defined as a collection of all classes included by a package
- An instance of POM relation is defined as a collection of all properties included by a package
- An instance of POM relation is defined as a collection of all datatypes included by a package

The POM relation object shall use the following attributes to describe the relationship among CIM packages, CIM package and its elements;

- Attribute “Domain of the function(MDC_P202)” shall be used to describe the ownership between a package and the objects included in it, such as classes, properties, and datatypes,
- Attribute “Codomain of the function(MDC_P203)” shall be used to describe the part-whole relationship between two packages. The ICID of the whole package shall be listed at the “Codomain of the function” of the part package
- Attribute “Role of the relation(MDC_P210)” shall be used to describe a generic type of a relation. While a relation is used to describe a CIM package, the value of this attribute shall be set to “CIM package”.

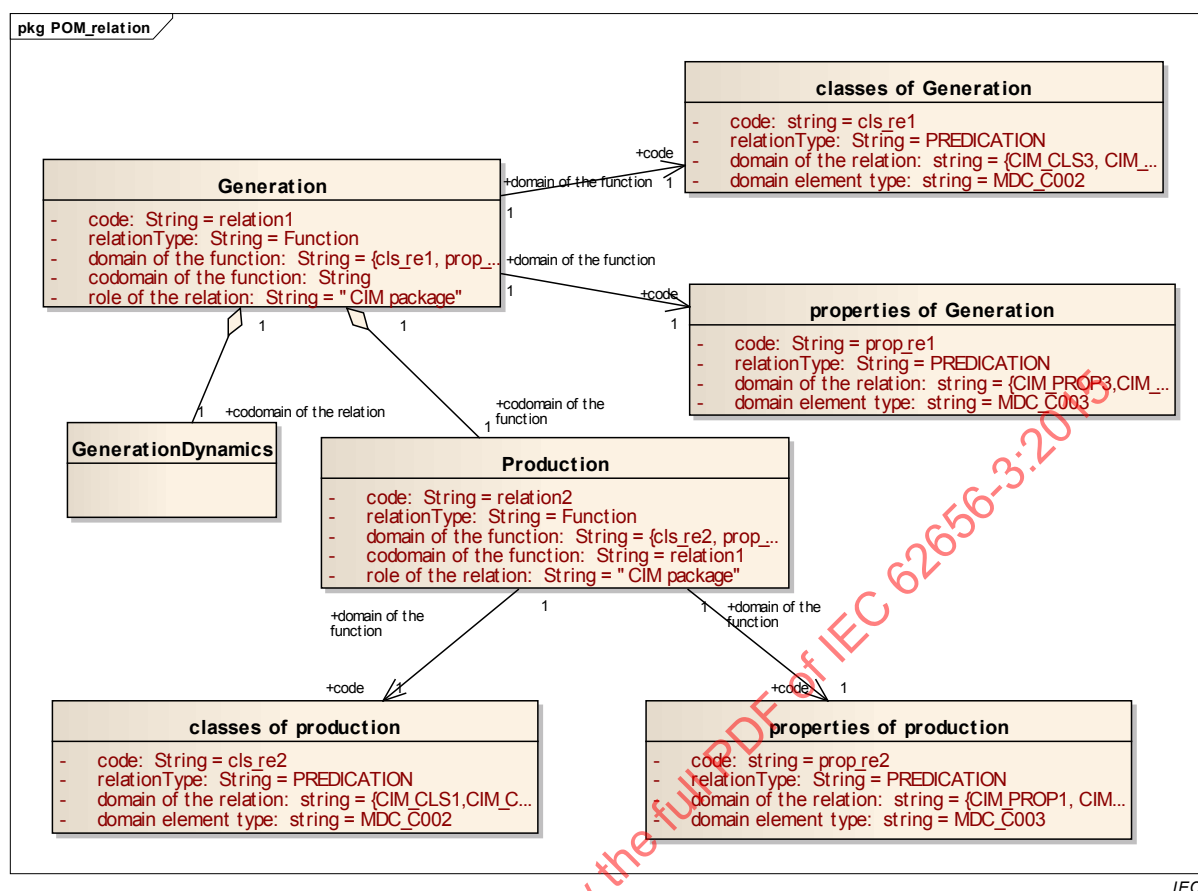


Figure 6 – POM relation model explanation in UML

Figure 6 explains schematically the use of relation objects in UML for the package example given in Figure 5. Each package represented as an instance of relation is described here as a UML class. The relationship between the "Generation" and "GenerationDynamics", "Production" package are described by the attribute "Codomain of the relation" of the part package.

At the same time, classes owned by a package shall be described as one instance of the relation. For example, "classes of generation" is defined as "cls_re1" relation and associated to the package of Generation with the attribute of "Domain of the relation".

5.8.3 CIM packages described in POM relation

5.8.3.1 General

The POM relation description of the three packages in Figure 5 is shown in Table 13. Each package shall be described as an instance of the object of relation. Each item used by CIM package is explained in this subclause 5.8.3.

Table 13 – CIM package described as POM relation

MDC_P001_13	MDC_P004_1. <lang>	MDC_P007_1. <lang>	Note	MDC_P200	MDC_P201	MDC_P202	MDC_P203	MDC_P208	MDC_P210
Code	Preferred name			Relation type	Domain of the relation	Domain of the function	CoDomain of the function	Domain element type	Role of the relation
Relation1	Generation		This package contains packages that have information for Unit Commitment and Economic Dispatch of Hydro and Thermal Generating Units, Load Forecasting, Automatic Generation Control, and Unit Modeling for Dynamic Training Simulator.	Function		{Cls_re1,Prop_re1 }			CIM package
Relation2	Production		The production package is responsible for classes which describe various kinds of generators. These classes also provide production costing information which is used to economically allocate demand among committed units and calculate reserve quantities.	Function		{Cls_re2,Prop_re2}	Relation1		CIM package
Relation3	generationDynamics		The Generation Dynamics package contains prime movers, such as turbines and boilers, which are needed for simulation and educational purposes.	Function		{Cls_re3,Prop_re3}	Relation1		CIM package
Cls_re1	Classes of Generation			PREDICATION	{CIM_cls1,CIM_cls2,...}			MDC_C002	
Prop_re1	Properties of Generation			PREDICATION	{CIM_prop1,CIM_prop2,...}			MDC_C003	
Cls_re2	Classes of production			PREDICATION	{CIM_cla11,CIM_cls12,...}			MDC_C002	
Prop_re2	Properties of production			PREDICATION	{CIM_prop11,CIM_prop12,...}			MDC_C003	

5.8.3.2 Name

Name of the package shall be transformed to the “Short name” of the POM relation. At the same time, the name of the CIM package shall be copied to the “Preferred name” of the same POM relation as an initial tentative name, when transforming from CIM to POM for the first time. The “Preferred name” in POM will be updated according to the maintenance procedure for database standards specified in the ISO/IEC Directives Supplement.

5.8.3.3 Description

CIM gives an explanation to a package which shall be transformed to the “Description” of the POM relation. And when it is the first time to transform from CIM to POM, the note shall be copied to the attribute of “Definition” of the same POM relation. The “Note” in POM will not be changed to keep the originality of CIM, and the “Definition” in POM will be updated according to the maintenance procedure for database standards specified in the ISO/IEC Directives Supplement.

5.8.3.4 Relation type

A CIM package shall be represented using a relation object in POM whose “Relation type”(MDC_P200) shall be specified as “Function”. A functional relation means that the instances at the ends of the relation have n to 1 relationship. To put it in simpler terms, for each element in a package, there is only one package to which the element belongs. Thus the relation that represents a package takes a form of “Function”.

5.8.3.5 Inclusion relation

In CIM, all the elements belong to a certain package. For each element, this relationship is described with the attribute “CIM Package”(MDC_P211) which is the synonymous name of the “Segment” attribute of IEC 62656-1, in addition to the expression of the relationship by a functional relation, in which the elements are listed in the attribute of “Codomain of the relation” (MDC_P203).

6 CIM description in IEC 62656 format

6.1 General

To represent the CIM ontology by POM, each CIM object in the POM format shall have the corresponding attributes listed in Annex B. In Clause 6, some of the fundamental attributes for each object are elaborated. All the POM objects have the following common attributes:

- **Code**
- **Version number**
- **Revision number**
- **Date of original definition**
- **Date of current version**
- **Preferred name**
- **Short name**
- **Definition**
- **Note**
- **Alternate ID(for CIM UML ID)**
- **Segment(for CIM Package)**

The “Code” attribute of each object is assigned with a unique ICID. Detailed definitions of these attributes for the class object are explained in 6.2. Although these attributes are common among all objects, some attributes are optional and their detailed definitions for other

objects are omitted. Some additional attributes are also necessary in each object. Those attributes used in this part of IEC62656 are explained further in this section.

In addition, each attribute requires several items defined in this section.

- **Destination ID:** this item shall be used to describe the data identifier(DI) of an ICID sequence assigned to an attribute in IEC 62656-1.
- **Destination name:** this item shall be used to describe the name of the attribute specified by the Destination ID given above according to IEC 62656-1 or in this standard.
- **Destination definition:** this item shall be used to describe the attribute definition in IEC 62656-1.
- **Destination value format:** this item shall be used to describe the format for the value of the attribute defined in IEC 62656-1.
- **Original CIM definition:** this item shall be used to describe whether a corresponding definition of this attribute is available in CIM or not.
- **Requirement:** this item shall be used to explain whether this attribute is “Mandatory” or “Optional” to an object.
- **Name of source CIM object:** this item shall be applied to describe the name assigned to this attribute in CIM when available.
- **Requirement for mapping:** this item shall be used to describe notes and requirements while translate from the CIM definition to this IEC 62656 attribute.

6.2 Class

6.2.1 General

Table B.1 lists essential part of the possible attributes for a class in parcellized CIM, including mandatory and optional ones. In 6.2 the common attributes and the essential attributes to the class are explained.

6.2.2 Code

- **Destination ID:** MDC_P001_5
- **Destination name:** Code
- **Destination definition:** same as IEC 62656-1
- **Destination value format:** same as IEC 62656-1
- **Original CIM definition:** not available
- **Requirement:** Mandatory
- **Name of source CIM object:** N/A
- **Requirement for mapping:** The value shall be described according to the ICID specification in IEC 62656-1. This attribute is not defined in the original CIM standard. However, as it is the identifier of a class, this attribute shall be added to each class in the parcellized CIM.

6.2.3 Version number

- **Destination ID:** MDC_P002_1
- **Destination name:** Version number
- **Destination definition:** same as IEC 62656-1
- **Destination value format:** same as IEC 62656-1
- **Original CIM definition:** not available
- **Requirement:** Mandatory
- **Name of source CIM object:** N/A

- **Requirement for mapping:** The version shall be identical to VI element of the ICID described in the code. This attribute is not defined in the original CIM standard. As it is used to identify the class updates, which influence the class instances, it shall be added to CIM class for the management of CIM updates.

6.2.4 Revision number

- **Destination ID:** MDC_P002_2
- **Destination name:** Revision number
- **Destination definition:** same as IEC 62656-1
- **Destination value format:** same as IEC 62656-1
- **Original CIM Definition:** available in IEC 61979-301:2013
- **Requirement:** Mandatory
- **Name of source CIM object:** N/A
- **Requirement for mapping:** Revision shall not affect the ranges of instances. This attribute is not defined in the original CIM standard. As it is used to identify the class updates, which do not influence the range of instances, it shall be added to the CIM class.

6.2.5 Date of original definition

- **Destination ID:** MDC_P003_1
- **Destination name:** Date of original definition
- **Destination definition:** same as IEC 62656-1
- **Destination value format:** same as IEC 62656-1
- **Original CIM definition:** not available
- **Requirement:** Mandatory
- **Name of source CIM object:** N/A
- **Requirement for mapping:** Date of the original definition shall be described in this attribute. Its value shall be in accordance with ISO 8601. The original CIM has no corresponding definition for this attribute.

6.2.6 Date of current version

- **Destination ID:** MDC_P003_2
- **Destination name:** Date of current definition
- **Destination definition:** same as IEC 62656-1
- **Destination value format:** same as IEC 62656-1
- **Original CIM definition:** not available
- **Requirement:** Mandatory
- **Name of source CIM object:** N/A
- **Requirement for mapping:** Date of the current version shall be described in this attribute. Its value shall be in accordance with ISO 8601. The original CIM has no corresponding definition for this attribute.

6.2.7 Preferred name

- **Destination ID:** MDC_P004_1.<lang>
- **Destination name:** Preferred name
- **Destination definition:** same as IEC 62656-1.
- **Destination value format:** same as IEC 62656-1
- **Original CIM definition:** not available
- **Requirement:** Mandatory
- **Name of source CIM object:** Name

- **Requirement for mapping:** The original CIM gives a name to each class object. While transforming from the CIM to the POM for the first time, the original name in CIM shall be copied to this attribute and it will be updated according to the maintenance procedure for database standards specified in the ISO/IEC Directives Supplement.

6.2.8 Short name

- **Destination ID:** MDC_P004_3.<lang>
- **Destination name:** Short name
- **Destination definition:** same as IEC 62656-1
- **Destination value format:** same as IEC 62656-1
- **Original CIM definition:** available in IEC 61970-301:2013
- **Requirement:** Mandatory
- **Name of source CIM object:** N/A
- **Requirement for mapping:** The name of a CIM class shall be mapped to the short name attribute of the corresponding POM class.

6.2.9 Definition

- **Destination ID:** MDC_P005.<lang>
- **Destination name:** Definition
- **Destination definition:** same as IEC 62656-1
- **Destination value format:** same as IEC 62656-1
- **Original CIM definition:** not available
- **Requirement:** Mandatory
- **Name of source CIM object:** Description
- **Requirement for mapping:** The original CIM provides a field called “Description” to explain each class. While transforming from CIM to POM, the CIM description shall be copied into both the “Definition” attribute and the “Note” attribute of POM. The description in POM will not be changed to keep the consistency with CIM, while the definition in POM will be updated according to the maintenance procedure for database standards conforming to the specification in ISO/IEC Directives Supplement.

6.2.10 Note

- **Destination ID:** MDC_P007_1.<lang>
- **Destination name:** Note
- **Destination definition:** same as IEC 62656-1
- **Destination value format:** same as IEC 62656-1
- **Original CIM definition:** available in IEC 61970-301:2013
- **Requirement:** Optional
- **Name of source CIM object:** Description
- **Requirement for mapping:** A CIM description shall be mapped to the Note attribute of the corresponding POM class.

6.2.11 Superclass

- **Destination ID:** MDC_P010
- **Destination name:** Superclass
- **Destination definition:** same as IEC 62656-1
- **Destination value format:** same as IEC 62656-1
- **Original CIM definition:** available in IEC 61970-301:2013
- **Requirement:** Mandatory

- **Name of source CIM object:** super class
- **Requirement for mapping:** The original CIM defines generalization relationship between a class and its super class to describe the hierarchical relationship between two classes. POM defines an attribute called “Superclass” for this purpose. However, the POM uses an ICID for the designation of the Superclass rather than a name. Therefore, in transformation from the CIM generalization relationship to the relevant POM Superclass, a mapping between a class name and its ICID is required.

6.2.12 Class type

- **Destination ID:** MDC_P011
- **Destination name:** class type
- **Destination definition:** same as IEC 62656-1
- **Destination value format:** same as IEC 62656-1
- **Original CIM definition:** not available
- **Requirement:** Mandatory
- **Name of source CIM object:** N/A
- **Requirement for mapping:** In POM possible values for this attribute are one of the following:
 - “ITEM_CLASS”;
 - “COMPONENT_CLSS”;
 - “MATERIAL_CLASS”;
 - “FEATURE_CLASS”;
 - “ITEM_CLASS_CASE_OF”;
 - “COMPONENT_CLASS_CASE_OF”;
 - “MATERIAL_CLASS_CASE_OF”;
 - “FEATURE_CLASS_CASE_OF”;

A CIM class shall always be represented as an “ITEM_CLASS”.

6.2.13 Alternate ID

- **Destination ID:** MDC_P101
- **Destination name:** Alternate ID
- **Destination definition:** CIM UML ID generated for each CIM object by a UML tool.
- **Destination value format:** same as IEC 62656-1
- **Original CIM definition:** N/A
- **Requirement:** Mandatory
- **Name of source CIM object:** CIM UML ID
- **Requirement for mapping:** A UML ID is not available in the CIM standards, but is generated by a UML tool as the identifier for each class. POM shall use this attribute to describe the UML ID generated for each CIM class.

6.2.14 CIM package

- **Destination ID:** MDC_P211
- **Destination Name:** CIM package
- **Destination definition:** the package name by which a class is contained
- **Destination value format:** same as IEC 62656-1
- **Original CIM definition:** available in IEC 61970-301:2013
- **Requirement:** Mandatory

- **Name of source CIM object:** package name
- **Requirement for mapping:** The CIM is partitioned into a set of packages. A package is a general purpose means of grouping related model elements and is defined depending on the application area and its purpose. Each object in the CIM, except for package, belongs to a certain package. The attribute “Segment” of POM, which is given a synonymous name as “CIM package” in this standard, is applied to describe the CIM package information. An ICID of a package is required for this attribute.

6.2.15 Applicable properties

- **Destination ID:** MDC_P014
- **Destination name:** Applicable properties
- **Destination definition:** same as IEC 62656-1
- **Destination value format:** same as IEC 62656-1
- **Original CIM definition:** available in IEC 61970-301:2013
- **Requirement:** Optional
- **Name of source CIM object:** N/A
- **Requirement for mapping:** The original CIM has a list of properties defined for each class and all the properties defined in a class can be applied to describe instances of the class. In POM, properties are also defined in a class. However, only the properties that are in the state of “applicable properties” can be applied to record the instances of the class. Otherwise, a property is said to be visible but not applicable. Consequently, ICIDs of CIM properties that are applicable need to be added to the attribute of “Applicable properties” of each class.

6.3 Property

6.3.1 General

A POM property, which is used to define an attribute of a CIM class and an association end in this standard, shall have the attributes listed in Table B.2. In 6.3 some selected attributes for a property in the parcellized CIM are elaborated in addition to the common ones.

6.3.2 Property data element type

- **Destination ID:** MDC_P020
- **Destination name:** Property data element type
- **Destination definition:** same as IEC 62656-1
- **Destination value format:** same as IEC 62656-1
- **Original CIM definition:** not available
- **Requirement:** Mandatory
- **Name of source CIM object:** N/A
- **Requirement for mapping:** This attribute in POM is to define the type of the property which decides whether a property is dependent on other properties and conditions or not. A property derived from the association end shall be assigned with the value as “INVERSIBLE_P_DET”. For properties from other CIM models, values shall be assigned as “NON_DEPENDENT_P_DET”.

6.3.3 Definition class

- **Destination ID:** MDC_P021
- **Destination name:** Definition class
- **Destination definition:** same as IEC 62656-1
- **Destination value format:** same as IEC 62656-1
- **Original CIM definition:** available in IEC 61970-301:2013

- **Requirement:** Mandatory
- **Name of source CIM object:** native attribute
- **Requirement for mapping:** The original CIM defines property as “native property” or “inherited property” to distinguish where it is defined. POM provides this attribute to record the class in which the property is defined. Since each property of POM requires the ICID of a class for this attribute, in the course of transformation from CIM to POM, a mapping between a class name and its ICID is required.

6.3.4 Data type

- **Destination ID:** MDC_P022
- **Destination name:** Data type
- **Destination definition:** same as IEC 62656-1
- **Destination value format:** same as IEC 62656-1
- **Original CIM definition:** available in IEC 61970-301:2013
- **Requirement:** Mandatory
- **Name of source CIM object:** Data type
- **Requirement for mapping:** Most CIM properties require a named data type when they are represented in POM data types (refer to 6.4). To transform from CIM to POM, the data type name in CIM shall be transformed into this attribute.

6.3.5 Applicable relations

- **Destination ID:** MDC_P230
- **Destination name:** Applicable relations
- **Destination definition:** same as IEC 62656-1
- **Destination value format:** M..0
- **Original CIM definition:** available in IEC 61970-301:2013
- **Requirement:** Mandatory
- **Name of source CIM object:** N/A
- **Requirement for mapping:** For a property derived from an association end, the relation ID corresponding to its owning association shall be described in this attribute.

6.3.6 Condition

- **Destination ID:** MDC_P028
- **Destination name:** Condition
- **Destination definition:** same as IEC 62656-1
- **Destination value format:** M..0
- **Original CIM definition:** available in IEC 61970-301:2013
- **Requirement:** Optional
- **Name of source CIM object:** N/A
- **Requirement for mapping:** For a property derived from an association end of a bidirectional association, the property ID representing the opposite association end shall be described in this attribute.

6.3.7 Unit in text

- **Destination ID:** MDC_P023_1
- **Destination name:** Unit in text
- **Destination definition:** same as IEC 62656-1
- **Destination value format:** same as IEC 62656-1
- **Original CIM definition:** available in IEC 61970-301:2013

- **Requirement:** Optional
- **Name of source CIM object:** Name of unit
- **Requirement for mapping:** CIM defines a (set of) unit of measurement(s) for each numerical data type wherever necessary. The unit for a property shall be in conformance with the unit defined by the data type (MDC_P022) of this property. i.e., value of this attribute shall be the same as given in 6.4.3.

6.3.8 Alternative units

- **Destination ID:** MDC_P111
- **Destination name:** Alternative units
- **Destination definition:** same as IEC 62656-1
- **Destination value format:** same as IEC 62656-1
- **Original CIM definitions:** In the CIM, there are definitions of possible prefixes to a unit.
- **Requirement:** Optional
- **Name of source CIM object:** UnitMultiplier
- **Requirement for mapping:** This attribute is used to describe available alternative units for a unit of measurement. As CIM defines unitMultiplier to describe possible prefixes for the unit of measurement, this attribute shall be used to record the available alternative units constrained by the prefix.

6.4 Data type

6.4.1 General

A data type which is used to define a CIM data type except basic ones shall have the attributes listed in Table B.3. In 6.4, some of the particular attributes are elaborated besides the common ones.

6.4.2 Definition class

- **Destination ID:** MDC_P021
- **Destination name:** Definition class
- **Destination definition:** same as IEC 62656-1
- **Destination value format:** same as IEC 62656-1
- **Original CIM definition:** available in IEC 61970-301:2013
- **Requirement:** Mandatory
- **Name of source CIM object:** native attribute
- **Requirement for mapping:** A data type in CIM is owned by a package, not by a class. Therefore, this attribute of the data type shall be kept null in POM for each CIM data type.

6.4.3 Unit in text

- **Destination ID:** MDC_P023_1
- **Destination name:** unit in text
- **Destination definition:** same as IEC 62656-1
- **Destination value format:** same as IEC 62656-1
- **Original CIM definition:** available in IEC 61970-301:2013
- **Requirement:** Optional
- **Name of source CIM object:** name of unit
- **Requirement for mapping:** CIM defines a SI unit inside the data type wherever necessary. The unit shall be transformed into the unit attribute of the datatype in POM.

6.4.4 Alternative units

- **Destination ID:** MDC_P111
- **Destination name:** Alternative units
- **Destination definition:** same as IEC 62656-1
- **Destination value format:** same as IEC 62656-1
- **Original CIM definitions:** In the CIM, there are definitions of possible prefixes to a unit.
- **Requirement:** Optional
- **Name of source CIM object:** UnitMultiplier
- **Requirement for mapping:** This attribute is used to describe available alternative units for a unit of measurement. As CIM defines unitMultiplier to describe possible prefixes for the unit of measurement, this attribute shall be used to record the available alternative units constrained by the prefix.

6.4.5 Data type

- **Destination ID:** MDC_P022
- **Destination name:** Data type
- **Destination definition:** same as IEC 62656-1
- **Destination value format:** same as IEC 62656-1
- **Original CIM definition:** available in IEC 61970-301:2013
- **Requirement:** Mandatory
- **Name of source CIM object:** N/A
- **Requirement for mapping:** This attribute is for the POM simple data types corresponding to the CIM basic data types listed in Table 10.

6.5 Enumeration

6.5.1 General

An enumeration which is used to describe the CIM enumeration class shall have the attributes listed in the Table B.4. In 6.5, some of the particular attributes for enumeration are elaborated.

6.5.2 Definition class

- **Destination ID:** MDC_P021
- **Destination name:** Definition class
- **Destination definition:** same as IEC 62656-1
- **Destination value format:** same as IEC 62656-1
- **Original CIM definition:** not available
- **Requirement:** Mandatory
- **Name of source CIM object:**
- **Requirement for mapping:** Enumeration in CIM is defined as an enumeration class. It is not owned by any class. However, POM assigns the “Definition class” to each enumeration. Since this attribute is mandatory in POM, it shall be added to CIM Enumeration and the value of this attribute can be set to null in POM.

6.5.3 Enumerated list of terms

- **Destination ID:** MDC_P043
- **Destination name:** Enumerated list of terms
- **Destination definition:** same as IEC 62656-1

- **Destination value format:** same as IEC 62656-1
- **Original CIM definition:** not available
- **Requirement:** Mandatory
- **Name of source CIM object:** N/A
- **Requirement for mapping:** Possible values are listed in this attribute for each enumeration in POM. In the course of transformation from CIM to POM, all possible values of each enumeration shall be added to this list.

6.5.4 Enumerate code list

- **Destination ID:** MDC_P044
- **Destination name:** Enumerated code list
- **Destination definition:** same as IEC 62656-1
- **Destination value format:** same as IEC 62656-1
- **Original CIM definition:** not available
- **Requirement:** Mandatory
- **Name of source CIM object:** N/A
- **Requirement for mapping:** Possible values represented in codes are listed in this attribute for each enumeration in POM. The list of this attribute shall correspond to the list of MDC_P043. In the course of transformation from CIM to POM, codes of all possible values of each enumeration shall be added to this list.

6.6 Term meta-class

6.6.1 General

Terms which are used to describe attributes of each CIM enumeration shall have the attributes listed in the Table B.5. In 6.6, one of the particular attributes for term is elaborated.

6.6.2 Preferred letter symbol in text

- **Destination ID:** MDC_P025_1
- **Destination name:** Preferred letter symbol in text
- **Destination definition:** same as IEC 62656-1
- **Destination value format:** same as IEC 62656-1
- **Original CIM definition:** not available
- **Requirement:** Optional
- **Name of source CIM object:** enumeration.enumerator
- **Requirement for mapping:** POM provides this attribute to describe the “Short name” of each term. A symbol in letters may be used to represent a term wherever necessary. This attribute shall be added to each term in CIM.

6.7 Relation

6.7.1 General

In this standard, the “Relation” object of POM is used to describe three types of the CIM models; (1)a CIM package,(2)elements collections of a package, and (3)CIM association. Annex G gives an explanation of the relation usages in details.

To describe the above three types of CIM models, the POM Relation shall have the attributes listed in the Table B.6. In 6.7, some of the particular attributes are elaborated.

6.7.2 Relation type

- **Destination ID:** MDC_P200
- **Destination name:** Relation type
- **Destination definition:** same as IEC 62656-1
- **Destination value format:** same as IEC 62656-1
- **Original CIM definition:** not available
- **Requirement:** Mandatory
- **Name of source CIM object:** N/A
- **Requirement for mapping:** A package in CIM shall be transformed to a POM relation with a relation type of “Function”. For other CIM models including the association, the relation type shall be set to “Predication”.

6.7.3 Domain of the relation

- **Destination ID:** MDC_P201
- **Destination name:** Domain of the relation
- **Destination definition:** same as IEC 62656-1
- **Destination value format:** same as IEC 62656-1
- **Original CIM definition:** not available
- **Requirement:** Optional
- **Name of source CIM object:** N/A
- **Requirement for mapping:** There are two types of data available in this attribute from this standard. One type of the data is for elements collections of a CIM package. In the CIM, each package contains a group of objects such as CIM classes and CIM attributes. In the POM, the containing objects of a package shall be defined as instances of a relation according to the object type. This attribute is used to describe the list of containing elements collection. For the example in Table 13, all classes contained by package “Generation” shall be defined as an instance of relation (e.g. cls_re1) which shall give a value of this attribute to the containing class lists, the value of “Domain element type” to “MDC_C002” and the value of the “Relation type(MDC_P200)” to “Predication”. For other types of CIM objects contained by a package, the same rule is applied.

The other type of the data is for the CIM association. This attribute shall be used to describe the list of the properties which represent the association ends contained by the association. Table 5 gives an example.

6.7.4 Domain of the function

- **Destination ID:** MDC_P202
- **Destination name:** Domain of the function
- **Destination definition:** same as IEC 62656-1
- **Destination value format:** same as IEC 62656-1
- **Original CIM definition:** not available
- **Requirement:** Optional
- **Name of source CIM object:** N/A
- **Requirement for mapping:** In CIM, each package contains a group of objects such as CIM classes and CIM attributes. In the POM, the containing objects of a package shall be defined as instances of Relation according to the object type such as the example described in 6.7.3. All the containing objects shall be described in this attribute with the value of the “Relation type(MDC_P200)” set to “function”.

6.7.5 Codomain of the function

- **Destination ID:** MDC_P203

- **Destination name:** Codomain of the function
- **Destination definition:** same as IEC 62656-1
- **Destination value format:** same as IEC 62656-1
- **Original CIM definition:** not available
- **Requirement:** Optional
- **Name of source CIM object:** N/A
- **Requirement for mapping:** In CIM, there are inclusion relationships among packages. The ICID of the containing package shall be described in this attribute of a package. For example, from IEC 61970-301:2013, the package “Generation” contains the package “Production” and “GenerationDynamics”. For the definition of the package “Production” and “GenerationDynamics”, the ICID of the package “Generation” shall be described in this attribute. See the example in Table 13.

6.7.6 Domain element type

- **Destination ID:** MDC_P208
- **Destination name:** Domain element type
- **Destination definition:** same as IEC 62656-1
- **Destination value format:** same as IEC 62656-1
- **Original CIM definition:** not available
- **Requirement:** Optional
- **Name of source CIM object:** N/A
- **Requirement for mapping:** This attribute is used to define the object type described in the “Domain of the relation” attribute while possible. The available value for this attribute is the name of the meta-property in the POM. For example, “MDC_C002” stands for the collection of classes, “MDC_C003” stands for the collection of properties, “MDC_C006” stands for the collection of data types.

6.7.7 Role of the relation

- **Destination ID:** MDC_P210
- **Destination name:** Role of the relation
- **Destination definition:** same as IEC 62656-1
- **Destination value format:** same as IEC 62656-1
- **Original CIM definition:** not available
- **Requirement:** Optional
- **Name of source CIM object:** N/A
- **Requirement for mapping:** This attribute is used to define the role of the relation. When a relation is derived from a CIM package, the value of this attribute shall be set to “CIM package”. When a relation is derived from a CIM association, the value of this attribute shall be set to “CIM Association”. For other relations available from this standard, such as a relation for a collection of classes, the value of this attribute shall be set to null. This attribute can be used to distinguish the “CIM package”, “CIM association” definitions from others.

7 XML schema for updated CIM instances in IEC CDD

7.1 General

A CIM RDF schema is defined in IEC 61970-501:2006, corresponding to the conceptual model of CIM in UML defined in IEC 61970-301:2013, for the purpose of providing a common exchange format and a set of data access interfaces to the metadata about electric power grids. A file conforming to the schema is designed to be both human and machine readable and it can be more easily extended than the CIM model defined in UML.

For a CIM file that is transformed into the POM, i.e., the parcellized CIM, updates of its instances that do not entail a CIM schema change or extension can be performed according to the maintenance procedures for the standards in the database format defined in Annex SL of the supplement specific to IEC of the ISO/IEC Directives. Updates to the instance content, originally modelled in CIM, shall be traceable to the original CIM instance. If no extension is made to the CIM schema, the original CIM RDF schema shall be used, since there is no need to record any schematic change or addition made by POM. However, if a CIM-schema extension does occur, for example, version and revision information in the POM is judged indispensable thus required to be exported, the original CIM RDF schema needs to be extended in a way that enables a clear demarcation of the boundaries between the original CIM RDF schema and the model extensions (vis-à-vis CIM) made by POM. Detail of this demarcation is elaborated in 7.2.

7.2 Principles

While designing the transformation rules from the parcellized CIM to a CIM RDF file, the existing RDF schema description in IEC 61970-501:2006 is stored for use without change so that any applications using the CIM RDF will function without considering the extended RDF schema in POM.

Attributes originating in CIM			Attributes extended by POM		
MDC_P004_1.en	CIM_P002	MDC_P007_1.en	MDC_1_6	MDC_P002_1	...
Name	CIM package	Note	Code	version	
Instances of CIM objects	Switch	Wires	A generic device designed to close, or open, or both, one or more electric circuits.	cim_cls_1	001
	Breaker	Wires	A mechanical switching device capable of making, carrying, and breaking currents under normal circuit conditions and also making...	cim_cls_2	001
	Jumper	Wires	A short section of conductor with negligible impedance which can be manually removed and replaced if the circuit is de-energized...	cim_cls_3	001
Extended instances in POM	Battery	N/A	AAA017	001	
	Switchgear	N/A	ACC100	001	
	...				

IEC

Figure 7 – Class instances of parcellized CIM

Figure 7 shows an example of the class instances of parcellized CIM. Note that contents shown in this figure are part of the formal definitions.

As shown in the rows of Figure 7, instances of parcellized CIM objects can be divided into two types according to their source definitions;

- “Instances of CIM objects” that derive from CIM are grouped as type1,
- “Extended instances in POM” are grouped as type2.

To describe these two types of instances, two groups of attributes are applied, as shown in the columns of Figure 7, specifically;

- “Attributes originating in CIM” correspond to the attributes which have original CIM definitions. Thus the IEC 61970-501:2006 RDF schema can be applied to describe their instances. For example, in Figure 7, for the instances in orange colours, i.e., the Name, CIM package, Note columns defined for a CIM class, can be described with the IEC 61970-501:2006 RDF schema.

- “Attributes extended by POM” are those that are only available in the POM. For the example in Figure 7, the Code, Version columns are attributes added by POM for a class. As they have no original CIM definitions thus IEC 61970-501:2006 has no RDF schemas to describe their instances. In this case, an extended XML schema subsuming IEC 61970-501:2006 definitions is created, conforming to the generic RDF extension rules.

Annex C illustrates XML schemas as informative information, to describe the latter case that contains the two types of attributes of the parcellized CIM. In Annex C, an additional namespace: “pom” is provided to describe the “attributes extended by POM” of an instance. For example, in the “pom” namespace, XML schemas for Code,Version shown in Figure 7 are expressed as <pom:code> and <pom:version> separately.

Annex D gives example XML files for each kind of the parcellized CIM models as informative information. In these examples, descriptions of the “attributes originating in CIM” come first, then the “attributes extended by POM” follow, being marked by the tag <pom:ext>.

8 Version control mechanism in POM for the parcellized CIM

In the parcellized CIM, an object is always identified by a globally unique ID and references to the objects are always made through the ID. The Figure 8 depicts this reference mechanism.

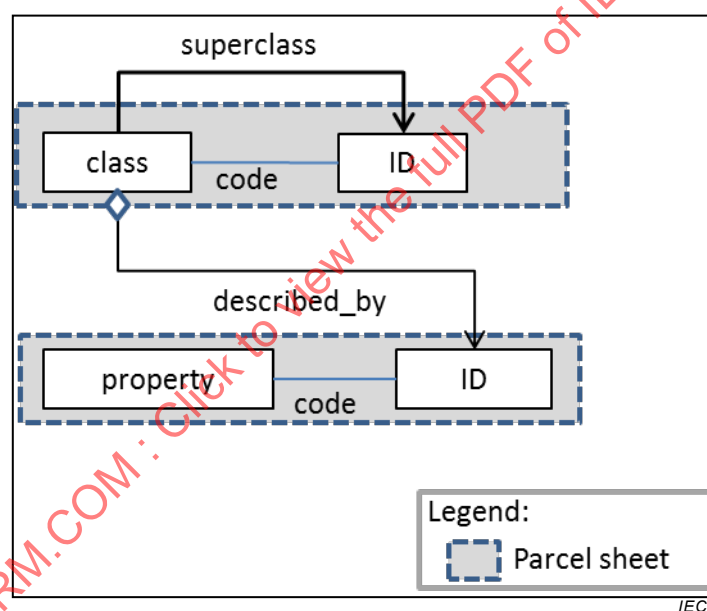


Figure 8 – Conceptual reference mechanism with ID in POM

In IEC 62656-1, as a globally unique ID, an ICID (International Concept ID), which includes the version information, is used to identify each object and the reference relationships between objects. As described in IEC 62656-1, each ICID is unique across the boundary of organizations in the parcel format and it has the following sequence:

ICID ::= RAI'#'DI'##'VI

Here, RAI stands for the registration authority identifier, DI for the data identifier, and VI for the version identifier. When an object is updated, the version included in the ICID will be updated accordingly. With the above ICID structure and reference relationship descriptions, it is clear whether an object is updated or not with the version number included in the ICID, consequently an entity-by-entity version control can be realized with the parcellized CIM.

Annex A

(normative)

Mapping rules from CIM to POM – Mapping rules described in POM relation meta-class

Mapping rules between UML meta-models available in CIM and the meta-models of POM can be described using the relation meta-class in POM. The UML meta-models available in CIM which require mapping rules are:

- Class
- Class attribute
- Association
- Enumeration
- Enumeration attribute
- Data type
- Package

The mapping rules for the above meta-models in CIM described as POM relations are listed in tables from Table A.1 to Table A.8. In these tables, the UML meta-model available in CIM are listed in the column of “Domain of the function”, and the corresponding POM meta-model are listed in the column of “Codomain of the function”.

About the mapping rules from POM meta-model to UML meta-model, beside the cases of 1:1 mapping relation, there are the cases of 1:n mappings from POM meta-models to UML meta-models. The mapping rules of 1:n for meta-models are listed in the Table A.9. In this table, the POM meta-models are listed in the column of “Domain of the function”, and the possible UML meta-models are listed in the column of “Codomain of the function”.

For example, each POM property can be mapped to (1) an attribute of CIM class, or (2) a CIM association end depending on the “Property data element type”. When the value of the “Property data element type” is “INVERSIBLE_P_DET”, the POM property shall be mapped to a CIM association end. For other POM property with “NON_DEPENDENT_P_DET”, it shall be mapped to an ordinary attribute of its corresponding CIM class. Accordingly, the attributes of the POM Property shall be mapped to different meta-models of the CIM. Another example is the POM relation. Each POM relation may be mapped to (1) A CIM package; (2) A CIM association; etc. With mapping rules of Table A.1 to Table A.9, it is available to transform the parcellized CIM meta-model to the UML meta-model.

Table A.1 – Mapping rule from CIM class to POM class

MDC_P001_13	MDC_P004_1.<lang>	MDC_P005.<lang>	MDC_P200	MDC_P202	MDC_P203	MDC_P007_1.<lang>
Code	Preferred name	definition	Relation type	Domain of the function	Codomain of the function	Note
cls_re_001	Name mapping for classes	Mapping between the name of a CIM class and the preferred name of a POM class	Function	cim.class.name	MDC_P004_3.<lang>	The short name of a POM class is designated by MDC_P004_3.<Lang>
cls_re_002	Description mapping for classes	Mapping between the description of a CIM class and the note of a POM class	Function	cim.class.description	MDC_P007_1.<lang>	The note of a POM class is designated by MDC_P007_1.<Lang>
cls_re_003	Mapping for generalization	Mapping between the generalization relation of CIM classes and superclass of a POM class	Function	cim.class.superclass	MDC_P010	The superclass of a POM class is designated by MDC_P010

Table A.2 – Mapping rule from CIM class attribute to POM property

MDC_P001_13	MDC_P004_1.<lang>	MDC_P005.<lang>	MDC_P200	MDC_P202	MDC_P203	MDC_P007_1.<lang>
Code	Preferred name	definition	Relation type	Domain of the function	Codomain of the function	Note
cls_attr_re_001	Name mapping for CIM class attributes	Mapping between the name of a CIM class attribute and the preferred name of a POM property	Function	cim.classAttribute.name	MDC_P004_3.<lang>	The short name of a POM property is designated by MDC_P004_3.<Lang>
cls_attr_re_002	Mapping for CIM class attribute type	Mapping between the type of a CIM class attribute and the data type of a POM property	Function	cim.classAttribute.type	MDC_P022	The data type of a POM property is designated by MDC_P022
cls_attr_re_003	Description mapping for CIM class attribute	Mapping between the description of a CIM class attribute and the note(MDC_P007_1.<lang>) of a POM property	Function	cim.classAttribute.description	MDC_P007_1.<lang>	The note of a POM property is designated by MDC_P007_1.<Lang>
cls_attr_re_004	Mapping for CIM attribute ownership	Mapping between the ownership of a CIM attribute and the definition class of a POM property	Function	cim.classAttribute.ownership	MDC_P021	The definition class of a POM property is designated by MDC_P021

Table A.3 – Mapping rule from CIM association to POM relation

MDC_P001_13	MDC_P004_1.<lang>	MDC_P005.<lang>	MDC_P200	MDC_P202	MDC_P203	MDC_P007_1.<lang>
Code	Preferred name	definition	Relation type	Domain of the function	Codomain of the function	Note
association_re_001	Mmapping for CIM association name	Mapping between the name of a CIM association and the preferred name of a POM relation	Function	cim.association.name	MDC_P004_3.en	The preferred name of a POM relation is used to describe the name of an association
association_re_002	mapping for association end members and "Domain of the relation"	Mapping between the association end member owned by an association and the Domain of the relation of a POM relation	Function	cim.association.associationEnds	MDC_P201	List of the property representing the association ends of an association shall be described in the "Domain of the relation" of a POM relation.

Table A.4 – Mapping rule from CIM association end to POM property

MDC_P001_13	MDC_P004_1.<lang>	MDC_P005.<lang>	MDC_P200	MDC_P202	MDC_P203	MDC_P007_1.<lang>
Code	Preferred name	definition	Relation type	Domain of the function	Codomain of the function	Note
associationend_re_001	Mapping for CIM associationend name	Mapping between the name of a CIM association end and the short name of a POM property	Function	cim.associationend.name	MDC_P004_3.en	The short name of a POM property is designated by MDC_P004_3.<Lang>
associationend_re_002	Mapping for CIM associationend description	Mapping between the description of a CIM association end and the POM property note	Function	cim.associationend.description	MDC_P007_1.en	The note of a POM property is designated by MDC_P007_1.<Lang>
associationend_re_003	Mapping for CIM association end ownership	Mapping between the class where the CIM associationend is defined and the definition class of a POM property	Function	cim.associationend.ownership	MDC_P021	The definition class of a POM property is designated by MDC_P021
associationend_re_004	Mapping for CIM aggregationKind	Mapping between the aggregationKind of a CIM associationend and the Remark of a POM property	Function	cim.associationend.aggregationKind	MDC_P007_2.en	The CIM aggregationKind of a POM property is described in the Remark designated by MDC_P007_2.en
associationend_re_005	Mapping for the opposite associationend of a bidirectional association	Mapping between the opposite associationend of a bidirectional association and the Condition of the POM property representing the associationend	Function	cim.associationend.opposite.end	MDC_P028	The opposite associationend of a bidirectional association shall be described in the MDC_P028(Remark) attribute of the POM property
associationend_re_006	Mapping for the class to which an associationend is connected	Mapping between the class to which a CIM associationend is connected and the class defined in the class_reference_type of the POM property data type	Function	cim.associationend.targetClass	MDC_P022	The class to which an associationend is connected shall be described in the class_reference_type of the data type(MDC_P022) attribute of the POM property.

Table A.5 – Mapping rule between CIM enumeration and POM enumeration

MDC_P001_13	MDC_P004_1.<lang>	MDC_P005.<lang>	MDC_P200	MDC_P202	MDC_P203	MDC_P007_1.<lang>
Code	Preferred name	definition	Relation type	Domain of the function	Codomain of the function	Note
enum_re_001	Name mapping for CIM enumeration	Mapping between the name of CIM enumeration and the preferred name of POM enumeration	Function	cim.enumeration.name	MDC_P004_3.<lang>	The short name of a POM enumeration is designated by MDC_P004_3.<lang>
enum_re_002	Description mapping for CIM enumeration	Mapping between the description of CIM enumeration and the note of POM enumeration	Function	cim.enumeration.description	MDC_P007_1.<lang>	The note of a POM enumeration is designated by MDC_P007_1.<lang>

Table A.6 – Mapping rule between CIM enumeration attribute and POM term

MDC_P001_13	MDC_P004_1.<lang>	MDC_P005.<lang>	MDC_P200	MDC_P202	MDC_P203	MDC_P007_1.<lang>
Code	Preferred name	definition	Relation type	Domain of the function	Codomain of the function	Note
enum_attr_re_001	enumeration attribute name	Mapping between the name of CIM enumeration attribute and the preferred name of a POM term	Function	cim.enumeration.attribute.name	MDC_P004_3.<lang>	The short name of a POM term is designated by MDC_P004_3.<lang>
enum_attr_re_002	enumeration attribute description	Mapping between the description of CIM enumeration attribute note and the note of POM term	Function	cim.enumeration.attribute.description	MDC_P007_1.<lang>	The note of a POM term is designated by MDC_P007_1.<lang>

Table A.7 – Mapping rule between CIM data type and POM data type

MDC_P001_13	MDC_P004_1.<lang>	MDC_P005.<lang>	MDC_P200	MDC_P202	MDC_P203	MDC_P007_1.<lang>
Code	Preferred name	definition	Relation type	Domain of the function	Codomain of the function	Note
datatype_re_001	Name mapping for CIM data type	Mapping between the name of CIM data type and the preferred name of POM data type	Function	cim.datatype.name	MDC_P004_3.<lang>	The short name of a POM data type is designated by MDC_P004_3.<lang>
datatype_re_002	Description mapping for CIM data type	Mapping between the description of CIM data type and the preferred name of POM data type	Function	cim.datatype.description	MDC_P007_1.<lang>	The note of a POM data type is designated by MDC_P007_1.<lang>
datatype_re_003	Mapping for CIM basic data type	Mapping between the CIM basic data type and the simple data type of POM	Function	cim.datatype.type	MDC_P022	The 1:1 relationships between CIM basic data type and POM simple data type are listed in Table 10.
datatype_re_005	Mapping for CIM unit	Mapping between the unit of a CIM data type and the unit in text of a POM data type	Function	cim.datatype.unit	MDC_P023_1	The unit in text of a POM data type is designated by MDC_P023_1
datatype_re_006	Mapping for CIM multiplier for unit	Mapping between the multiplier of CIM unit and the alternative units of a POM unit	Function	cim.datatype.multiplier	MDC_P111	The variable prefixes for CIM unit is designated by alternative units(MDC_P111)

Table A.8 – Mapping rule between CIM Package and POM relation

MDC_P001_13	MDC_P004_1.<lang>	MDC_P005.<lang>	MDC_P200	MDC_P202	MDC_P203	MDC_P007_1.<lang>
Code	Preferred name	definition	Relation type	Domain of the function	Codomain of the function	Note
package_re_001	Mapping for CIM Package name	Mapping between the CIM package name and the POM relation name attribute	Function	cim.package.name	MDC_P004_3.<lang>	a package name shall be mapped to the short name of a relation in POM
package_re_002	Mapping for CIM package description	Mapping between the CIM package description and the POM relation note	Function	cim.package.description	MDC_P007_1.<lang>	a description for a CIM package shall be mapped to a package note of POM
package_re_003	Mapping for elements of the CIM package	Mapping between the elements list and the POM "domain of the relation"	Function	cim.package.elements	MDC_P201	The elements lists shall be described by POM MDC_P201 of a package
package_re_004	Mapping for CIM part-whole package	Mapping between the CIM package part-whole relation and the POM relation "codomain of the relation"	Function	cim.package.whole	MDC_P202	The "whole" package shall be listed in the MDC_P202 attribute of the POM relation for "part" package
package_re_005	Mapping for CIM package version	Mapping between the CIM package version and the POM relation version	Function	cim.package.version	MDC_P002_1	The version for a CIM package shall be mapped to the POM relation version
package_re_006	Mapping for CIM package date	Mapping between the CIM package date and the POM relation "date of current definition"	Function	cim.package.date	MDC_P003_2	The date of the CIM package shall be mapped to the POM relation date_of_current_definition
Package_re_007	Mapping for CIM elements type	Mapping between the CIM elements type, such as a class, a property, a datatype and the POM "domain element type"	Function	cim.package.element.type	MDC_P208	Possible types for CIM elements are: (1)MDC_C002 for class elements (2)MDC_C003 for property elements (3)MDC_C006 for data type elements

Table A.9 – Mapping rule from POM property to CIM aggregation/association/class attribute

MDC_P001_13	MDC_P004_1.<lang>	MDC_P005.<lang>	MDC_P200	MDC_P202	MDC_P203	MDC_P007_1.<lang>
Code	Preferred name	definition	Relation type	Domain of the function	Code main of the function	Note
Pom_re_1	Mapping for POM property short name	Mapping from the short name of a POM property to the name of a CIM association end or the name of a class attribute	Function	MDC_P004_3.en	cim.associationend.name cim.classAttribute.name	The short name of a POM property is designated by MDC_P004_3.en
Pom_re_2	Mapping for POM property note	Mapping from the note of a POM property to the description of a CIM aggregation, CIM association or a class attribute	Function	MDC_P007_1.en	cim.aggregationend.description cim.classAttribute.description	The note of a POM property is designated by MDC_P007_1.<lang>
Pom_re_3	Mapping for POM property definition class	Mapping from the definition class of a POM property to the ownership of a CIM aggregation, CIM association or a class attribute	Function	MDC_P021	cim.associationend.define dclass cim.classAttribute.defined class	The definition class of a POM property is designated by MDC_P021
Pom_re_4	Mapping for POM relation name	Mapping from the POM relation to a CIM package or a CIM association	Function	MDC_P004_3.en	cim.package.name cim.association.name	The short name of a POM relation is designated by MDC_P004_3.en
Pom_re_5	Mapping for POM Relation note	Mapping from the note of a POM relation to the description of a CIM package or the description of a CIM association	Function	MDC_P007_1.en	cim.package.description cim.association.description	The note of a POM relation is designated by MDC_P007_1.<lang>

Annex B

(normative)

CIM in IEC 62656 format – Normative meta-properties of the CIM in IEC 62656 model

Tables B.1 to B.6 summarize the CIM model described in IEC 62656 format. The normative meta-properties of POM class, property, datatype, enumeration, term and relation are listed. For other informative meta-properties of POM, reference to IEC 62656-1 is recommended. In the “additional information” column;

- “Original CIM definition” sub-column is used to describe whether the meta-property defined in this row is available in the original CIM(IEC 61970-301:2013) or not. “N/A” means “not available”.
- “IEC62656-Part1 definition” sub-column is used to describe whether the meta-property defined in this row is available in IEC 62656-1 or not. “N/A” means “not available”.
- “CIM name” sub-column is used to describe the name available in the original CIM for this meta-property.

Other columns have the same meanings as IEC 62656-1.

Most of the properties of meta-classes are expressed using STRING type, thus wherever possible and necessary the length of the characters are indicated by value format (MDC_P024). For example, “M..0” signifies a string of any length while “M..14” signifies that the length of the string shall be within 14 characters.

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 ISO 639 code.

Table B.1 – Meta-properties for class meta-class

MMDC_P004_1.EN	MMDC_P001	MMDC_P008	MMDC_P100	MMDC_P102	Additional Information		
Name in English	Property ID	Data type	Value format	Requirement	Original CIM definition	IEC62656-1 definition	CIM name
code	MDC_P001_5	STRING_TYPE	M..255	MAND	N/A	available	
Version number	MDC_P002_1	STRING_TYPE	M..9	MAND	N/A	available	
Revision number	MDC_P002_2	STRING_TYPE	M..3	MAND	N/A	available	
Date of original definition	MDC_P003_1	STRING_TYPE	M..10	MAND	N/A	available	
Date of current version	MDC_P003_2	STRING_TYPE	M..10	MAND	N/A	available	
Preferred name	MDC_P004_1.<lang>	TRANSLATABLE_STRING_TYPE	M..255	MAND	available	available	Name
Synonymous name	MDC_P004_2.<lang>	SET(0,?) OF LIST(2,2) OF STRING_TYPE	M..255	OPT	N/A	available	
Short name	MDC_P004_3.<lang>	TRANSLATABLE_STRING_TYPE	M..255	MAND	available	available	Name
Definition	MDC_P005.<lang>	TRANSLATABLE_STRING_TYPE	M..0	MAND	available	available	Description
Note	MDC_P007_1.<lang>	TRANSLATABLE_STRING_TYPE	M..0	OPT	available	available	Description
Simplified drawing	MDC_P008_1	STRING_TYPE	M..120	OPT	N/A	available	
Superclass	MDC_P010	STRING_TYPE	M..0	MAND	available	available	Parent class name
Class type	MDC_P011	STRING_TYPE	M..0	MAND	N/A	available	
Applicable properties	MDC_P014	SET(0,?) OF STRING_TYPE	M..0	OPT	N/A	available	
Alternative ID	MDC_P101	LIST(0,?) OF STRING_TYPE	M..0	MAND	available	available	CIM UML ID
CIM package	MDC_P211	STRING_TYPE	M..255	MAND	available	available	CIM package

Table B.2 – Meta-properties for property meta-class

MMDC_P004_1.EN	MMDC_P001	MMDC_P008	MMDC_P100	MMDC_P102	Additional Information	
Name in English	Property ID	Data type	Value format	Requirement	Original CIM definition	IEC62656-1 definition
Code	MDC_P001_6	STRING_TYPE	M..255	MAND	N/A	available
Version number	MDC_P002_1	STRING_TYPE	M..9	OPT	N/A	available
Revision number	MDC_P002_2	STRING_TYPE	M..3	MAND	N/A	available
Date of original definition	MDC_P003_1	STRING_TYPE	M..10	MAND	N/A	available
Date of current version	MDC_P003_2	STRING_TYPE	M..10	MAND	N/A	available
Preferred name	MDC_P004_1.<lang>	TRANSLATABLE_STRING_TYPE	M..255	MAND	available	available
Synonymous name	MDC_P004_2.<lang>	SET(0,?) OF LIST(2,2) OF STRING_TYPE	M..255	OPT	N/A	available
Short name	MDC_P004_3.<lang>	TRANSLATABLE_STRING_TYPE	M..255	MAND	available	available
Definition	MDC_P005.<lang>	TRANSLATABLE_STRING_TYPE	M..255	MAND	available	Description
Note	MDC_P007_1.<lang>	TRANSLATABLE_STRING_TYPE	M..0	OPT	available	Description
Graphics	MDC_P008_2	STRING_TYPE	M..120	OPT	N/A	available
Property data element type	MDC_P020	STRING_TYPE	M..0	MAND	N/A	available
Definition class	MDC_P021	STRING_TYPE	M..0	MAND	available	A native CIM attribute
Data type	MDC_P022	STRING_TYPE	M..0	MAND	available	CIM datatype
Unit in text	MDC_P23_1	STRING_TYPE	M..120	OPT	N/A	unit
Preferred letter symbol in text	MDC_P025_1	STRING_TYPE	M..0	OPT	N/A	available
Condition	MDC_P028	SET(0,?) OF STRING_TYPE	M..0	OPT	Available	Opposite association end
Code for unit	MDC_P041	STRING_TYPE	M..120	OPT	N/A	available
Alternate ID	MDC_P101	LIST(0,?) OF STRING_TYPE	M..0	MAND	available	CIM UML ID
Segment	MDC_P211	STRING_TYPE	M..255	MAND	available	CIM package
Applicable relations	MDC_P230	SET(0,?) OF ICID_STRING	M..0	MAND	available	Association owing an association end
Alternative units	MDC_P111	STRING_TYPE	M..0	OPT	available	unitMultiplier

Table B.3 –Meta-properties for datatype meta-class

MMDC_P004_1.EN	MMDC_P001	MMDC_P008	MMDC_P100	MMDC_P002	Additional information		
Name in English	Property ID	Data type	Value format	Requirement	Original CIM definition	IEC62656-1 definition	CIM name
Code	MDC_P001_7	STRING_TYPE	M..255	MAND	N/A	available	
Version number	MDC_P002_1	STRING_TYPE	M..9	OPT	N/A	available	
Revision number	MDC_P002_2	STRING_TYPE	M..3	MAND	N/A	available	
Date of original definition	MDC_P003_1	STRING_TYPE	M..10	MAND	N/A	available	
date of current version	MDC_P003_2	STRING_TYPE	M..10	MAND	N/A	available	
Preferred name	MDC_P004_1.<lang>	TRANSLATABLE_STRING_TYPE	M..255	MAND	available	available	Name
Synonymous name	MDC_P004_2.<lang>	SET(0,?) OF LIST(2,2) OF STRING_TYPE	M..255	OPT	N/A	available	
Short name	MDC_P004_3.<lang>	TRANSLATABLE_STRING_TYPE	M..255	MAND	available	available	Name
Definition	MDC_P005.<lang>	TRANSLATABLE_STRING_TYPE	M..0	MAND	available	N/A	Description
Note	MDC_P007_1.<lang>	TRANSLATABLE_STRING_TYPE	M..0	OPT	available	available	Description
Definition class	MDC_P021	STRING_TYPE	M..0	MAND	N/A	available	
Data type	MDC_P022	STRING_TYPE	M..0	MAND	N/A	available	
Unit in text	MDC_P023_1	STRING_TYPE	M..0	OPT	available	available	Unit
Preferred letter symbol in text	MDC_P025_1	STRING_TYPE	M..0	OPT	N/A	available	
Code for unit	MDC_P041	STRING_TYPE	M..120	OPT	N/A	available	
Alternate ID	MDC_P101	LIST(0,?) OF STRING_TYPE	M..0	MAND	available	Available	CIM UML ID
Alternative units	MDC_P111	STRING_TYPE	M..0	OPT	available	Available	unitMultiplier
Segment	MDC_P211	STRING_TYPE	M..255	MAND	available		CIM package

Table B.4 – Meta-properties for enumeration meta-class

MMDC_P004_1.EN	MMDC_P001	MMDC_P008	MMDC_P100	MMDC_P102	Additional Information		
Name in English	Property ID	Data type	Value format	Requirement	Original CIM definition	IEC62656-1 definition	CIM name
Code	MDC_P001_12	STRING_TYPE	M..255	MAND	N/A	available	
Version number	MDC_P002_1	STRING_TYPE	M..9	OPT	N/A	available	
Revision number	MDC_P002_2	STRING_TYPE	M..3	MAND	N/A	available	
Date of original definition	MDC_P003_1	STRING_TYPE	M..10	MAND	N/A	available	
Date of current version	MDC_P003_2	STRING_TYPE	M..10	MAND	N/A	available	
Preferred name	MDC_P004_1.<lang>	TRANSLATABLE_STRING_TYPE	M..255	MAND	available	available	Name
Synonymous name	MDC_P004_2.<lang>	SET(0,?) OF LIST(2,2) OF STRING_TYPE	M..255	OPT	N/A	available	
Short name	MDC_P004_3.<lang>	TRANSLATABLE_STRING_TYPE	M..255	MAND	available	available	Name
Name icon	MDC_P004_4	STRING_TYPE	M..20	OPT	N/A	available	
Definition	MDC_P005.<lang>	TRANSLATABLE_STRING_TYPE	M..255	MAND	available	available	Description
Note	MDC_P007_1.<lang>	TRANSLATABLE_STRING_TYPE	M..0	OPT	available	available	Description
Definition class	MDC_P021	STRING_TYPE	M..255	MAND	N/A	available	
Enumerated list of terms	MDC_P043	LIST(1,?) of STRING_TYPE	M..255	MAND	N/A	available	
Enumerated code list	MDC_P044	LIST(1,?) of STRING_TYPE	M..18	MAND	N/A	available	
Alternate ID	MDC_P101	LIST(0,?) OF STRING_TYPE	M..0	MAND	available	available	CIM UML ID
Segment	MDC_P211	STRING_TYPE	M..255	MAND	available	available	CIM package

Table B.5 – Meta-properties for term meta-class

MMDC_P004_1.EN	MMDC_P001	MMDC_P008	MMDC_P100	MMDC_P102	Additional information		
Name in English	Property ID	Data type	Value format	Requirement	Original CIM definition	IEC 62656-1 definition	CIM name
Code	MDC_P001_11	STRING_TYPE	M..255	MAND	N/A	available	
Version number	MDC_P002_1	STRING_TYPE	M..9	OPT	N/A	available	
Revision number	MDC_P002_2	STRING_TYPE	M..3	MAND	N/A	available	
Date of original definition	MDC_P003_1	STRING_TYPE	M..10	MAND	N/A	available	
Date of current definition	MDC_P003_2	STRING_TYPE	M..10	MAND	N/A	available	
Preferred name	MDC_P004_1.<lang>	TRANSLATABLE_STRING_TYPE	M..255	MAND	available	available	Enumeration value
Synonymous name	MDC_P004_2.<lang>	SET(0,?) OF LIST(2,2) OF STRING_TYPE	M..255	OPT	N/A	available	
Short name	MDC_P004_3.<lang>	TRANSLATABLE_STRING_TYPE	M..255	MAND	available	available	Enumeration value
Definition	MDC_P005.<lang>	TRANSLATABLE_STRING_TYPE	M..255	MAND	available	available	Description
Note	MDC_P007_1.<lang>	TRANSLATABLE_STRING_TYPE	M..0	OPT	available	available	Description
Definition class	MDC_P021	STRING_TYPE	M..0	MAND	N/A	available	
Preferred letter symbol in text	MDC_P025_1	STRING_TYPE	M..0	OPT	available	available	Enumeration.enumerator
Alternate ID	MDC_P101	LIST(0,?) OF STRING_TYPE	M..0	MAND	available	available	CIM UML ID
Segment	MDC_P211	STRING_TYPE	M..255	MAND	available	available	CIM package

Table B.6 – Meta-properties for relation meta-class

MMDC_P004_1.EN	MMDC_P001	MMDC_P008	MMDC_P100	MMDC_P102	Additional information		
Name in English	Property ID	Data type	Value format	Requirement	Original CIM definition	IEC 62656-1 definition	CIM name
Code	MDC_P001_13	STRING_TYPE	M..255	MAND	N/A	available	
Version number	MDC_P002_1	STRING_TYPE	M..9	OPT	N/A	available	
Revision number	MDC_P002_2	STRING_TYPE	M..3	MAND	N/A	available	
Date of original definition	MDC_P003_1	STRING_TYPE	M..10	MAND	N/A	available	
Date of current definition	MDC_P003_2	STRING_TYPE	M..10	MAND	N/A	available	
Preferred name	MDC_P004_1.<lang>	TRANSLATABLE_STRING_TYPE	M..255	MAND	available	available	Package name
Short name	MDC_P004_3.<lang>	TRANSLATABLE_STRING_TYPE	M..255	MAND	available	available	Package name
Definition	MDC_P005.<lang>	TRANSLATABLE_STRING_TYPE	M..255	MAND	available	available	Package description
Note	MDC_P007_1.<lang>	TRANSLATABLE_STRING_TYPE	M..0	OPT	available	available	Package description
Alternate ID	MDC_P101	LIST(0..?) OF STRING_TYPE	M..0	MAND	available	available	CIM UML ID
Relation type	MDC_P200	STRING_TYPE	M..0	MAND	N/A	available	
Domain of the relation	MDC_P201	LIST OF ICID_STRING	M..0	OPT	N/A	available	
Domain of the function	MDC_P202	LIST OF ICID_STRING	M..0	OPT	N/A	available	
Codomain of the function	MDC_P203	STRING_TYPE	M..0	OPT	N/A	available	
Domain element type	MDC_P208	STRING_TYPE	M..0	OPT	N/A	available	
Role of the relation	MDC_P210	STRING_TYPE	M..0	OPT	N/A	available	

Annex C (informative)

XML Schema for parcellized CIM

C.1 XML schema for POM models

A set of XML(eXtensible Markup Language) schema for POM models are introduced in this annex. POM users can use this schema to represent a parcellized CIM and performs a file exchange with the original CIM RDF definitions. XML schemas listed in this annex are created according to the “RDF/XML Syntax Specification” ([1])¹.

Table C.1 shows a relation between the POM model and its XML schema. In the following Clauses C.2 to C.7, XML samples representing each type of POM model available in the parcellized CIM are elaborated.

Table C.1 – POM models and XML schemas

POM Model	XML Schema
Class	C.2
Property	C.3, C.4
Datatype	C.5
Enum	C.6
Term	C.6
Relation	C.7

C.2 XML schema for a class

C.2.1 General

Schemas listed from C.2.2 to C.2.6 are the set of schema for one CIM class. With these schemas a class of parcellized CIM model with extended attributes by POM can be described.

C.2.2 class.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:n2="http://www.w3.org/1999/02/22-rdf-syntax-ns"
xmlns:n1="http://www.w3.org/2000/01/rdf-schema#" xmlns:xs="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://www.w3.org/1999/02/22-rdf-syntax-ns">
  <xs:import namespace="http://www.w3.org/2000/01/rdf-schema#"
schemaLocation="class1.xsd"/>
  <xs:complexType name="T_RDF">
    <xs:sequence>
      <xs:element ref="n1:Class"/>
    </xs:sequence>
  </xs:complexType>
  <xs:attribute name="resource" type="xs:anyURI"/>
  <xs:attribute name="about" type="xs:anyURI"/>
  <xs:element name="RDF" type="n2:T_RDF"/>
</xs:schema>
```

¹ Numbers in square brackets refer to the Bibliography.

```
</xs:schema>
```

C.2.3 class1.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:n3="TC57-cims-ns" xmlns:n2="http://www.w3.org/1999/02/22-rdf-syntax-ns"
  xmlns:n1="http://www.w3.org/2000/01/rdf-schema#" xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:xml="http://www.w3.org/XML/1998/namespace" xmlns:n4="parcellized-cim-ns"
  targetNamespace="http://www.w3.org/2000/01/rdf-schema#" elementFormDefault="qualified">
  <xs:import namespace="parcellized-cim-ns" schemaLocation="class3.xsd"/>
  <xs:import namespace="http://www.w3.org/XML/1998/namespace"/>
  <xs:import namespace="http://www.w3.org/1999/02/22-rdf-syntax-ns"
    schemaLocation="class.xsd"/>
  <xs:import namespace="TC57-cims-ns" schemaLocation="class4.xsd"/>
  <xs:complexType name="T_Class">
    <xs:sequence>
      <xs:element name="label" type="n1:T3_label"/>
      <xs:element name="comment" type="xs:string"/>
      <xs:element ref="n3:belongsToCategory"/>
      <xs:element name="subClassOf" type="n1:T3_subClassOf"/>
      <xs:element ref="n4:ext"/>
    </xs:sequence>
    <xs:attribute ref="n2:about" use="required"/>
  </xs:complexType>
  <xs:complexType name="T3_subClassOf">
    <xs:attribute ref="n2:resource" use="required"/>
  </xs:complexType>
  <xs:complexType name="T3_label">
    <xs:simpleContent>
      <xs:extension base="xs:string">
        <xs:attribute ref="xml:lang" use="required"/>
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>
  <xs:element name="Class" type="n1:T_Class"/>
</xs:schema>
```

C.2.4 class2.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:attribute name="type" type="xs:string"/>
</xs:schema>
```

C.2.5 class3.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:xml="http://www.w3.org/XML/1998/namespace" xmlns:n4="parcellized-cim-ns"
  targetNamespace="parcellized-cim-ns" elementFormDefault="qualified">
  <xs:import namespace="http://www.w3.org/XML/1998/namespace"/>
  <xs:import schemaLocation="class2.xsd"/>
  <xs:complexType name="T_ext">
    <xs:sequence>
      <xs:element name="code" type="xs:string"/>
      <xs:element name="preferred_name" type="n4:T2_preferred_name"/>
    </xs:sequence>
  </xs:complexType>
```

```

    <xs:element name="version" type="xs:byte" />
    <xs:element name="revision" type="xs:byte" />
    <xs:element name="date_of_original_definition" type="xs:date" />
    <xs:element name="date_of_current_version" type="xs:date" />
    <xs:element name="class_type" type="xs:string" />
    <xs:element name="alternate_id" type="xs:string" />
    <xs:element name="definition" type="n4:T2_definition" />
    <xs:element name="applicable_properties" type="xs:string" />
    <xs:element name="simplified_drawing" type="xs:string" />
    <xs:element name="synonymous_name" type="xs:string" />
  </xs:sequence>
  <xs:attribute ref="type" use="required" />
</xs:complexType>
<xs:complexType name="T2_preferred_name">
  <xs:simpleContent>
    <xs:extension base="xs:string">
      <xs:attribute ref="xml:lang" use="required" />
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>
<xs:complexType name="T2_definition">
  <xs:simpleContent>
    <xs:extension base="xs:string">
      <xs:attribute ref="xml:lang" use="required" />
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>
<xs:element name="ext" type="n4:T_ext" />
</xs:schema>

```

C.2.6 class4.xsd

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:n3="TC57-cims-ns" xmlns:n2="http://www.w3.org/1999/02/22-rdf-syntax-ns"
  xmlns:xs="http://www.w3.org/2001/XMLSchema" targetNamespace="TC57-cims-ns">
  <xs:import namespace="http://www.w3.org/1999/02/22-rdf-syntax-ns"
    schemaLocation="class.xsd" />
  <xs:complexType name="T_belongsToCategory">
    <xs:attribute ref="n2:resource" use="required" />
  </xs:complexType>
  <xs:element name="belongsToCategory" type="n3:T_belongsToCategory" />
</xs:schema>

```

C.3 XML schema for a property

C.3.1 General

Schemas listed from C.3.2 to C.3.6 are the set of schema for a property of parcellized CIM model. With these schemas a property with extended attributes by POM can be described.

C.3.2 property.xsd

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:n3="TC57-cims-ns" xmlns:n1="http://www.w3.org/1999/02/22-rdf-syntax-ns"
  xmlns:n2="http://www.w3.org/2000/01/rdf-schema#" xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:n4="parcellized-cim-ns" targetNamespace="http://www.w3.org/1999/02/22-rdf-syntax-ns"
  elementFormDefault="qualified">

```

```

<xs:import namespace="parcellized-cim-ns" schemaLocation="property2.xsd"/>
<xs:import namespace="http://www.w3.org/2000/01/rdf-schema#"
schemaLocation="property4.xsd"/>
<xs:import namespace="TC57-cims-ns" schemaLocation="property3.xsd"/>
<xs:complexType name="T_RDF">
  <xs:sequence>
    <xs:element name="Property" type="n1:T2_Property"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="T2_Property">
  <xs:sequence>
    <xs:element ref="n2:label"/>
    <xs:element ref="n2:comment"/>
    <xs:element ref="n3:profile"/>
    <xs:element ref="n2:domain"/>
    <xs:element ref="n3:dataType"/>
    <xs:element ref="n4:ext"/>
  </xs:sequence>
  <xs:attribute ref="n1:about" use="required"/>
</xs:complexType>
<xs:attribute name="resource" type="xs:anyURI"/>
<xs:attribute name="about" type="xs:anyURI"/>
<xs:element name="RDF" type="n1:T_RDF"/>
</xs:schema>

```

C.3.3 property1.xsd

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:attribute name="type" type="xs:string"/>
  <xs:attribute name="lang" type="xs:string"/>
</xs:schema>

```

C.3.4 property2.xsd

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:xml="http://www.w3.org/XML/1998/namespace" xmlns:n4="parcellized-cim-ns"
targetNamespace="parcellized-cim-ns" elementFormDefault="qualified">
  <xs:import namespace="http://www.w3.org/XML/1998/namespace"/>
  <xs:import schemaLocation="property1.xsd"/>
  <xs:complexType name="T_ext">
    <xs:sequence>
      <xs:element name="code" type="xs:string"/>
      <xs:element name="preferred_name" type="n4:T3_preferred_name"/>
      <xs:element name="version" type="xs:byte"/>
      <xs:element name="revision" type="xs:byte"/>
      <xs:element name="date_of_original_definition" type="xs:date"/>
      <xs:element name="date_of_current_version" type="xs:date"/>
      <xs:element name="definition" type="n4:T3_definition"/>
      <xs:element name="data_element_type" type="xs:string"/>
      <xs:element name="alternate_id" type="xs:string"/>
      <xs:element name="graphics" type="n4:T3_graphics"/>
      <xs:element name="unit_in_text" type="n4:T3_unit_in_text"/>
      <xs:element name="alternative_units" type="n4:T3_alternative_units"/>
      <xs:element name="code_for_unit" type="n4:T3_code_for_unit"/>
      <xs:element name="preferred_letter_symbol_in_text"/>
    </xs:sequence>
  </xs:complexType>

```

```

type="n4:T3_preferred_letter_symbol_in_text"/>
    <xs:element name="synonymous_name" type="xs:string"/>
    <xs:element name="cim_package" type="xs:string"/>
  </xs:sequence>
  <xs:attribute ref="type" use="required"/>
</xs:complexType>
<xs:complexType name="T3_unit_in_text"/>
<xs:complexType name="T3_preferred_name">
  <xs:simpleContent>
    <xs:extension base="xs:string">
      <xs:attribute ref="xml:lang" use="required"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>
<xs:complexType name="T3_preferred_letter_symbol_in_text"/>
<xs:complexType name="T3_graphics"/>
<xs:complexType name="T3_definition">
  <xs:simpleContent>
    <xs:extension base="xs:string">
      <xs:attribute ref="lang" use="required"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>
<xs:complexType name="T3_code_for_unit"/>
<xs:complexType name="T3_alternative_units"/>
<xs:element name="ext" type="n4:T_ext"/>
</xs:schema>

```

C.3.5 property3.xsd

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:n3="TC57-cims-ns" xmlns:n1="http://www.w3.org/1999/02/22-rdf-syntax-ns"
  xmlns:xs="http://www.w3.org/2001/XMLSchema" targetNamespace="TC57-cims-ns">
  <xs:import namespace="http://www.w3.org/1999/02/22-rdf-syntax-ns"
    schemaLocation="property.xsd"/>
  <xs:complexType name="T_dataType">
    <xs:attribute ref="n1:resource" use="required"/>
  </xs:complexType>
  <xs:element name="profile" type="xs:string"/>
  <xs:element name="dataType" type="n3:T_dataType"/>
</xs:schema>

```

C.3.6 property4.xsd

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:n1="http://www.w3.org/1999/02/22-rdf-syntax-ns"
  xmlns:n2="http://www.w3.org/2000/01/rdf-schema#" xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:xml="http://www.w3.org/XML/1998/namespace"
  targetNamespace="http://www.w3.org/2000/01/rdf-schema#">
  <xs:import namespace="http://www.w3.org/XML/1998/namespace"/>
  <xs:import namespace="http://www.w3.org/1999/02/22-rdf-syntax-ns"
    schemaLocation="property.xsd"/>
  <xs:complexType name="T_label">
    <xs:simpleContent>
      <xs:extension base="xs:string">
        <xs:attribute ref="xml:lang" use="required"/>
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>

```

```

        </xs:simpleContent>
    </xs:complexType>
    <xs:complexType name="T_domain">
        <xs:attribute ref="n1:resource" use="required"/>
    </xs:complexType>
    <xs:element name="label" type="n2:T_label"/>
    <xs:element name="domain" type="n2:T_domain"/>
    <xs:element name="comment" type="xs:string"/>
</xs:schema>

```

C.4 XML schema for a property derived from an association end

C.4.1 General

Schemas listed from C.4.2 to C.4.6 are the set of schema for a property derived from a CIM association end. With these schemas a property with extended attributes by POM can be described.

C.4.2 association.xsd

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:n3="TC57-cims-ns" xmlns:n1="http://www.w3.org/1999/02/22-rdf-syntax-ns"
xmlns:n2="http://www.w3.org/2000/01/rdf-schema#" xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:n4="parcellized-cim-ns" targetNamespace="http://www.w3.org/1999/02/22-rdf-syntax-ns"
elementFormDefault="qualified">
    <xs:import namespace="parcellized-cim-ns" schemaLocation="association2.xsd"/>
    <xs:import namespace="http://www.w3.org/2000/01/rdf-schema#"
schemaLocation="association4.xsd"/>
    <xs:import namespace="TC57-cims-ns" schemaLocation="association3.xsd"/>
    <xs:complexType name="T_RDF">
        <xs:sequence>
            <xs:element name="Property" type="n1:T2_Property" maxOccurs="unbounded"/>
        </xs:sequence>
    </xs:complexType>
    <xs:complexType name="T2_Property">
        <xs:sequence>
            <xs:element ref="n2:label"/>
            <xs:element ref="n2:comment"/>
            <xs:element ref="n2:domain"/>
            <xs:element ref="n2:range"/>
            <xs:element ref="n3:multiplicity"/>
            <xs:element ref="n3:isAggregate" minOccurs="0"/>
            <xs:element ref="n3:inverseRoleName"/>
            <xs:element ref="n4:ext"/>
        </xs:sequence>
        <xs:attribute ref="n1:about" use="required"/>
    </xs:complexType>
    <xs:attribute name="resource" type="xs:anyURI"/>
    <xs:attribute name="about" type="xs:anyURI"/>
    <xs:element name="RDF" type="n1:T_RDF"/>
</xs:schema>

```

C.4.3 association1.xsd

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

```



```

    <xs:attribute name="type" type="xs:string" />
    <xs:attribute name="lang" type="xs:string" />
</xs:schema>

```

C.4.4 association2.xsd

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:xml="http://www.w3.org/XML/1998/namespace" xmlns:n4="parcellized-cim-ns"
  targetNamespace="parcellized-cim-ns" elementFormDefault="qualified">
  <xs:import namespace="http://www.w3.org/XML/1998/namespace" />
  <xs:import schemaLocation="association1.xsd" />
  <xs:complexType name="T_ext">
    <xs:sequence>
      <xs:element name="code" type="xs:string" />
      <xs:element name="preferred_name" type="n4:T3_preferred_name" />
      <xs:element name="version" type="xs:byte" />
      <xs:element name="revision" type="xs:byte" />
      <xs:element name="date_of_original_definition" type="xs:date" />
      <xs:element name="date_of_current_version" type="xs:date" />
      <xs:element name="definition" type="n4:T3_definition" />
      <xs:element name="data_element_type" type="xs:string" />
      <xs:element name="condition" type="n4:T3_condition" />
      <xs:element name="applicable_relations" type="n4:T3_applicable_relations" />
      <xs:element name="alternate_id" type="xs:string" />
      <xs:element name="graphics" type="n4:T3_graphics" />
      <xs:element name="unit_in_text" type="n4:T3_unit_in_text" />
      <xs:element name="alternative_units" type="n4:T3_alternative_units" />
      <xs:element name="code_for_unit" type="n4:T3_code_for_unit" />
      <xs:element name="preferred_letter_symbol_in_text"
type="n4:T3_preferred_letter_symbol_in_text" />
      <xs:element name="synonymous_name" type="xs:string" />
    </xs:sequence>
    <xs:attribute ref="type" use="required" />
  </xs:complexType>
  <xs:complexType name="T3_unit_in_text" />
  <xs:complexType name="T3_preferred_name">
    <xs:simpleContent>
      <xs:extension base="xs:string">
        <xs:attribute ref="xml:lang" use="required" />
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>
  <xs:complexType name="T3_preferred_letter_symbol_in_text" />
  <xs:complexType name="T3_graphics" />
  <xs:complexType name="T3_definition">
    <xs:simpleContent>
      <xs:extension base="xs:string">
        <xs:attribute ref="lang" use="required" />
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>
  <xs:complexType name="T3_condition" />
  <xs:complexType name="T3_code_for_unit" />
  <xs:complexType name="T3_applicable_relations" />
  <xs:complexType name="T3_alternative_units" />

```

```
<xs:element name="ext" type="n4:T_ext" />
</xs:schema>
```

C.4.5 association3.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:n3="TC57-cims-ns" xmlns:n1="http://www.w3.org/1999/02/22-rdf-syntax-ns"
xmlns:xs="http://www.w3.org/2001/XMLSchema" targetNamespace="TC57-cims-ns">
  <xs:import namespace="http://www.w3.org/1999/02/22-rdf-syntax-ns"
schemaLocation="association.xsd" />
  <xs:complexType name="T_multiplicity">
    <xs:attribute ref="n1:resource" use="required" />
  </xs:complexType>
  <xs:complexType name="T_inverseRoleName">
    <xs:attribute ref="n1:resource" use="required" />
  </xs:complexType>
  <xs:element name="multiplicity" type="n3:T_multiplicity" />
  <xs:element name="isAggregate" type="xs:boolean" />
  <xs:element name="inverseRoleName" type="n3:T_inverseRoleName" />
</xs:schema>
```

C.4.6 association4.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:n1="http://www.w3.org/1999/02/22-rdf-syntax-ns"
xmlns:n2="http://www.w3.org/2000/01/rdf-schema#" xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:xml="http://www.w3.org/XML/1998/namespace"
targetNamespace="http://www.w3.org/2000/01/rdf-schema#">
  <xs:import namespace="http://www.w3.org/XML/1998/namespace" />
  <xs:import namespace="http://www.w3.org/1999/02/22-rdf-syntax-ns"
schemaLocation="association.xsd" />
  <xs:complexType name="T_range">
    <xs:attribute ref="n1:resource" use="required" />
  </xs:complexType>
  <xs:complexType name="T_label">
    <xs:simpleContent>
      <xs:extension base="xs:string">
        <xs:attribute ref="xml:lang" use="required" />
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>
  <xs:complexType name="T_domain">
    <xs:attribute ref="n1:resource" use="required" />
  </xs:complexType>
  <xs:element name="range" type="n2:T_range" />
  <xs:element name="label" type="n2:T_label" />
  <xs:element name="domain" type="n2:T_domain" />
  <xs:element name="comment" type="xs:string" />
</xs:schema>
```

C.5 XML schema for a data type

C.5.1 General

Schemas listed from C.5.2 to C.5.6 are the set of schema for a named data type of parcellized CIM model. With these schemas a data type with extended attributes by POM can be described.

C.5.2 datatype.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:n2="http://www.w3.org/1999/02/22-rdf-syntax-ns"
  xmlns:n1="http://www.w3.org/2000/01/rdf-schema#" xmlns:xs="http://www.w3.org/2001/XMLSchema#"
  targetNamespace="http://www.w3.org/1999/02/22-rdf-syntax-ns">
  <xs:import namespace="http://www.w3.org/2000/01/rdf-schema#"
    schemaLocation="datatype1.xsd"/>
  <xs:complexType name="T_RDF">
    <xs:sequence>
      <xs:element ref="n1:Class"/>
    </xs:sequence>
  </xs:complexType>
  <xs:attribute name="resource" type="xs:anyURI"/>
  <xs:attribute name="about" type="xs:anyURI"/>
  <xs:element name="RDF" type="n2:T_RDF"/>
</xs:schema>
```

C.5.3 datatype1.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:n3="TC57-cims-ns" xmlns:n2="http://www.w3.org/1999/02/22-rdf-syntax-ns"
  xmlns:n1="http://www.w3.org/2000/01/rdf-schema#" xmlns:xs="http://www.w3.org/2001/XMLSchema#"
  xmlns:xml="http://www.w3.org/XML/1998/namespace" xmlns:n4="parcellized-cim-ns"
  targetNamespace="http://www.w3.org/2000/01/rdf-schema#" elementFormDefault="qualified">
  <xs:import namespace="parcellized-cim-ns" schemaLocation="datatype3.xsd"/>
  <xs:import namespace="http://www.w3.org/XML/1998/namespace"/>
  <xs:import namespace="http://www.w3.org/1999/02/22-rdf-syntax-ns"
    schemaLocation="datatype.xsd"/>
  <xs:import namespace="TC57-cims-ns" schemaLocation="datatype4.xsd"/>
  <xs:complexType name="T_Class">
    <xs:sequence>
      <xs:element name="label" type="n1:T3_label"/>
      <xs:element name="comment" type="xs:string"/>
      <xs:element ref="n3:profile"/>
      <xs:element ref="n3:belongsToCategory"/>
      <xs:element ref="n4:ext"/>
    </xs:sequence>
    <xs:attribute ref="n2:about" use="required"/>
  </xs:complexType>
  <xs:complexType name="T3_label">
    <xs:simpleContent>
      <xs:extension base="xs:string">
        <xs:attribute ref="xml:lang" use="required"/>
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>
  <xs:element name="Class" type="n1:T_Class"/>
</xs:schema>
```

C.5.4 datatype2.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:attribute name="type" type="xs:string"/>
  <xs:attribute name="lang" type="xs:string"/>
</xs:schema>
```

C.5.5 datatype3.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:xml="http://www.w3.org/XML/1998/namespace" xmlns:n4="parcellized-cim-ns"
  targetNamespace="parcellized-cim-ns" elementFormDefault="qualified">
  <xs:import namespace="http://www.w3.org/XML/1998/namespace"/>
  <xs:import schemaLocation="datatype2.xsd"/>
  <xs:simpleType name="T2_unit_in_text">
    <xs:restriction base="xs:string">
      <xs:enumeration value="S"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:complexType name="T_ext">
    <xs:sequence>
      <xs:element name="code" type="xs:string"/>
      <xs:element name="preferred_name" type="n4:T2_preferred_name"/>
      <xs:element name="version" type="xs:byte"/>
      <xs:element name="revision" type="xs:byte"/>
      <xs:element name="date_of_original_definition" type="xs:date"/>
      <xs:element name="date_of_current_version" type="xs:date"/>
      <xs:element name="definition" type="n4:T2_definition"/>
      <xs:element name="definition_class" type="xs:string"/>
      <xs:element name="datatype" type="xs:string"/>
      <xs:element name="alternate_id" type="xs:string"/>
      <xs:element name="unit_in_text" type="n4:T2_unit_in_text"/>
      <xs:element name="alternative_units" type="n4:T2_alternative_units"/>
      <xs:element name="code_for_unit" type="n4:T2_code_for_unit"/>
      <xs:element name="preferred_letter_symbol_in_text"
type="n4:T2_preferred_letter_symbol_in_text"/>
      <xs:element name="synonymous_name" type="xs:string"/>
    </xs:sequence>
    <xs:attribute ref="type" use="required"/>
  </xs:complexType>
  <xs:complexType name="T2_preferred_name">
    <xs:simpleContent>
      <xs:extension base="xs:string">
        <xs:attribute ref="xml:lang" use="required"/>
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>
  <xs:complexType name="T2_preferred_letter_symbol_in_text"/>
  <xs:complexType name="T2_definition">
    <xs:simpleContent>
      <xs:extension base="xs:string">
        <xs:attribute ref="lang" use="required"/>
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>
  <xs:complexType name="T2_code_for_unit"/>
  <xs:complexType name="T2_alternative_units"/>
  <xs:element name="ext" type="n4:T_ext"/>
</xs:schema>
```

C.5.6 datatype4.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
```

```

<xs:schema xmlns:n3="TC57-cims-ns" xmlns:n2="http://www.w3.org/1999/02/22-rdf-syntax-ns"
xmlns:xs="http://www.w3.org/2001/XMLSchema" targetNamespace="TC57-cims-ns">
  <xs:import namespace="http://www.w3.org/1999/02/22-rdf-syntax-ns"
schemaLocation="datatype.xsd" />
  <xs:complexType name="T_belongsToCategory">
    <xs:attribute ref="n2:resource" use="required" />
  </xs:complexType>
  <xs:element name="profile" type="xs:string" />
  <xs:element name="belongsToCategory" type="n3:T_belongsToCategory" />
</xs:schema>

```

C.6 XML schema for an enumeration and its enumerators

C.6.1 General

In the CIM RDF, an enumeration is described as an RDF class with the stereotype as "enumeration". And each of its possible enumerators is defined with <RDF:description> tag. In this standard, an enumeration is mapped to a POM enumeration and each of its possible enumerators is described as a "term" in the POM. From C.6.2 to C.6.6, one set of XML schemas for an enumeration and its enumerators of parcellized CIM model are listed.

C.6.2 enum_term.xsd

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:n2="http://www.w3.org/1999/02/22-rdf-syntax-ns"
xmlns:n1="http://www.w3.org/2000/01/rdf-schema#" xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:n3="parcellized-cim-ns" targetNamespace="http://www.w3.org/1999/02/22-rdf-syntax-ns"
elementFormDefault="qualified">
  <xs:import namespace="parcellized-cim-ns" schemaLocation="enum_term2.xsd" />
  <xs:import namespace="http://www.w3.org/2000/01/rdf-schema#"
schemaLocation="enum_term3.xsd" />
  <xs:complexType name="T_RDF">
    <xs:sequence>
      <xs:element ref="n1:Class" />
      <xs:element name="Description" type="n2:T2_Description" />
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="T4_type">
    <xs:attribute ref="n2:resource" use="required" />
  </xs:complexType>
  <xs:complexType name="T2_Description">
    <xs:sequence>
      <xs:element ref="n1:label" />
      <xs:element ref="n1:comment" />
      <xs:element name="type" type="n2:T4_type" />
      <xs:element ref="n3:ext" />
    </xs:sequence>
    <xs:attribute ref="n2:about" use="required" />
  </xs:complexType>
  <xs:attribute name="resource" type="xs:anyURI" />
  <xs:attribute name="about" type="xs:anyURI" />
  <xs:element name="RDF" type="n2:T_RDF" />
</xs:schema>

```

C.6.3 enum_term1.xsd

```

<?xml version="1.0" encoding="UTF-8"?>

```

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:attribute name="type" type="xs:string" />
</xs:schema>
```

C.6.4 enum_term2.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:xml="http://www.w3.org/XML/1998/namespace" xmlns:n3="parcellized-cim-ns"
  targetNamespace="parcellized-cim-ns" elementFormDefault="qualified">
  <xs:import namespace="http://www.w3.org/XML/1998/namespace" />
  <xs:import schemaLocation="enum_term1.xsd" />
  <xs:complexType name="T_ext">
    <xs:sequence>
      <xs:element name="code" type="xs:string" />
      <xs:element name="preferred_name" type="xs:string" />
      <xs:element name="version" type="xs:byte" />
      <xs:element name="revision" type="xs:byte" />
      <xs:element name="date_of_original_definition" type="xs:date" />
      <xs:element name="date_of_current_version" type="xs:date" />
      <xs:choice>
        <xs:element name="alternate_id" type="xs:string" />
        <xs:sequence>
          <xs:element name="definition" type="n3:T5_definition" />
          <xs:element name="enumerated_list_of_terms" type="xs:string" />
          <xs:element name="enumerated_code_list" type="xs:string" />
          <xs:element name="alternate_id" type="xs:string" />
        </xs:sequence>
      </xs:choice>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="T5_definition">
    <xs:simpleContent>
      <xs:extension base="xs:string">
        <xs:attribute ref="xml:lang" use="required" />
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>
  <xs:element name="ext" type="n3:T_ext" />
</xs:schema>
```

C.6.5 enum_term3.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:n5="TC57-cims-ns" xmlns:n2="http://www.w3.org/1999/02/22-rdf-syntax-ns"
  xmlns:n1="http://www.w3.org/2000/01/rdf-schema#" xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:xml="http://www.w3.org/XML/1998/namespace" xmlns:n3="parcellized-cim-ns"
  targetNamespace="http://www.w3.org/2000/01/rdf-schema#" elementFormDefault="qualified">
  <xs:import namespace="parcellized-cim-ns" schemaLocation="enum_term2.xsd" />
  <xs:import namespace="http://www.w3.org/XML/1998/namespace" />
  <xs:import namespace="http://www.w3.org/1999/02/22-rdf-syntax-ns"
    schemaLocation="enum_term.xsd" />
  <xs:import namespace="TC57-cims-ns" schemaLocation="enum_term4.xsd" />
  <xs:complexType name="T_label">
    <xs:simpleContent>
      <xs:extension base="xs:string">
```

```

        <xs:attribute ref="xml:lang" use="required" />
    </xs:extension>
</xs:simpleContent>
</xs:complexType>
<xs:complexType name="T_comment" />
<xs:complexType name="T_Class">
    <xs:sequence>
        <xs:element name="label" type="n1:T6_label" />
        <xs:element name="comment" type="xs:string" />
        <xs:element ref="n5:belongsToCategory" />
        <xs:element ref="n5:stereotype" />
        <xs:element ref="n3:ext" />
    </xs:sequence>
    <xs:attribute ref="n2:about" use="required" />
</xs:complexType>
<xs:complexType name="T6_label">
    <xs:simpleContent>
        <xs:extension base="xs:string">
            <xs:attribute ref="xml:lang" use="required" />
        </xs:extension>
    </xs:simpleContent>
</xs:complexType>
<xs:element name="label" type="n1:T_label" />
<xs:element name="comment" type="n1:T_comment" />
<xs:element name="Class" type="n1:T_Class" />
</xs:schema>

```

C.6.6 enum_term4.xsd

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:n5="TC57-cims-ns" xmlns:n2="http://www.w3.org/1999/02/22-rdf-syntax-ns"
  xmlns:xs="http://www.w3.org/2001/XMLSchema" targetNamespace="TC57-cims-ns">
    <xs:import namespace="http://www.w3.org/1999/02/22-rdf-syntax-ns"
      schemaLocation="enum_term.xsd" />
    <xs:complexType name="T_belongsToCategory">
        <xs:attribute ref="n2:resource" use="required" />
    </xs:complexType>
    <xs:element name="stereotype" type="xs:string" />
    <xs:element name="belongsToCategory" type="n5:T_belongsToCategory" />
</xs:schema>

```

C.7 XML schema for a CIM package

C.7.1 General

In this standard, a CIM package and its containing CIM elements are described with the "relation" model of IEC62656. In the annex, schemas listed from C.7.2 to C.7.6 are the set of schema for a CIM package of parcellized CIM model.

C.7.2 package.xsd example

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:n1="TC57-cims-ns" xmlns:n2="http://www.w3.org/1999/02/22-rdf-syntax-ns"
  xmlns:xs="http://www.w3.org/2001/XMLSchema" targetNamespace="http://www.w3.org/1999/02/22-rdf-syntax-ns">
    <xs:import namespace="TC57-cims-ns" schemaLocation="Package1.xsd" />
    <xs:complexType name="T_RDF">

```



```

        <xs:sequence>
            <xs:element ref="n1:ClassCategory" />
        </xs:sequence>
    </xs:complexType>
    <xs:attribute name="about" type="xs:anyURI" />
    <xs:element name="RDF" type="n2:T_RDF" />
</xs:schema>

```

C.7.3 package1.xsd example

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:n1="TC57-cims-ns" xmlns:n2="http://www.w3.org/1999/02/22-rdf-syntax-ns"
  xmlns:n3="http://www.w3.org/2000/01/rdf-schema#" xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:n4="parcellized-cim-ns" targetNamespace="TC57-cims-ns">
    <xs:import namespace="parcellized-cim-ns" schemaLocation="Package3.xsd" />
    <xs:import namespace="http://www.w3.org/2000/01/rdf-schema#"
  schemaLocation="Package4.xsd" />
    <xs:import namespace="http://www.w3.org/1999/02/22-rdf-syntax-ns"
  schemaLocation="Package.xsd" />
    <xs:complexType name="T_ClassCategory">
        <xs:sequence>
            <xs:element ref="n3:label" />
            <xs:element ref="n3:comment" />
            <xs:element ref="n4:ext" />
        </xs:sequence>
        <xs:attribute ref="n2:about" use="required" />
    </xs:complexType>
    <xs:element name="ClassCategory" type="n1:T_ClassCategory" />
</xs:schema>

```

C.7.4 package2.xsd example

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
    <xs:attribute name="type" type="xs:string" />
    <xs:attribute name="lang" type="xs:string" />
</xs:schema>

```

C.7.5 package3.xsd example

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:n4="parcellized-cim-ns"
  targetNamespace="parcellized-cim-ns" elementFormDefault="qualified">
    <xs:import schemaLocation="Package2.xsd" />
    <xs:complexType name="T_ext">
        <xs:sequence>
            <xs:element name="code" type="xs:string" />
            <xs:element name="preferred_name" type="xs:string" />
            <xs:element name="version" type="xs:byte" />
            <xs:element name="revision" type="xs:byte" />
            <xs:element name="date_of_original_definition" type="xs:date" />
            <xs:element name="date_of_current_version" type="xs:date" />
            <xs:element name="definition" type="n4:T2_definition" />
            <xs:element name="relation_type" type="xs:string" />
            <xs:element name="domain_of_the_realtion" type="n4:T2_domain_of_the_realtion" />
            <xs:element name="domain_of_the_function"
  type="n4:T2_domain_of_the_function" />
            <xs:element name="codomain_of_the_function"
  type="n4:T2_codomain_of_the_function" />
        </xs:sequence>
    </xs:complexType>

```



```

        <xs:element name="domain_element_type" type="n4:T2_domain_element_type"/>
        <xs:element name="role_of_relation" type="xs:string"/>
    </xs:sequence>
    <xs:attribute ref="type" use="required"/>
</xs:complexType>
<xs:complexType name="T2_domain_of_the_realtion"/>
<xs:complexType name="T2_domain_of_the_function"/>
<xs:complexType name="T2_domain_element_type"/>
<xs:complexType name="T2_definition">
    <xs:simpleContent>
        <xs:extension base="xs:string">
            <xs:attribute ref="lang" use="required"/>
        </xs:extension>
    </xs:simpleContent>
</xs:complexType>
<xs:complexType name="T2_codomain_of_the_function"/>
<xs:element name="ext" type="n4:T_ext"/>
</xs:schema>

```

C.7.6 package4.xsd example

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:n3="http://www.w3.org/2000/01/rdf-schema#"
  xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:xml="http://www.w3.org/XML/1998/namespace"
  targetNamespace="http://www.w3.org/2000/01/rdf-schema#">
    <xs:import namespace="http://www.w3.org/XML/1998/namespace"/>
    <xs:complexType name="T_label">
        <xs:simpleContent>
            <xs:extension base="xs:string">
                <xs:attribute ref="xml:lang" use="required"/>
            </xs:extension>
        </xs:simpleContent>
    </xs:complexType>
    <xs:element name="label" type="n3:T_label"/>
    <xs:element name="comment" type="xs:string"/>
</xs:schema>

```

Annex D (informative)

XML examples for parcellized CIM

D.1 XML example for a CIM class with extended attributes

In this annex, a CIM class “PowerSystemResource” with POM extended attributes is described with the XML schemas represented in Clause C.2. The method of how the XML schema described in Annex C can be applied to describe a parcellized CIM model can be referred to. The CIM RDF schema for this class is from IEC 61970-501:2006.

```
<?xml version="1.0" encoding="UTF-8" ?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#" xmlns:cims="TC57-cims-ns"
  xmlns:pom="parcellized-cim-ns">
  <!-- sample of CIM original attributes-->
  <rdfs:Class rdf:about="http://iec.ch/TC57/2003/CIMschema-cim10#PowerSystemResource">
    <rdfs:label xml:lang="en">PowerSystemResource</rdfs:label>
    <rdfs:comment>
```

A power system resource can be an item of equipment such as a switch, an equipment container containing many individual items of equipment such as a substation, or an organisational entity such as sub-control area. Power system resources can have measurements associated.

```
</rdfs:comment>
  <cims:belongsToCategory rdf:resource="http://iec.ch/TC57/2003/CIM-schema-cim10#Core"/>
  <rdfs:subClassOf rdf:resource="http://iec.ch/TC57/2003/CIMschema-cim10#identifiedObject"/>
  <!-- sample of extended attributes -->
  <pom:ext type="class">
    <pom:code>CIM_CLASS_1</pom:code>
    <pom:preferred_name xml:lang="en">name improved in IEC CDD</pom:preferred_name>
    <pom:version>001</pom:version>
    <pom:revision>001</pom:revision>
    <pom:date_of_original_definition>2010-10-02</pom:date_of_original_definition>
    <pom:date_of_current_version>2011-04-22</pom:date_of_current_version>
    <pom:class_type>ITEM_CLASS</pom:class_type>
    <pom:alternate_id>_0340C084_3759_40cc_AAF2_386101DE842F</pom:alternate_id>
    <pom:definition xml:lang="en">definition improved by IEC CDD.</pom:definition>
    <pom:applicable_properties>[CIM_PROP_1,CIM_PROP_2]</pom:applicable_properties>
    <pom:simplified_drawing>drawing file uri</pom:simplified_drawing>
    <pom:synonymous_name>{(syn1,"en"),(syn2,"ja")}</pom:synonymous_name>
  </pom:ext>
</rdfs:Class>
</rdf:RDF>
```

D.2 XML example for a property with extended attributes

In this annex, a CIM property named “yPerSection” with extended attributes is described with the XML schemas represented in Clause C.3. The CIM RDF schema for this property is from IEC 61970-501:2006.

```
<?xml version="1.0" encoding="UTF-8" ?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xmlns:cims="TC57-cims-ns"
  xmlns:pom="parcellized-cim-ns">
```

```

xmlns:cims="TC57-cims-ns" xmlns:pom="parcellized-cim-ns" >
  <rdf:Property rdf:about="http://iec.ch/TC57/2003/CIM-schema-
cim10#Compensator.yPerSection">
    <rdfs:label xml:lang="en">yPerSection</rdfs:label>
    <rdfs:comment>For a capacitor bank, the admittance of each switchable section.
Calculated using the MVar per section and corrected for network voltage.
</rdfs:comment>
    <cims:profile>Nerc</cims:profile>
    <rdfs:domain rdf:resource="http://iec.ch/TC57/2003/CIM-schemacim10#Compensator"/>
    <cims:dataType rdf:resource="http://iec.ch/TC57/2003/CIM-
schemacim10#Admittance"/>
    <pom:ext type="property">
      <pom:code>AKH000</pom:code>
      <pom:preferred_name xml:lang="en">name improved by IEC
CDD</pom:preferred_name>
      <pom:version>001</pom:version>
      <pom:revision>001</pom:revision>
      <pom:date_of_original_definition>2010-10-02</pom:date_of_original_definition>
      <pom:date_of_current_version>2011-04-22</pom:date_of_current_version>
      <pom:definition lang="en">definitions improved by IEC CDD</pom:definition>
      <pom:data_element_type>NON_DEPENDENT_P_DET</pom:data_element_type>
      <pom:alternate_id>BE30DEA0_E5FB_4fb9_AD03_E9DDB91857B5</pom:alternate_id>
      <pom:graphics></pom:graphics>
      <pom:unit_in_text></pom:unit_in_text>
      <pom:alternative_units></pom:alternative_units>
      <pom:code_for_unit></pom:code_for_unit>
      <pom:preferred_letter_symbol_in_text></pom:preferred_letter_symbol_in_text>
      <pom:synonymous_name>{(syn1,"en"),(syn2,"ja")}</pom:synonymous_name>
      <pom:cim_package>Wires</pom:cim_package>
    </pom:ext>
  </rdf:Property>
</rdf:RDF>

```

D.3 XML example for a CIM association and aggregation with extended attributes

In this annex, the association example shown in Figure 3 is described with the XML schemas represented in Clause C.4. The CIM RDF schema for this association and aggregation is in conformance with IEC 61970-501:2006.

```

<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns"
xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#" xmlns:cims="TC57-cims-ns"
xmlns:pom="parcellized-cim-ns" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

  <rdf:Property rdf:about="http://iec.ch/TC57/2003/CIM-schemacim10#EquipmentContainer.Equipments">
    <rdfs:label xml:lang="en">Equipment</rdfs:label>
    <rdfs:comment>The association is used in the naming hierarchy</rdfs:comment>
    <rdfs:domain rdf:resource="http://iec.ch/TC57/2003/CIM-schemacim10#EquipmentContainer"/>
    <rdfs:range rdf:resource="http://iec.ch/TC57/2003/CIM-schemacim10#Equipment"/>
    <cims:multiplicity rdf:resource="http://iec.ch/TC57/1999/rdfs-schema-extensions-19990926#M:0..n"/>
    <cims:inverseRoleName rdf:resource="http://iec.ch/TC57/2003/CIM-schema-
cim10#Equipment.EquipmentContainer"/>
    <pom:ext type="property">
      <pom:code>CIM_PROPERTY_3</pom:code>

```

```

        <pom:preferred_name xml:lang="en">name improved by IEC
CDD</pom:preferred_name>
        <pom:version>001</pom:version>
        <pom:revision>001</pom:revision>
        <pom:date_of_original_definition>2010-10-02</pom:date_of_original_definition>
        <pom:date_of_current_version>2011-04-22</pom:date_of_current_version>
        <pom:definition lang="en">definitions improved by IEC CDD</pom:definition>
        <pom:data_element_type>INVERSIBLE_P_DET</pom:data_element_type>
        <pom:condition></pom:condition>
        <pom:applicable_relations></pom:applicable_relations>
        <pom:alternate_id>68BA73EF_8533_48ba_964A_DE0F722292D2-
B</pom:alternate_id>
        <pom:graphics></pom:graphics>
        <pom:unit_in_text></pom:unit_in_text>
        <pom:alternative_units></pom:alternative_units>
        <pom:code_for_unit></pom:code_for_unit>
        <pom:preferred_letter_symbol_in_text></pom:preferred_letter_symbol_in_text>
        <pom:synonymous_name>{(syn1,"en"),(syn2,"ja")}</pom:synonymous_name>

    </pom:ext>
</rdf:Property>

<rdf:Property rdf:about="http://iec.ch/TC57/2003/CIM-schemacim10#Equipment.EquipmentContainer">
<rdfls:label xml:lang="en">EquipmentContainer</rdfls:label>
<rdfls:comment>Container of this equipment.</rdfls:comment>
<rdfls:domain rdf:resource="http://iec.ch/TC57/2003/CIM-schemacim10#Equipment"/>
<rdfls:range rdf:resource="http://iec.ch/TC57/2003/CIM-schemacim10#EquipmentContainer"/>
<cims:multiplicity rdf:resource="http://iec.ch/TC57/1999/rdflschema-extensions-19990926#M:0..1"/>
<cims:isAggregate>true</cims:isAggregate>
<cims:inverseRoleName rdf:resource="http://iec.ch/TC57/2003/CIMschema-
cim10#EquipmentContainer.Equipment"/>
    <pom:ext type="property">
        <pom:code>CIM_PROPERTY_4</pom:code>
        <pom:preferred_name xml:lang="en">name improved by IEC CDD</pom:preferred_name>
        <pom:version>001</pom:version>
        <pom:revision>001</pom:revision>
        <pom:date_of_original_definition>2010-10-02</pom:date_of_original_definition>
        <pom:date_of_current_version>2011-04-22</pom:date_of_current_version>
        <pom:definition lang="en">definitions improved by IEC CDD</pom:definition>
        <pom:data_element_type>INVERSIBLE_P_DET</pom:data_element_type>
        <pom:condition></pom:condition>
        <pom:applicable_relations></pom:applicable_relations>
        <pom:alternate_id>68BA73EF_8533_48ba_964A_DE0F722292D2-
A</pom:alternate_id>
        <pom:graphics></pom:graphics>
        <pom:unit_in_text></pom:unit_in_text>
        <pom:alternative_units></pom:alternative_units>
        <pom:code_for_unit></pom:code_for_unit>
        <pom:preferred_letter_symbol_in_text></pom:preferred_letter_symbol_in_text>
        <pom:synonymous_name>{(synA,"en"),(synB,"ja")}</pom:synonymous_name>
    </pom:ext>
</rdf:Property>
</rdf:RDF>

```

D.4 XML example for a CIM data type with extended attributes

In this annex, a user defined data type “Admittance” with extended attributes is described with the XML schemas represented in Clause C.5. The CIM RDF schema for this data type is from IEC 61970-501:2006.

```
<?xml version="1.0" encoding="UTF-8" ?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns"
xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#" xmlns:cims="TC57-cims-ns"
xmlns:pom="parcellized-cim-ns">

  <rdfs:Class rdf:about="http://iec.ch/TC57/2003/CIM-schemacim10#Admittance">
    <rdfs:label xml:lang="en">Admittance</rdfs:label>
    <rdfs:comment>Ratio of current to voltage.</rdfs:comment>
    <cims:profile>Nerc</cims:profile>
    <cims:belongsToCategory rdf:resource="http://iec.ch/TC57/2003/CIM-schema-cim10#Domain"/>
    <pom:ext type="datatype">
      <pom:code>CIM_DATATYPE_1</pom:code>
      <pom:preferred_name xml:lang="en">name improved in IEC CDD</pom:preferred_name>
      <pom:version>001</pom:version>
      <pom:revision>001</pom:revision>
      <pom:date_of_original_definition>2010-10-02</pom:date_of_original_definition>
      <pom:date_of_current_version>2011-04-22</pom:date_of_current_version>
      <pom:definition lang="en">definitions improved by IEC CDD</pom:definition>
      <pom:definition_class>Universal</pom:definition_class>
      <pom:datatype>REAL_MEASURE_TYPE</pom:datatype>
      <pom:alternate_id>30AC6821_E972_456a_8F64_0765D7361767</pom:alternate_id>
      <pom:unit_in_text>S</pom:unit_in_text>
      <pom:alternative_units></pom:alternative_units>
      <pom:code_for_unit></pom:code_for_unit>
      <pom:preferred_letter_symbol_in_text></pom:preferred_letter_symbol_in_text>
      <pom:synonymous_name>{(syn1,"en"),(syn2,"ja")}</pom:synonymous_name>
    </pom:ext>
  </rdfs:Class>
</rdf:RDF>
```

D.5 XML example for a CIM enumeration and its enumerators with extended attributes

In this annex, a CIM enumeration: “GeneratorOperatingMode” and one of its enumerators “off” are described with the XML schemas represented in Clause C.6. The CIM RDF schema for this enumeration and enumerator is from IEC 61970-501:2006.

```
<?xml version="1.0" encoding="UTF-8" ?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns"
xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#" xmlns:cims="TC57-cims-ns"
xmlns:pom="parcellized-cim-ns" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" >
  <!-- enumeration definition as a RDF class -->
  <rdfs:Class rdf:about="http://iec.ch/TC57/2003/CIM-schemacim10#GeneratorOperatingMode">
    <rdfs:label xml:lang="en">GeneratorOperatingMode</rdfs:label>
    <rdfs:comment>Operating mode for secondary generator control, e.g.: Unavailable, Manual,
Fixed, Load Frequency Control, AGC, EDC, RPN, MRN, or REG</rdfs:comment>
    <cims:belongsToCategory rdf:resource="http://iec.ch/TC57/2003/CIM-schema-
cim10#Domain"/>
    <cims:stereotype>"Enumeration"</cims:stereotype>
```

```

<!--extended attributes by POM-->
<pom:ext type="enuemration">
  <pom:code>CIM_enumeration_1</pom:code>
  <pom:preferred_name>name improved by IEC CDD</pom:preferred_name>

  <pom:version>001</pom:version>
  <pom:revision>001</pom:revision>
  <pom:date_of_original_definition>2010-10-02</pom:date_of_original_definition>
  <pom:date_of_current_version>2011-04-22</pom:date_of_current_version>
  <pom:definition xml:lang="en">improved definition for an enumeration by the IEC
CDD</pom:definition>

  <pom:enumerated_list_of_terms>[Off,AGC,REG,LFC,Fixed,EDC,Manual,MRN]</pom:enumerated_list_of
_terms>

  <pom:enumerated_code_list>[CIM_term_1,...]</pom:enumerated_code_list>

  <pom:alternate_id>EAID_93CE1FF6_33A4_475d_9FFA_746224F1591E</pom:alternate_id>
</pom:ext>
</rdfs:Class>
<!-- enumerator definition for an enuемration.
all possible enumerators for above enumeration are: (AGC,REG,LFC,Off,Fixed,EDC,Manual,MRN)-->
<rdf:Description rdf:about="http://iec.ch/TC57/2003/CIM-
schemacim10#GeneratorOperatingMode.Off">
  <rdfs:label xml:lang="en">Off</rdfs:label>
  <rdfs:comment/>
  <rdf:type rdf:resource="http://iec.ch/TC57/2003/CIM-
schemacim10#GeneratorOperatingMode"/>
  <!--extended attributes by POM-->
  <pom:ext type="term">
    <pom:code>CIM_term1_1</pom:code>
    <pom:preferred_name>name improved by IEC CDD</pom:preferred_name>
    <pom:version>001</pom:version>
    <pom:revision>001</pom:revision>
    <pom:date_of_original_definition>2010-10-02</pom:date_of_original_definition>
    <pom:date_of_current_version>2011-04-22</pom:date_of_current_version>

    <pom:alternate_id>EAID_C4247525_8906_4f70_A24D_00AF77AEBEB8</pom:alternate_id>
  </pom:ext>
</rdf:Description>
</rdf:RDF>

```

D.6 XML example for a CIM package with extended attributes

In this annex, a CIM package: "Core" with extended attributes is described with the XML schemas represented in Clause C.7. The CIM RDF schema for this package is from IEC 61970-501:2006.

```

<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns"
xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#" xmlns:cims="TC57-cims-ns"
xmlns:pom="parcellized-cim-ns" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" >
  <cims:ClassCategory rdf:about="http://iec.ch/TC57/2003/CIM-schemacim10#Core">
    <rdfs:label xml:lang="en">Core</rdfs:label>
    <rdfs:comment>Contains the core PowerSystemResource and ConductingEquipment

```

entities shared by all applications plus common collections of Those entities. Not all applications require all the Core entities.

```
</rdfs:comment>
  <pom:ext type="package">
    <pom:code>CIM_PACKAGE_1</pom:code>
    <pom:preferred_name>name improved by IEC CDD</pom:preferred_name>
    <pom:version>001</pom:version>
    <pom:revision>001</pom:revision>
    <pom:date_of_original_definition>2010-10-02</pom:date_of_original_definition>
    <pom:date_of_current_version>2011-04-22</pom:date_of_current_version>
    <pom:definition lang="en">definitions improved by IEC CDD</pom:definition>
    <pom:relation_type>function</pom:relation_type>
    <pom:domain_of_the_realtion></pom:domain_of_the_realtion>
    <pom:domain_of_the_function></pom:domain_of_the_function>
    <pom:codomain_of_the_function></pom:codomain_of_the_function>
    <pom:domain_element_type></pom:domain_element_type>
    <pom:role_of_relation>CIM Package</pom:role_of_relation>
  </pom:ext>
</cims:ClassCategory>
</rdf:RDF>
```

IECNORM.COM : Click to view the full PDF of IEC 62656-3:2015

Annex E

(informative)

Property reference between CIM and IEC CDD – “Power transformer” definitions in both IEC CDD and CIM

In CIM, necessary resources for power systems are defined for energy management systems. In the IEC CDD (Common data dictionary), electric/electronic components and materials used in electrotechnical equipment and systems are defined.

In both series of standards, there are some identical classes. For example, the power transformer, switch etc. However, even for a class with the same name, there are very few similar properties for they are defined from different points of view.

Due to this standard, properties of a class defined in one series of the standards can be referred to by the same class defined in the other series of standards. In another word, properties of a class in IEC CDD can be enriched by properties of the same class in the CIM, and vice versa.

In this annex, the “power transformer” class is used as the example to explain how the IEC CDD and CIM can become a supplement to each other. Table E.1 and Table E.2 list the properties, defined separately, for the “power transformer” class in the IEC CDD and in IEC 61970-301:2013.

In the IEC CDD, the class of “power transformer” has the following properties as shown in Table E.1.

IECNORM.COM : Click to view the full PDF of IEC 62656-3:2015

Table E.1 – Native properties of Power transformer class in the IEC CDD

MDC_P001_6 Code	MDC_P002_1 Version	MDC_P002_2 Revision	MDC_P004_1.en PreferredName.EN	MDC_P005.en Definition.EN	MDC_P023 Unit in text	MDC_P022 Data_type ENUM_CODE_TYPE(ASA022)	MDC_P006_1 DefinitionSource
AAE003	1	2	adjustability type	Code of the type of mechanical adjustability of a capacitor, resistor, inductor or transformer.			
AAE158	1	2	clearance to earth	The minimum clearance to earth (in m) of a power transformer	m	LEVEL(MIN) OF REAL_MEASURE_TYPE	IEC 60050 (441-17-33) (1984)
AAE159	1	2	creepage distance	The minimum creepage distance (in m) between live and non-live parts of a power transformer or connector	m	LEVEL(MIN) OF REAL_MEASURE_TYPE	IEC 60742 (2.48) (1983)
AAE160	1	2	nominal output current	The nominal output current (in A) of a power transformer, at nominal input voltage and nominal frequency	A	LEVEL(NOM) OF INT_MEASURE_TYPE	IEC 60742 (2.14) (1983)
AAE163	1	2	input voltage	The value as specified by level (miNoMax) of the rms sinusoidal input voltage (in V) of a power transformer	V	LEVEL(MIN,MAX,NOM) OF INT_MEASURE_TYPE	
AAE165	1	2	output power	The nominal rms power (in VA) of a power transformer	VA	LEVEL(NOM) OF REAL_MEASURE_TYPE	IEC 60076-1 (3.4.6) (1976)
AAE166	1	2	operating frequency	The value as specified by level (miNoMax) of the frequency (in Hz) of a power transformer.	Hz	LEVEL(MIN,MAX,NOM) OF REAL_MEASURE_TYPE	IEC 60076-1 (3.4.5) (1976)
AAF048	1	2	number of primary coils	The number of primary coils of a power transformer or relay	1	LEVEL(NOM) OF INT_MEASURE_TYPE	
AAF098	1	2	power transformer application	Code indicating the application of a fixed power transformer		ENUM_CODE_TYPE(ASA162)	
AAF099	1	2	number of secondary coils	The number of secondary coils of a power transformer	1	LEVEL(NOM) OF INT_MEASURE_TYPE	

In IEC 61970-301:2013, the class of “PowerTransformer” has the following properties.

Table E.2 – Native properties of Power transformer class in IEC 61970-301:2013

name	description	type
bmagSat	Core shunt magnetizing susceptance in the saturation region	Percent
magBaseU	The reference voltage at which the magnetizing saturation measurements were made	Voltage
magSatFlux	Core magnetizing saturation curve knee flux level.	Percent

IECNORM.COM : Click to view the full PDF of IEC 62656-3:2015

Annex F

(informative)

Four Layer architecture of UML and POM – Four layer architecture of MOF (meta object facility)

As the specifications of UML model and POM model are consistent with MOF which is defined in ISO/IEC19502:2005, the four layer architectures of MOF are introduced briefly in this annex in order to make the understanding of this standard easier.

The general concept of the MOF is shown in Figure 6.1 of ISO/IEC 19502:2005. The upper layer gives definitions and constraints of the lower layer. See IEC 62656-1 and ISO/IEC19502:2005 for details.

This standard defines the mapping rules between UML meta-model and POM meta-model, which are defined in the M2 layer. With applying the mapping rules to the CIM of M1 layer, the output of “Parcellized CIM” can be generated. Anyone can develop his own “Parcellized CIM Generator” according to the specifications in this standard.

IECNORM.COM : Click to view the full PDF of IEC 62656-3:2015

Annex G (informative)

POM Relation usage in IEC 62656-3

G.1 General

In this standard, the Relation object of POM is applied to describe three kinds of objects as following;

- Meta-model mapping rules between UML and POM(M2)
- UML association(M2)
- CIM Package and its elements(M1)

In this annex, three kinds of relation usage are explained.

G.2 Meta-model mapping rules between UML and POM(M2)

The relation object is used to describe the meta-model mapping rules between UML and POM. Details of the mapping rules are explained in Annex A.

G.3 UML Association(M1)

The relation model is used to define the association of the UML model. For each association, its association end members shall be listed in the “Domain of the relation” attribute. Details are explained in Table 5.

G.4 Model level relation definition(M1)

In the M1 layer, the relation is used to define the CIM models which includes;

- CIM Package
- Element collections of a package, such as class collection, property collection, data type collection etc.

Details of the CIM package and its elements in the relation object are explained in Table 12.

Annex H (informative)

Parcellized CIM files access – URL for the parcellized CIM files

Parcellized CIM files generated according to this standard are available from the following URL.

http://collaboration.iec.ch/LotusQuickr/iec_tc3/PageLibraryC12573B600379067.nsf/h_Toc/3B4972E82F74583C12572DC00355A52/?OpenDocument

NOTE These files are only accessible by IEC members.

IECNORM.COM : Click to view the full PDF of IEC 62656-3:2015

Bibliography

- [1] RDF/XML Syntax Specification (Revised), W3C Recommendation 10 February 2004, available from <<http://www.w3.org/TR/rdf-syntax-grammar/>>
- [2] IEC/TS 62656-2, *Standardized product ontology register and transfer by spreadsheets – Part 2: Application guide for use with the IEC Common Data Dictionary (CDD)*
- [3] ISO/IEC 19505-1:2012, *Information technology – Object Management Group Unified Modeling Language (OMG UML) – Part 1: Infrastructure*
- [4] ISO/IEC 19505-2:2012, *Information technology – Object Management Group Unified Modeling Language (OMG UML) – Part 2: Superstructure*
- [5] ISO/IEC 19502:2005, *Information technology – Meta Object Facility (MOF)*

IECNORM.COM : Click to view the full PDF of IEC 62656-3:2015

IECNORM.COM : Click to view the full PDF of IEC 62656-3:2015

SOMMAIRE

AVANT-PROPOS	97
INTRODUCTION.....	99
1 Domaine d'application	100
2 Références normatives	101
3 Termes, définitions et abréviations	101
3.1 Termes et définitions	101
3.2 Abréviations.....	103
4 Règles et principes génériques pour la conception d'interface CIM.....	103
4.1 Analyse comparative des structures et fonctionnalités du POM et du CIM.....	103
4.2 Série de normes CIM	105
5 Règles de transformation du CIM vers le POM	105
5.1 Règles générales.....	105
5.2 Classe du CIM	106
5.2.1 Généralités.....	106
5.2.2 Name ("Nom")	107
5.2.3 Description	108
5.2.4 Généralisation-Spécialisation	108
5.3 Attributs de classe du CIM	108
5.3.1 Généralités.....	108
5.3.2 Name ("Nom")	109
5.3.3 Attribut natif.....	109
5.3.4 Type	109
5.3.5 Description	109
5.4 Association du CIM	110
5.4.1 Généralités.....	110
5.4.2 Structure d'extrémité d'association du CIM.....	110
5.4.3 Règle de transformation générale pour association	111
5.4.4 Règle de transformation générale pour extrémité d'association	112
5.5 Classe Enumeration ("énumération") du CIM	117
5.5.1 Généralités.....	117
5.5.2 Name ("Nom")	118
5.5.3 Description	118
5.6 Attribut de Enumeration ("énumération") du CIM	118
5.6.1 Généralités.....	118
5.6.2 Name ("Nom")	118
5.6.3 Description	119
5.7 Data type ("Type de données")	119
5.7.1 Datatype de base du CIM	119
5.7.2 Data types ("types de données") nommés du CIM	120
5.8 Paquetage	121
5.8.1 Généralités.....	121
5.8.2 Objet relation du POM	123
5.8.3 Paquetages CIM décrits en relation du POM.....	124
6 Description du CIM au format IEC 62656.....	127
6.1 Généralités	127
6.2 Class ("Classe").....	128

6.2.1	Généralités	128
6.2.2	Code	128
6.2.3	Version number ("Numéro de version")	128
6.2.4	Revision number ("Numéro de révision")	129
6.2.5	Date of original definition ("Date de la définition originale")	129
6.2.6	Date of current version ("Date de la version actuelle")	129
6.2.7	Preferred name ("Nom préférentiel")	129
6.2.8	Short name ("Nom court")	130
6.2.9	Definition ("Définition")	130
6.2.10	Note	130
6.2.11	Superclass ("Superclasse")	131
6.2.12	Class type (Type de classe)	131
6.2.13	Alternate ID ("ID alternatif")	131
6.2.14	CIM package ("Paquetage du CIM")	132
6.2.15	Applicable properties ("Propriétés applicables")	132
6.3	Property ("Propriété")	132
6.3.1	Généralités	132
6.3.2	Property data element type ("Type d'élément de données de propriété")	132
6.3.3	Definition class ("Classe de définition")	133
6.3.4	Data type ("Type de données")	133
6.3.5	Applicable relations ("Relations applicables")	133
6.3.6	Condition	133
6.3.7	Unit in text ("Unité en texte")	134
6.3.8	Alternative units ("Unités alternatives")	134
6.4	Data type ("Type de données")	134
6.4.1	Généralités	134
6.4.2	Definition class ("Classe de définition")	134
6.4.3	Unit in text ("Unité en texte")	135
6.4.4	Alternative units ("Unités alternatives")	135
6.4.5	Data type ("Type de données")	135
6.5	Enumeration ("Énumération")	135
6.5.1	Généralités	135
6.5.2	Definition class ("Classe de définition")	136
6.5.3	Enumerated list of terms ("Liste énumérée de termes")	136
6.5.4	Enumerated code list ("Liste de codes énumérés")	136
6.6	Métaclass "Term" ("Terme")	136
6.6.1	Généralités	136
6.6.2	Preferred letter symbol in text ("Symbole littéral préférentiel en texte")	136
6.7	Relation	137
6.7.1	Généralités	137
6.7.2	Relation type ("Type de relation")	137
6.7.3	Domain of the relation ("Domaine de la relation")	137
6.7.4	Domain of the function ("Domaine de la fonction")	138
6.7.5	Codomain of the function ("Co-domaine de la fonction")	138
6.7.6	Domain element type ("Type d'élément de domaine")	138
6.7.7	Role of the relation ("Rôle de la relation")	139
7	Schéma XML pour des instances du CIM mises à jour dans le CDD de l'IEC	139
7.1	Généralités	139
7.2	Principes	139

8	Mécanisme de contrôle des versions dans le POM pour le CIM paqueté.....	141
Annexe A (normative) Règles de mise en correspondance du CIM vers le POM –		
	Règles de mise en correspondance décrites dans la métaclasse "relation" du POM.....	143
Annexe B (normative) CIM en format IEC 62656 – Métapropriétés normatives du CIM		
	dans le modèle IEC 62656.....	154
Annexe C (informative) Schéma XML pour le CIM paqueté		
C.1	Schéma XML pour les modèles POM	162
C.2	Schéma XML pour une classe.....	162
C.2.1	Généralités.....	162
C.2.2	class.xsd	162
C.2.3	class1.xsd	163
C.2.4	class2.xsd	163
C.2.5	class3.xsd	163
C.2.6	class4.xsd	164
C.3	Schéma XML pour une propriété.....	164
C.3.1	Généralités.....	164
C.3.2	property.xsd	164
C.3.3	property1.xsd.....	165
C.3.4	property2.xsd.....	165
C.3.5	property3.xsd.....	166
C.3.6	property4.xsd.....	166
C.4	Schéma XML pour une propriété issue d'une extrémité d'association.....	167
C.4.1	Généralités.....	167
C.4.2	association.xsd.....	167
C.4.3	association1.xsd	168
C.4.4	association2.xsd	168
C.4.5	association3.xsd	169
C.4.6	association4.xsd	169
C.5	Schéma XML pour un type de données ("data type").....	170
C.5.1	Généralités.....	170
C.5.2	datatype.xsd	170
C.5.3	datatype1.xsd	170
C.5.4	datatype2.xsd	171
C.5.5	datatype3.xsd	171
C.5.6	datatype4.xsd	172
C.6	Schéma XML pour une énumération ("enumeration") et ses énumérateurs.....	172
C.6.1	Généralités.....	172
C.6.2	enum_term.xsd	172
C.6.3	enum_term1.xsd	173
C.6.4	enum_term2.xsd	173
C.6.5	enum_term3.xsd	173
C.6.6	enum_term4.xsd	174
C.7	Schéma XML pour un paquetage du CIM ("CIM package").....	174
C.7.1	Généralités.....	174
C.7.2	package.xsd example	175
C.7.3	package1.xsd example	175
C.7.4	package2.xsd example	175
C.7.5	package3.xsd example	175
C.7.6	package4.xsd example	176

Annexe D (informative) Exemples XML pour le CIM paqueté.....	177
D.1 Exemple XML pour une classe du CIM avec attributs étendus.....	177
D.2 Exemple XML pour une propriété avec attributs étendus.....	177
D.3 Exemple XML pour une association du CIM et une agrégation avec attributs étendus.....	178
D.4 Exemple XML pour un type de données du CIM avec attributs étendus.....	180
D.5 Exemple XML pour une énumération du CIM et ses énumérateurs avec attributs étendus.....	180
D.6 Exemple XML pour un paquetage du CIM avec attributs étendus.....	181
Annexe E (informative) Référence de propriété entre le CIM et le CDD de l'IEC – Définitions de "Power transformer" ("Transformateur de puissance") dans le CDD de l'IEC et le CIM	183
Annexe F (informative) Architecture à quatre couches UML et POM – Architecture à quatre couches de MOF (meta object facility)	187
Annexe G (informative) Utilisation de la relation du POM dans l'IEC 62656-3.....	188
G.1 Généralités	188
G.2 Règles de mise en correspondance de métamodèle entre UML et POM(M2).....	188
G.3 UML Association(M1) (Association UML)	188
G.4 Model level relation definition(M1) (Définition de la relation de niveau du modèle)	188
Annexe H (informative) Accès aux fichiers du CIM paqueté – URL pour les fichiers du CIM paqueté	189
Bibliographie.....	190
Figure 1 – Exemple de généralisation du CIM (provenant de la Figure 2 de l'IEC 61970-301:2013)	108
Figure 2 – Exemple d'association simple (provenant la figure 3 de l'IEC 61970-301:2013)	114
Figure 3 – Exemple d'agrégation (provenant de la figure 4 de l'IEC 61970-301:2013).....	114
Figure 4 – Définition de type d'élément de données de propriété (property data element type definition).....	116
Figure 5 – Exemple de paquetage du CIM (provenant de la Figure 1 de l'IEC 61970-301:2013)	123
Figure 6 – Explication du modèle de relation du POM en UML	124
Figure 7 – Instances de classe du CIM paqueté	140
Figure 8 – Mécanisme de référence conceptuel avec ID dans le POM	142
Tableau 1 – Mise en correspondance des objets CIM et des objets du POM	106
Tableau 2 – Correspondance entre une classe du CIM en UML et une classe du POM	107
Tableau 3 – Mise en correspondance d'un attribut de classe du CIM et d'une propriété du POM	109
Tableau 4 – Structure d'extrémité d'association du CIM.....	111
Tableau 5 – Associations du CIM définies comme relations du POM.....	111
Tableau 6 – Mise en correspondance d'une extrémité d'association du CIM et d'une propriété du POM	113
Tableau 7 – Extrémités d'association du CIM décrites comme propriétés du POM	114
Tableau 8 – Mise en correspondance d'une énumération du CIM et d'une énumération du POM	118

Tableau 9 – Mise en correspondance d'un attribut d'énumération du CIM et d'un terme du POM	118
Tableau 10 – Mise en correspondance du data type de base du CIM et du data type simple du POM	119
Tableau 11 – Mise en correspondance du type de données du CIM et d'un type de données nommé dans le POM	120
Tableau 12 – Mise en correspondance d'un paquetage du CIM et d'une relation du POM	122
Tableau 13 – Paquetage du CIM décrit comme relation du POM	125
Tableau A.1 – Règle de mise en correspondance d'une classe du CIM vers une classe du POM	144
Tableau A.2 – Règle de mise en correspondance d'un attribut de classe du CIM vers une propriété du POM	145
Tableau A.3 – Règle de mise en correspondance d'une association du CIM vers une relation du POM	146
Tableau A.4 – Règle de mise en correspondance d'une extrémité d'association du CIM vers une propriété du POM	147
Tableau A.5 – Règle de mise en correspondance entre énumération du CIM et énumération du POM	149
Tableau A.6 – Règle de mise en correspondance entre un attribut d'énumération du CIM et un terme du POM	149
Tableau A.7 – Règle de mise en correspondance entre un data type du CIM et un data type du POM	150
Tableau A.8 – Règle de mise en correspondance entre Paquetage du CIM et relation du POM	151
Tableau A.9 – Règle de mise en correspondance de propriété du POM vers agrégation/association/attribut de classe du CIM	153
Tableau B.1 – Métapropriétés pour la métaclasse "class"	155
Tableau B.2 – Métapropriétés pour la métaclasse "property"	156
Tableau B.3 – Métapropriétés pour la métaclasse "datatype"	158
Tableau B.4 – Métapropriétés pour la métaclasse "enumeration"	159
Tableau B.5 – Métapropriétés pour la métaclasse "term"	160
Tableau B.6 – Métapropriétés pour la métaclasse "relation"	161
Tableau C.1 – Modèles POM et schémas XML	162
Tableau E.1 – Propriétés natives de la classe "Power transformer" ("Transformateur de puissance") dans le CDD de l'IEC	184
Tableau E.2 – Propriétés natives de la classe "Power transformer" ("Transformateur de puissance") dans l'IEC61970-301:2013	186

COMMISSION ÉLECTROTECHNIQUE INTERNATIONALE

**ENREGISTREMENT D'ONTOLOGIE DE PRODUITS
NORMALISÉS ET TRANSFERT PAR TABLEURS –****Partie 3: Interface pour un modèle d'information commun****AVANT-PROPOS**

- 1) La Commission Electrotechnique Internationale (IEC) est une organisation mondiale de normalisation composée de l'ensemble des comités électrotechniques nationaux (Comités nationaux de l'IEC). L'IEC a pour objet de favoriser la coopération internationale pour toutes les questions de normalisation dans les domaines de l'électricité et de l'électronique. A cet effet, l'IEC – entre autres activités – publie des Normes internationales, des Spécifications techniques, des Rapports techniques, des Spécifications accessibles au public (PAS) et des Guides (ci-après dénommés "Publication(s) de l'IEC"). Leur élaboration est confiée à des comités d'études, aux travaux desquels tout Comité national intéressé par le sujet traité peut participer. Les organisations internationales, gouvernementales et non gouvernementales, en liaison avec l'IEC, participent également aux travaux. L'IEC collabore étroitement avec l'Organisation Internationale de Normalisation (ISO), selon des conditions fixées par accord entre les deux organisations.
- 2) Les décisions ou accords officiels de l'IEC concernant les questions techniques représentent, dans la mesure du possible, un accord international sur les sujets étudiés, étant donné que les Comités nationaux de l'IEC intéressés sont représentés dans chaque comité d'études.
- 3) Les Publications de l'IEC se présentent sous la forme de recommandations internationales et sont agréées comme telles par les Comités nationaux de l'IEC. Tous les efforts raisonnables sont entrepris afin que l'IEC s'assure de l'exactitude du contenu technique de ses publications; l'IEC ne peut pas être tenue responsable de l'éventuelle mauvaise utilisation ou interprétation qui en est faite par un quelconque utilisateur final.
- 4) Dans le but d'encourager l'uniformité internationale, les Comités nationaux de l'IEC s'engagent, dans toute la mesure possible, à appliquer de façon transparente les Publications de l'IEC dans leurs publications nationales et régionales. Toutes divergences entre toutes Publications de l'IEC et toutes publications nationales ou régionales correspondantes doivent être indiquées en termes clairs dans ces dernières.
- 5) L'IEC elle-même ne fournit aucune attestation de conformité. Des organismes de certification indépendants fournissent des services d'évaluation de conformité et, dans certains secteurs, accèdent aux marques de conformité de l'IEC. L'IEC n'est responsable d'aucun des services effectués par les organismes de certification indépendants.
- 6) Tous les utilisateurs doivent s'assurer qu'ils sont en possession de la dernière édition de cette publication.
- 7) Aucune responsabilité ne doit être imputée à l'IEC, à ses administrateurs, employés, auxiliaires ou mandataires, y compris ses experts particuliers et les membres de ses comités d'études et des Comités nationaux de l'IEC, pour tout préjudice causé en cas de dommages corporels et matériels, ou de tout autre dommage de quelque nature que ce soit, directe ou indirecte, ou pour supporter les coûts (y compris les frais de justice) et les dépenses découlant de la publication ou de l'utilisation de cette Publication de l'IEC ou de toute autre Publication de l'IEC, ou au crédit qui lui est accordé.
- 8) L'attention est attirée sur les références normatives citées dans cette publication. L'utilisation de publications référencées est obligatoire pour une application correcte de la présente publication.
- 9) L'attention est attirée sur le fait que certains des éléments de la présente Publication de l'IEC peuvent faire l'objet de droits de brevet. L'IEC ne saurait être tenue pour responsable de ne pas avoir identifié de tels droits de brevets et de ne pas avoir signalé leur existence.

La Norme internationale IEC 62656-3 a été établie par le sous-comité 3D, Product properties and classes and their identification (*Propriétés et classes des produits et leur identification* – disponible en anglais seulement), du comité d'études 3 de l'IEC: Structures d'informations, documentation et symboles graphiques.

Le texte de cette norme est issu des documents suivants:

FDIS	Rapport de vote
3D/234/FDIS	3D/245/RVD

Le rapport de vote indiqué dans le tableau ci-dessus donne toute information sur le vote ayant abouti à l'approbation de cette norme.

Cette publication a été rédigée selon les Directives ISO/IEC, Partie 2.

Une liste de toutes les parties de la série IEC 62656, publiées sous le titre général *Enregistrement d'ontologie de produits normalisés et transfert par tableurs*, peut être consultée sur le site web de l'IEC.

Le comité a décidé que le contenu de cette publication ne sera pas modifié avant la date de stabilité indiquée sur le site web de l'IEC sous "<http://webstore.iec.ch>" dans les données relatives à la publication recherchée. A cette date, la publication sera

- reconduite,
- supprimée,
- remplacée par une édition révisée, ou
- amendée.

IMPORTANT – Le logo 'colour inside' qui se trouve sur la page de couverture de cette publication indique qu'elle contient des couleurs qui sont considérées comme utiles à une bonne compréhension de son contenu. Les utilisateurs devraient, par conséquent, imprimer cette publication en utilisant une imprimante couleur.

IECNORM.COM : Click to view the full PDF of IEC 62656-3:2015

INTRODUCTION

La série IEC 62656, *Enregistrement d'ontologie de produits normalisés et transfert par tableurs* est une série de normes internationales qui définissent collectivement les méthodes de transfert et d'enregistrement des ontologies de divers produits et services depuis et vers des enregistrements et des applications d'ontologies basés sur le modèle commun de dictionnaire de données de la norme IEC 61360-ISO 13584. Le dictionnaire de données communes de l'IEC (IEC common data dictionary), ou le CDD de l'IEC en abrégé, est l'un de ces ensembles d'enregistrements maintenus en ligne comme Norme internationale IEC 61360-4 basée sur la procédure des bases de données IEC stipulée dans les "Directives ISO/IEC – Procédures spécifiques à l'IEC". Le CDD de l'IEC est un dictionnaire de données interdomaine couvrant tous les produits et services électrotechniques, maintenu et mis à jour par le SC3D de l'IEC, dont l'infrastructure de base de données est administrée par le Bureau central de l'IEC.

Le modèle d'information commun défini dans la série de normes IEC 61968/61970, souvent appelé par son nom abrégé "CIM", fournit une façon normalisée de représenter tous les objets principaux d'une entreprise de distribution d'électricité, généralement nécessaires pour modéliser les aspects opérationnels d'une entreprise d'électricité. Ce modèle comprend les classes et les attributs publics pour ces objets, ainsi que les relations entre eux. Il est connu comme modèle d'information pour système de gestion d'énergie (EMS – Energy Management System) des réseaux électriques et il est actuellement reconnu comme modèle d'ontologie normalisé pour les réseaux intelligents ("Smart Grids") développés et maintenus par le comité d'études 57 de l'IEC. Une spécification d'ontologies conforme au modèle de données CIM est disponible au format UML selon l'IEC 61970-301 et au format RDF selon l'IEC 61970-501.

La série IEC 62656 est constituée des parties suivantes, publiées sous le titre général Enregistrement d'ontologie de produits normalisés et transfert par tableurs:

- Partie 1: Structure logique pour les paquets de données;
- Partie 2: Guide d'application pour l'utilisation avec le Dictionnaire de données communes (CDD) de l'IEC;
- Partie 3: Interface pour un modèle d'information commun.

ENREGISTREMENT D'ONTOLOGIE DE PRODUITS NORMALISÉS ET TRANSFERT PAR TABLEURS –

Partie 3: Interface pour un modèle d'information commun

1 Domaine d'application

La présente partie de l'IEC 62656 spécifie une interface entre la série IEC 62656 et le métamodèle pour CIM défini à l'origine dans l'IEC 61968 et dans la série de normes IEC 61970. Le CIM actuel comprend la série IEC 62325 et l'interface spécifiée dans cette partie de l'IEC 62656 s'applique également au modèle défini dans l'IEC 62325-301:2004. Plus précisément, la présente Norme définit une mise en correspondance formelle entre l'IEC 62656 et le métamodèle pour CIM afin d'importer l'ontologie CIM dans le CDD de l'IEC et assurer l'interopérabilité des ontologies des deux normes, ou même parmi un plus large spectre de normes. Pour la base de la mise en correspondance du CIM vers le modèle de données défini dans l'IEC 62656-1, il est fait référence à la représentation UML du CIM.

La spécification d'interface fournie par la présente partie de l'IEC 62656 rend l'ontologie des réseaux intelligents du CIM accessible et interopérable parmi les autres éléments d'ontologie provenant d'autres normes d'ontologies, intégrant les domaines d'informations matériel, environnemental, mécanique et logistique. En outre, la spécification peut également couvrir le cycle de vie de produits.

La présente partie de l'IEC 62656 définit également des méthodes pour transformer le contenu du CDD de l'IEC au format défini dans l'IEC 61968 et la série IEC 61970. Cependant, la présente Norme ne vise aucune normalisation des éléments définis dans le CDD de l'IEC comme partie intégrante de l'ontologie CIM, car toutes les définitions du CDD de l'IEC font déjà partie d'une Norme internationale horizontale reconnue, à savoir l'IEC 61360-4-DB. Elle rend plutôt les éléments d'informations stockés dans le CDD de l'IEC disponibles pour une normalisation ou une adaptation ultérieure pour les domaines de l'énergie électrique ou pour des communautés d'utilisateurs plus spécifiques en tant que ressource de base, en particulier dans le but d'étendre la série de normes CIM.

Le modèle de données de la présente partie est basé sur l'IEC 62656-1 qui spécifie la structure logique pour un paquet de données, ou simplement "Paquet," qui est le nom abrégé pour le support d'enregistrement ou de transfert d'ontologie de produits. Un tel paquet peut être typiquement mis en œuvre dans une feuille de tableur. Le modèle de données bâti sur un ensemble de paquets est appelé "Parcellized Ontology Model" (Modèle d'ontologie Paquetée), qui est souvent abrégé "POM" dans le reste du présent document.

Cette partie de la norme IEC 62656 spécifie:

- les règles et principes de mise en correspondance pour importer des objets CIM UML vers le POM;
- les exigences syntaxiques et sémantiques relatives aux objets du CIM paqueté;
- le mécanisme de contrôle de version par parties dans le POM pour les objets du CIM paqueté.

Les éléments suivants ne font pas partie du domaine d'application de la présente partie de l'IEC 62656:

- définition de l'interface entre le CIM-UML et le CIM RDF;
- spécification des exigences syntaxiques et sémantiques relatives au CIM UML;
- spécification des exigences syntaxiques et sémantiques relatives au CIM RDF.

2 Références normatives

Les documents suivants sont cités en référence de manière normative, en intégralité ou en partie, dans le présent document et sont indispensables pour son application. Pour les références datées, seule l'édition citée s'applique. Pour les références non datées, la dernière édition du document de référence s'applique (y compris les éventuels amendements).

IEC 62656-1:2014, *Enregistrement d'ontologie de produits normalisés et transfert par tableurs – Partie 1: Structure logique pour les paquets de données*

IEC 61360-1:2009, *Types normalisés d'éléments de données avec plan de classification pour composants électriques – Partie 1: Définitions – Principes et méthodes*

IEC 61360-4, *Types normalisés d'éléments de données avec plan de classification pour composants électriques – Partie 4: Collection de référence CEI des types normalisés d'éléments de données, des classes de composants et des termes, disponible à <<http://std.iec.ch/iec61360>>*

IEC 61968-1:2003, *System interfaces for distribution management – Part 1: Interface architecture and general requirements* (disponible en anglais seulement)

IEC 61968-11:2010, *Intégration d'applications pour les services électriques – Interfaces système pour la gestion de la distribution – Partie 11: Extensions du modèle d'information commun (CIM) pour la distribution*

IEC 61970-1:2005, *Interface de programmation d'application pour système de gestion d'énergie (EMS-API) – Partie 1: Lignes directrices et exigences générales*

IEC 61970-2:2004, *Interface de programmation d'application pour système de gestion d'énergie (EMS-API) – Partie 2: Glossaire*

IEC 61970-301:2013, *Interface de programmation d'application pour système de gestion d'énergie (EMS-API) – Partie 301: Base de modèle d'information commun (CIM)*

IEC 61970-501:2006, *Système de gestion d'énergie – Interface de programmation d'application (EMS-API) – Partie 501: Schéma de cadre général pour la description des ressources du modèle d'information commun (CIM RDF)*

IEC 62325-301:2014, *Cadre pour les communications pour le marché de l'énergie – Partie 301: Extensions du modèle d'information commun (CIM) pour les marchés*

ISO 639-1:2002, *Codes pour la représentation des noms de langue – Partie 1: Code alpha-2*

ISO 8601:2004, *Éléments de données et formats d'échange – Échange d'information -- Représentation de la date et de l'heure*

ISO/IEC Directives Supplement:2013, *Procedure Specified to IEC* (disponible en anglais seulement)

3 Termes, définitions et abréviations

3.1 Termes et définitions

Pour les besoins du présent document, les termes et définitions de l'IEC 61970-2:2004 et de l'IEC 62656-1, ainsi que les suivants, s'appliquent.

3.1.1

attribut

- a) attribut défini dans l'IEC 62656-1, équivalent d'une métapropriété
- b) attribut de classe utilisé dans l'IEC 61970-301 correspondant à une propriété dans le POM
- c) attribut d'énumération utilisé dans l'IEC 61970-301 correspondant à un terme dans le POM

Note 1 à l'article: dans le cas de b), il est plus spécifiquement appelé "attribut du CIM" ou "attribut de classe du CIM" dans la présente norme.

3.1.2

aggregationKind

genre d'association UML comprenant l'association normale, l'agrégation et la composition

Note 1 à l'article: la valeur d'aggregationKind peut être "none" ("aucune") pour une association normale, "shared" ("partagée") pour une agrégation et "composition" ("composition") pour une composition, comme dans les définitions de l'ISO/IEC 19505-2:2012.

3.1.3

multiplicité CIM à la source

multiplicité des instances de la classe située du côté source de deux classes reliées par une association

3.1.4

data type (type de données) de base du CIM

data type (type de données) de base défini dans le CIM comprenant les types "String" (chaîne), "Boolean" (booléen), "Float" (flottant) et "Integer" (entier)

Note 1 à l'article: dans l'édition du CIM, il existe d'autres types de données de base utilisables.

3.1.5

objet CIM

classe, attribut, paquetage ou relation défini(e) dans les séries de normes IEC 61970, IEC 61968 et la série IEC 62325 pour représenter des ressources de système d'énergie

3.1.6

paquetage du CIM

moyen polyvalent permettant de grouper les éléments de modèle liés du CIM

Note 1 à l'article: "Paquetage du CIM" est un synonyme donné à l'attribut Segment(MDC_P211) de l'IEC 62656-1:2014.

3.1.7

CIM UML ID

ID de chaque objet CIM, assigné par un outil UML par lequel l'objet est conçu

3.1.8

INVERSIBLE_P_DET

INVERSIBLE_PROPERTY

genre de type d'élément de données de propriété qui peut dépendre du même type de propriété

Note 1 à l'article: une association UML avec une navigabilité non spécifiée est définie avec INVERSIBLE_P_DET car la navigabilité peut être assignée ultérieurement.

3.1.9

navigabilité

état d'accessibilité des instances d'une classe à une extrémité d'une association à partir des instances de la classe à l'autre extrémité de l'association

Note 1 à l'article: Dans la présente norme, on distingue la navigabilité bidirectionnelle, unidirectionnelle et non spécifiée.

3.1.10

CIM paqueté

collection d'objets CIM traduits et représentés par un ensemble de tableurs conformes aux objets du POM définis dans l'IEC 62656-1, avec une liste étendue d'attributs ajoutés dans la présente partie de la Norme

3.1.11

objet du POM

ensemble d'objets définis dans une méta-classe de l'IEC 62656-1:2014 et utilisés dans la présente partie de l'IEC 62656.

EXAMPLE classe, propriété, datatype, énumération, unité, terme et relation

3.1.12

propriété

- a) instance de la métaclasse Property ou la métaclasse en soi comme schéma; définie dans l'IEC 62656-1
- b) propriété définie dans l'IEC 61970-2 qui représente un aspect spécifique, une caractéristique, un attribut ou une relation utilisé(e) pour décrire une ressource réseau

Note 1 à l'article: Afin d'établir une distinction claire entre ces deux cas, dans le cas a), on l'appelle "propriété du POM" ou "propriété dans le POM". Dans le cas b), on l'appelle "propriété CIM" dans la présente partie de l'IEC 62656; chaque fois que cela est nécessaire.

3.2 Abréviations

Pour les besoins du présent document, les abréviations suivantes s'appliquent.

Abréviation	Anglais	Français
CIM	common information model	modèle d'information commun
POM	parcellized ontology model	modèle d'ontologie paquetée
IEC CDD	IEC common data dictionary	dictionnaire de données communes de l'IEC
UML	unified modeling language	langage de modélisation unifié
RDF	resource description framework	structure de description de ressources
RDFS	resource description framework schema	schéma de structure de description de ressources
XML	extensible markup language	langage de balisage extensible

4 Règles et principes génériques pour la conception d'interface CIM

4.1 Analyse comparative des structures et fonctionnalités du POM et du CIM

Le CIM actuel est modélisé avec l'UML qui est largement accepté par les ingénieurs TI pour concevoir les concepts des produits et des services. L'UML est donc utilisé comme un outil de conception pratique pour modéliser le CIM. Cependant, du point de vue de l'utilisateur, le CIM actuel modélisé en UML est totalement dépourvu de tout mécanisme pour mettre à jour ou étendre localement les entités du modèle l'une après l'autre, alors que le POM est judicieusement équipé pour enregistrer en continu l'historique des changements. Autrement dit, la norme CIM actuelle est conçue pour mettre à jour l'ensemble du fichier du modèle. Cependant, cela constitue un obstacle significatif pour la maintenance du modèle comme une ontologie dans une base de données et pour les mises à jour individuelles des éléments de la base de données. Étant donné que chaque mise à jour doit entraîner un changement holistique du modèle, ce changement perturbera le fonctionnement des objets déjà installés et utilisés dans diverses applications et divers systèmes.

Afin de répondre à cette préoccupation, une structure évoluée pour la maintenance de l'ontologie et la gestion des modifications des éléments de l'ontologie de la base de données est requise et doit être dotée

- a) d'une gestion de l'évolution des ontologies
- b) d'un mécanisme d'extension des ontologies
- c) d'une gestion hiérarchique tant pour la spécialisation ("is-a") que pour la composition ("has-a").

Les trois exigences susmentionnées sont complètement respectées par le POM, alors que le CIM en UML ou en RDF est conçu pour d'autres exigences et d'autres objectifs.

En outre, par son domaine d'application, le CIM se concentre sur l'échange d'informations entre les systèmes et les produits liés à l'énergie électrique. De ce fait, les informations couvertes par l'IEC 61968, l'IEC 61970 et la série IEC 62325 ne sont pas exhaustives pour décrire toute la gamme et tout le cycle de vie des produits et services utilisés dans les réseaux électriques: à savoir la fabrication, l'installation, la maintenance, le transport et les caractéristiques relatives à la mise au rebut des produits et services ne sont pas couverts. Certes, certaines parties de l'IEC 61968 et de la série IEC 62325 sont destinées à associer des informations de différents domaines, mais elles ne normalisent ni ne peuvent normaliser les domaines entiers de l'ontologie électrotechnique, tout simplement parce que cela ne relève pas du domaine d'application d'un TC/SC.

Cela exige presque la totalité des informations de tous comités d'études de l'IEC: c'est pourquoi la description générique des classes, des propriétés et de l'identification des produits couvrant tous les domaines techniques est assignée à un comité horizontal. La gestion du CDD de l'IEC 61360 étant assignée à un comité horizontal, l'intégration et la maintenance de toutes les gammes d'ontologies électrotechniques sont possibles.

Par conséquent, la présente partie de l'IEC 62656 fournit les moyens et les méthodes pour importer dans le CDD de l'IEC l'ontologie définie dans le CIM et permet ainsi l'utilisation interopérable des ontologies tant du CDD de l'IEC que du CIM d'une manière harmonieuse, et ce, au profit de tous les systèmes et de toutes les applications électrotechniques. Le cas de la classe "Transformer" ("Transformateur") est indiqué à l'Annexe E comme un exemple typique permettant de montrer de quelle façon les définitions du CIM et du CDD de l'IEC sont complémentaires.

Avec cette perspective à l'esprit, la présente partie de l'IEC 62656 définit en premier lieu les règles et les principes de la mise en correspondance entre les métamodèles du CIM et du POM dans le but d'importer les objets du premier dans le CDD de l'IEC, sous la forme du second. Elle définit également les attributs supplémentaires au modèle CIM actuel qui sont indispensables pour gérer le contenu de la base de données du CDD de l'IEC. En termes de choix de représentation des données sources, la présente Norme met directement en correspondance les objets du CIM UML dans le POM, car le CIM est à l'origine défini en UML. En conséquence, les profils CIM, qui sont des sous-ensembles du modèle CIM, peuvent également être transformés en POM parce qu'il faut qu'ils soient compatibles avec les définitions du CIM UML.

En second lieu, la présente Norme vise à spécifier les règles et le principe pour transformer les objets conformes à l'IEC 61360 représentés dans l'IEC 62656-1 en un format conforme à la norme CIM, à savoir, le format CIM RDFS (Ressource Description Framework Schema). Cela signifie virtuellement que les objets dans le CDD de l'IEC 61360 sont transformables en éléments d'une ontologie qui peut être considérée comme étant une extension possible du CIM.

Le choix du format, le RDFS, est dû à des raisons pratiques: La présente Partie de l'IEC 62656 ne vise pas à automatiser totalement la production d'objets CIM à partir du CDD de l'IEC. Un convertisseur basé sur la présente Norme peut créer des objets semblables à ceux du CIM en RDFS, et les informations contenues dans les objets doivent être utiles pour

la mise à jour ou l'extension de l'ontologie CIM de la prochaine génération. Cependant, la sélection et la validation du point de vue tant syntaxique que sémantique nécessitent d'être accomplies par des experts du domaine de l'énergie électrique. De plus, la normalisation technosémantique dans cette perspective ne s'inscrit pas dans le domaine d'application de cette partie de l'IEC 62656. Il s'agit d'un sous-comité horizontal chargé du développement et de la maintenance du modèle et du contenu du CDD de l'IEC à travers les domaines de l'IEC, mais il n'a pas nécessairement l'expertise de chaque domaine technique spécifique.

Cependant, les étapes de l'intégration des objets CIM dans le CDD de l'IEC sont sensiblement facilitées par les méthodes décrites dans la présente Norme. En effet, la présente Norme décrit de façon exhaustive les règles pour la transformation syntaxique des objets CIM en des formes compatibles avec le CDD de l'IEC par le biais de la projection des anciennes constructions sur les structures de données de paquets. La présente Norme fournit en outre un ensemble de règles et de méthodes de base qui sont utiles pour la validation sémantique du contenu du CIM paqueté (Parcelized CIM) dans le CDD de l'IEC, dans le but d'assurer l'interopérabilité des ontologies provenant de comités de normalisation différents.

Alors que la validation sémantique détaillée doit rester dans les mains d'experts humains, les règles de la Gestion des changements du contenu enregistré dans le CDD de l'IEC en combinaison avec le manuel guide de la qualité donnent des conseils pratiques et une aide à cette fin.

A l'article 5, les règles et les principes de la mise en correspondance entre le CIM et le POM sont définis comme un contenu normatif. La description détaillée des mises en correspondances est résumée dans l'Annexe A. La transformation inverse des objets du CDD de l'IEC en CIM RDF et sa relation avec le CIM-RDF d'origine sont énumérées dans l'Annexe C à titre informatif.

4.2 Série de normes CIM

Le modèle d'information commun (CIM) spécifie la sémantique de l'Interface de Programmation d'Application (API) pour les systèmes de gestion d'énergie (EMS). Le CIM étant un modèle abstrait qui représente la sémantique de tous les objets types d'une entreprise de distribution d'électricité, il comporte plusieurs parties de norme, chacune traitant d'un domaine et d'un thème spécifiques relatifs à l'ontologie des réseaux électriques. L'IEC 61970-301 est la norme fondamentale et elle définit un ensemble de base du CIM, sous forme d'un ensemble de paquetages, qui donne une vue logique des aspects fonctionnels de l'EMS, y compris les SCADA. L'IEC 61970-302 est une autre partie de la série et elle définit les aspects de programmation financiers et énergétiques. L'IEC 61968 est une autre série du CIM qui traite d'autres vues logiques, telles que l'exploitation des entreprises de distribution, y compris les ressources, l'emplacement, les activités, la documentation et la gestion des tâches. De plus, l'IEC 62325-301 est une partie de norme qui définit les extensions du CIM pour les communications pour les marchés de l'énergie. Il existe un seul modèle unifié d'information qui comporte toutes les parties du CIM, même si elles ont des documents normatifs individuels différents.

La méthodologie et le modèle définis dans l'IEC 61970-301:2013 a été choisie comme la base logique pour la présente partie de la norme.

5 Règles de transformation du CIM vers le POM

5.1 Règles générales

Comme l'indique le Tableau 1, le métamodèle UML disponible dans le CIM doit être projeté sur le métamodèle POM. En tout premier lieu, chaque classe dans le CIM doit être mise en correspondance avec une classe dans le POM. Chaque attribut d'une classe du CIM doit être transformé en une propriété correspondante dans le POM. Une énumération dans le CIM doit être transformée en une énumération dans le POM et les attributs d'une énumération dans le CIM doivent être transformés en des termes correspondants dans le POM. Une association

entre classes dans le CIM, y compris l'agrégation, doit être également transformée en une propriété dans le POM. Chaque data type (type de données) dans le CIM doit être mis en correspondance avec une instance de la métaclasse datatype; il est donc traité comme un type de données défini par un utilisateur qui augmente les types de données intégrés du POM. Un paquetage du CIM doit aussi être transformé en une relation dans le POM.

L'Annexe A énumère en détail les règles de transformation des éléments du Tableau 1 sous la forme d'un ensemble de relations dans le POM: noter que le POM a une capacité intégrée permettant de décrire diverses règles de mise en correspondance entre ontologies comme des relations. Cependant, il est important d'avoir à l'esprit la distinction entre le tableau de mise en correspondance et le modèle CIM paqueté, qui est le résultat produit par les règles de transformation décrites dans le Tableau 1.

Tableau 1 – Mise en correspondance des objets CIM et des objets du POM

Métamodèle UML			Métamodèle POM
Métaobjet	(1)	Classe	Classe
	(2)	Classe Enumeration	Énumération
	(3)	Classe Datatype	Datatype
	(4)	Attributs de classe primaire	Propriété
	(5)	Attributs d'énumération	Terme
	(6)	Paquetage	Relation

NOTE "Classe primaire" fait référence à une classe définie dans l'UML, à l'exception des classes pour Enumeration (énumération) et Datatype (type de données).

Les détails des règles de transformation pour chaque objet sont expliqués ci-après dans les paragraphes 5.2 à 5.8.

5.2 Classe du CIM

5.2.1 Généralités

Du point de vue d'une modélisation de données, une classe en UML est une abstraction d'une structure d'informations qui représente un composant statique d'un système d'informations. Les composants sont représentés sous forme de classes avec leurs attributs explicites, et les relations parmi les classes sont indiquées sous forme d'associations. Une classe peut avoir une relation hiérarchique avec une classe généralisée ou une classe spécialisée, appelée respectivement "classe parent" ou "classe enfant", dans laquelle l'héritage transmet les attributs de la classe parent à sa descendance. En bref, une classe en UML est plus un élément d'outil de conception logicielle générique ou de méthodologie générique pour la programmation orientée objet qu'une structure pour représenter un concept de produits ou de services du monde réel. De ce fait, elle est adaptée à guider de manière générale la conception logicielle globale. Toutefois, elle n'est pas nécessairement la meilleure pour maintenir et mettre à jour les spécifications des produits et des services comme une seule entité d'informations appelée "ontologie" qui est séparée du corps de la description procédurale du logiciel, car le volume de son contenu augmente, et la spécification des éléments évolue individuellement au fil du temps.

Au contraire, une classe dans le POM est une abstraction d'un concept de produits avec une collection de propriétés qui sont chacune conçues comme une abstraction de la "propriété de produit" qui a en général une unité de mesure et un type de données correspondant, lorsque le type est quantifiable. De plus, une classe est générée comme une instance d'une

métaclasses de classes, c'est-à-dire, une "usine" de classes ou parfois, avec un nom raccourci, comme "paquet de classes" et elle est conçue pour être définie et mise à jour au sein d'une base de données. Par conséquent, une classe est censée avoir un nom préférentiel, des noms synonymes, des définitions, des traductions en plusieurs langues, des identificateurs (ID) uniques au niveau global, des versions et des révisions, un ensemble de caractéristiques, etc., en tant que partie intégrante d'une structure par défaut et prédéfinie pour représenter un concept de produits ou de services. Il va sans dire que l'héritage et les relations "toutes les parties" sont intégrés comme partie intégrante du modèle. En bref, une classe dans le POM est conçue plus précisément comme une partie intégrante d'une structure de base de données pour représenter un concept de produits et de services dans le monde réel.

Ceci étant accepté, les classes du CIM peuvent être transformées en classes du POM, comme le spécifie un ensemble de règles de mise en correspondance qui sont résumées dans le Tableau 2.

Dans le Tableau, la colonne "métamodèle UML dans le CIM" représente le métamodèle UML d'origine appliqué dans le CIM. Avec deux sous-colonnes: "Objet UML" qui représente le nom de l'objet UML, tandis que "Définition du CIM" représente la définition CIM correspondante de l'objet UML placé à gauche.

La colonne "métamodèle POM" représente le métamodèle du POM. Avec trois sous-colonnes: "Objet du POM" qui représente le nom de l'objet du POM correspondant au concept du CIM UML, "Nom de l'attribut" qui représente le nom de l'attribut du POM et "ID d'attribut" qui représente l'ID défini dans le POM pour le nom d'attribut du POM placé à gauche.

La colonne "Description" sert à préciser une instruction interprétable par l'homme donnant des lignes directrices pour transformer le métamodèle UML dans le CIM en métamodèle correspondant du POM.

Tableau 2 – Correspondance entre une classe du CIM en UML et une classe du POM

Métamodèle UML dans le CIM		Métamodèle POM			Description
Objet UML	Définition du CIM	Objet du POM	Nom d'attribut	ID d'attribut	
Classe	Name ("Nom")	Classe	Short name ("Nom court")	MDC_P004_3.<lang>	<lang> dans le POM doit être un code ISO 639 de deux lettres. Le CIM ne donnant que des noms en anglais, lang doit être "en" ici.
	Description		Note	MDC_P007_1.<lang>	<lang> dans le POM doit être un code ISO 639 de deux lettres. Le CIM ne donnant que des notes en anglais, lang doit être "en" ici.
	Généralisation (inverse de Spécialisation)		Superclass ("Superclasse")	MDC_P010	

5.2.2 Name ("Nom")

Le nom d'une classe du CIM doit être transformé en l'attribut "short name" (MDC_P004_3.<lang>) de la classe du POM. En même temps, sachant que le POM requiert un "Preferred name" (MDC_P004_1.<lang>) pour chaque classe du POM comme un élément obligatoire, le nom CIM d'origine doit être copié vers "Preferred name" de la même classe comme nom provisoire initial, la première fois qu'il y a transformation du CIM vers le POM. Le "Preferred name" d'une classe dans le POM peut être mis à jour selon la procédure de

maintenance pour les normes de bases de données spécifiées dans le supplément aux Directives ISO/IEC.

5.2.3 Description

L'attribut "Description" d'une classe du CIM sert à expliquer ou décrire une classe. Par conséquent, la "Description" d'une classe du CIM doit être mise en correspondance avec "Note"(MDC_P007_1.<lang>) d'une classe du POM. En même temps, sachant que le POM requiert une "Definition"(MDC_P005.<lang>) pour chaque classe du POM comme un élément obligatoire, la note du CIM d'origine doit être copiée vers "Definition" de la même classe du POM comme définition provisoire initiale, la première fois qu'il y a transformation du CIM vers le POM. La "Definition" dans le POM est mise à jour selon la procédure de maintenance pour les normes de bases de données spécifiée dans le supplément aux Directives ISO/IEC.

5.2.4 Généralisation-Spécialisation

L'UML a la capacité de décrire la relation généralisation-spécialisation entre deux classes. Dans le cas de l'exemple indiqué à la Figure 1, la classe "PowerSystemResource" est une généralisation de la classe "Equipment". La spécialisation est l'inverse de la généralisation. La généralisation est également connue comme étant une relation "is-a" qui peut être décrite par un attribut "Superclass" ("superclasse") d'une classe du POM. Dans cet exemple, la classe "Equipment" a une superclasse nommée "PowerSystemResource" dans le POM.

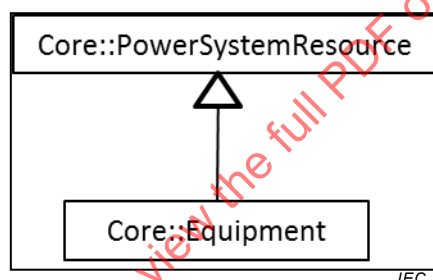


Figure 1 – Exemple de généralisation du CIM
(provenant de la Figure 2 de l'IEC 61970-301:2013)

5.3 Attributs de classe du CIM

5.3.1 Généralités

Les attributs d'une classe UML sont les éléments descriptifs d'une classe, alors qu'une propriété du POM décrit un aspect caractéristique d'une classe. De ce fait, les attributs de classe UML peuvent être mis en correspondance avec les propriétés du POM avec la spécification requise du type de données, etc.

Tableau 3 – Mise en correspondance d'un attribut de classe du CIM et d'une propriété du POM

Métamodèle UML dans le CIM		Métamodèle POM			Description
Objet UML	Définition du CIM	Objet du POM	Nom d'attribut	ID d'attribut	
Attribut de classe	Name ("Nom")	Propriété	Short name ("nom court")	MDC_P004_3.<lang>	<lang> dans le POM doit être un code ISO 639 de deux lettres. Le CIM ne donnant que des noms en anglais, lang doit être "en" ici.
	(pour attribut natif)		Définition class ("classe de Définition")	MDC_P021	pour l'attribut natif, sa classe de définition doit être décrite ici
	Type		Data type (Type de données)	MDC_P022	
	Description		Note	MDC_P007_1.<lang>	<lang> dans le POM doit être un code ISO 639 de deux lettres. Le CIM ne donnant que des notes en anglais, lang doit être "en" ici.

5.3.2 Name ("Nom")

Le nom d'un attribut de classe du CIM doit être transformé en l'attribut "Short name" (MDC_P004_3.<lang>) de la propriété du POM. En même temps, le nom d'origine de l'attribut de classe du CIM doit être copié vers "Preferred name" de la même propriété du POM comme nom provisoire initial, la première fois qu'il y a transformation du CIM vers le POM. Le "Preferred name" d'une propriété dans le POM peut être mis à jour selon la procédure de maintenance pour les normes de bases de données spécifiées dans le supplément aux Directives ISO/IEC.

5.3.3 Attribut natif

Dans le CIM, il y a deux types pour les attributs de classe: un type est un ensemble d'attributs natifs, qui sont définis dans la classe elle-même. L'autre type est l'ensemble d'attributs hérités, qui sont hérités d'une super classe ou de classes ancêtres.

Le POM fournit la structure ci-après pour établir la différence entre les attributs natifs et les attributs hérités:

- Chaque attribut natif doit être transformé en une propriété du POM qui a sa "Definition class" ("classe de définition") dans la classe où la propriété est définie. En même temps, on doit ajouter la propriété à l'attribut des "Applicable properties" ("propriétés applicables") (expliquée en 6.2.15.) de la classe du POM.
- Les attributs hérités d'une classe doivent provenir des "Applicable properties" ("propriétés applicables") de ses classes supérieures.

5.3.4 Type

Le type d'un attribut du CIM doit être transformé en datatype d'une propriété du POM qui définit le type de données possible des valeurs de l'instance.

5.3.5 Description

La "Description" de l'attribut de classe du CIM doit être transformée en l'attribut "Note" d'une propriété du POM. La note doit être copiée vers l'attribut "Definition" de la même propriété du POM lorsque la conversion du CIM vers le POM est accomplie pour la première fois, comme

une définition provisoire. La "Note" dans le POM n'est pas modifiée pour conserver l'originalité du CIM, et la "Définition" dans le POM est mise à jour conformément à la procédure de maintenance pour les normes de bases de données spécifiées dans le supplément aux Directives ISO/IEC.

5.4 Association du CIM

5.4.1 Généralités

Une association est une connexion conceptuelle entre les classes. Dans l'UML, une association peut avoir un "name" ("nom") et avoir plus de deux membres "association ends" ("extrémités d'association"). Pour chaque extrémité d'association, les attributs de "aggregationKind", "navigability" et "multiplicity" sont définis. Une agrégation en UML est un cas spécialisé d'une association. Comparée à l'association normale, elle décrit une relation "part-whole" (toutes les parties) entre des classes. De plus, il existe un type spécial d'agrégation défini comme "composition" qui impose que l'objet composé doit être responsable de l'existence et du stockage des objets composés, c'est-à-dire il convient que les objets aient le même cycle de vie selon la composition. Comme l'agrégation et la composition sont des types d'association spécifiés, leurs trois sortes doivent être décrites avec le même modèle dans le POM. Dans le présent paragraphe 5.4, il est expliqué comment représenter une association, y compris l'agrégation et la composition dans le POM.

Dans le CIM, un nom d'association n'est pas spécifié normalement. Chaque extrémité d'association contient les informations, y compris la multiplicité indiquée dans le Tableau 4. Les attributs de "aggregationKind" et "navigability" ne sont pas directement expliqués dans la structure d'extrémité d'association dans le CIM, mais on peut y accéder à partir du diagramme de classe UML ou du fichier XMI (XML Metadata Interchange) correspondant pour le CIM.

Dans la présente partie de la Norme, chaque extrémité d'association avec son aggregationKind, sa navigabilité et sa multiplicité est mise en correspondance avec une propriété du POM. Les détails des règles de mise en correspondance sont expliqués au paragraphe 5.4.3.

Bien que la composition n'a pas encore été appliquée dans les modèles CIM, la méthode de représentation de la composition dans le POM est introduite dans la présente norme pour une utilisation possible de la construction dans une future édition du CIM.

5.4.2 Structure d'extrémité d'association du CIM

Chaque extrémité d'association contient les informations indiquées dans le Tableau 4, dont la rédaction est adaptée à partir de l'originale définie dans l'IEC 61970-301:2013.

Comme décrit dans la norme CIM, la première colonne indique la multiplicité des instances de la classe (par une seule instance de la classe reliée à l'extrémité nommée) à partir de laquelle l'association ayant le nom d'extrémité est étendue (c'est-à-dire la multiplicité de la classe pour laquelle les extrémités d'association sont décrites). Dans la deuxième colonne, la multiplicité de l'extrémité d'association, à savoir la plage des instances de la classe reliée à l'extrémité d'association avec un nom d'extrémité d'association spécifique en texte clair, est placée entre crochets à l'avant du nom. Le nom de la classe reliée à l'extrémité d'association est indiqué dans la colonne "type". Une multiplicité zéro indique une association facultative. Une multiplicité égale à "*" indique que n'importe quel nombre est autorisé. Par exemple, une multiplicité [1..*] indique qu'une plage d'instances de 1 à n'importe quel nombre est autorisée.

Tableau 4 – Structure d'extrémité d'association du CIM

Multi à partir de	[Mult jusqu'à] nom	Type	description
[0..*]	[1..1]BaseVoltage	BaseVoltage	La tension de base utilisée pour tous les équipements au sein du VoltageLevel

5.4.3 Règle de transformation générale pour association

Dans l'UML, il convient que chaque association n'ait pas moins de deux extrémités d'association. Chaque extrémité d'association est définie comme un membre de l'association. Comme seule une association binaire comprenant deux extrémités d'association est disponible dans le CIM actuel, dans la présente norme il est expliqué comment représenter une association binaire.

Dans la présente norme, chaque extrémité d'association est transformée en une instance d'une propriété du POM et chaque association est transformée en une instance de la relation du POM. Les extrémités d'association possédées par une association sont définies dans le "Domain of the relation" ("Domaine de la relation") issu de l'association UML.

Pour chaque association expliquée dans la relation du POM, les valeurs des attributs suivants doivent être assignées de la façon suivante:

- Relation type(MDC_P200) (type de relation) a une valeur de "Predication"
- Role of the relation(MDC_P210) (rôle de la relation) a une valeur de "UML Association" ("Association UML")
- Domain of the relation(MDC_P201) (domaine de la relation) a une valeur d'une liste d'ID de propriété représentant l'extrémité d'association possédée par cette association.

Pour une association sans nom donné, le code assigné à la relation correspondante peut être copié vers l'attribut "Preferred name" (MDC_P004_1) (nom préférentiel).

Le Tableau 5 indique des échantillons de la façon de décrire une association dans la relation du POM.

Dans le Tableau 5, "Asso1" représente l'association indiquée à la Figure 2 avec sa définition des extrémités d'association du Tableau 7 dans les deux premiers enregistrements. "Asso2" représente l'association indiquée à la Figure 3 avec sa définition des extrémités d'association du Tableau 7 dans le troisième et le quatrième enregistrements.

Tableau 5 – Associations du CIM définies comme relations du POM

MDC_P001_13	MDC_P004_1.<lang>	MDC_P007_1.<lang>	MDC_P200	MDC_P201	MDC_P202	MDC_P203	MDC_P208	MDC_P210
Code	Preferred name (nom préférentiel)	Note	Relation type (type de relation)	Domain of the relation (domaine de la relation)	Domain of the function (domaine de la fonction)	Codomain of the function (co-domaine de la fonction)	Domain element type (type d'élément du domaine)	Role of the relation (rôle de la relation)
Asso1			Predication	{Property_code1,Property_code2}				Association UML
Asso2			Predication	{Property_code3,Property_code4}				Association UML

5.4.4 Règle de transformation générale pour extrémité d'association

5.4.4.1 Généralités

L'extrémité d'association du CIM doit être mise en correspondance avec une propriété du POM. Dans cette propriété, il convient que l'attribut des "applicable relations" ("relations applicables") indique l'ID de la relation qui représente l'association contenant cette extrémité d'association. La navigabilité d'une extrémité d'association est distinguée avec les attributs "property data element type" ("type d'élément de données de propriété") et "condition" de la propriété. L'aggregationKind est distingué de l'attribut "remark" ("remarque") de la propriété. Le Tableau 6 résume la représentation de l'extrémité d'association du CIM dans le POM.

IECNORM.COM : Click to view the full PDF of IEC 62656-3:2015

Tableau 6 – Mise en correspondance d'une extrémité d'association du CIM et d'une propriété du POM

Métamodèle UML dans le CIM		Métamodèle POM			Description
Objet UML	Définition du CIM	Objet du POM	Nom d'attribut	ID d'attribut	
Association end ("Extrémité d'association")	Association end name ("Nom d'extrémité d'association")	Property ("Propriété")	Short name ("Nom court")	MDC_P004_3. <lang>	<lang> dans le POM doit être un code ISO 639 de deux lettres. Le CIM ne donnant que des noms en anglais, lang doit être "en" ici.
	Description		Note	MDC_P007_1. <lang>	<lang> dans le POM doit être un code ISO 639 de deux lettres. Le CIM ne donnant que des notes en anglais, lang doit être "en" ici.
			Definition class ("Classe de définition")	MDC_P021	Dans le cas d'une extrémité d'association, sa classe de définition doit être affectée à la classe à laquelle elle est reliée.
			Property data element type ("Type d'élément de données de propriété")	MDC_P020	Pour une propriété issue d'une extrémité d'association, la valeur de cet attribut doit être mise à "INVERSIBLE_P_DET", pour les autres propriétés du CIM, la valeur de cet attribut doit être mise à "NON_DEPENDET_P_DET"
			Condition	MDC_P028	Pour une propriété issue d'une extrémité d'association avec navigabilité, l'ID de la propriété représentant l'extrémité d'association de l'autre côté doit être décrit dans cet attribut.
			Applicable relations ("Relations applicables")	MDC_P230	Cet attribut sert à décrire l'association contenant cette extrémité d'association.
			Remark ("Remarque")	MDC_P007_2	Cet attribut sert à décrire l'aggregationKind dont la valeur est "none" ("aucune"), "shared" ("partagée") ou "composition" ("composition").
	[Mult to] (Mult jusqu'à) nom		Data type ("Type de données")	MDC_P022	Pour une propriété issue d'une extrémité d'association, il est requis d'avoir le datatype comme "class reference type" ("type de référence de classe"), avec une référence à sa classe reliée. Dans le cas de l'exemple d'extrémité d'association du Tableau 4, on doit la définir comme une propriété avec le type de données suivant: CLASS_REFERENCE_TYPE(ICID de la classe BaseVoltage)
	Type				

Soit une association simple indiquée à la Figure 2 et une agrégation indiquée à la Figure 3 comme des échantillons pour expliquer les règles de transformation.

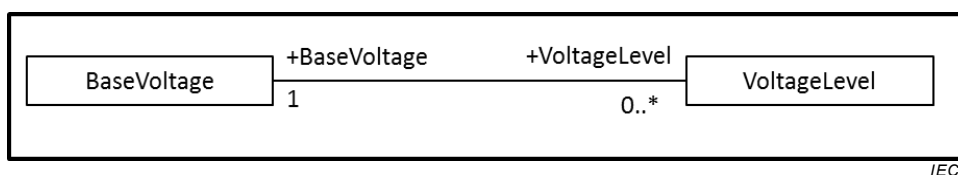


Figure 2 – Exemple d'association simple
(provenant la figure 3 de l'IEC 61970-301:2013)

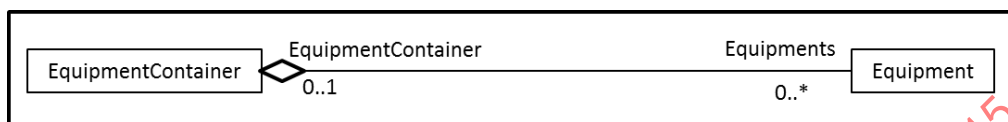


Figure 3 – Exemple d'agrégation
(provenant de la figure 4 de l'IEC 61970-301:2013)

Pour ces deux échantillons, le Tableau 7 présente leurs descriptions du POM correspondantes. Chaque propriété du Tableau 7 représente une extrémité d'association avec ses multiplicités.

Tableau 7 – Extrémités d'association du CIM décrites comme propriétés du POM

MDC_P001_6	MDC_P004_1. en	MDC_P005. en	MDC_P007_2. en	MDC_P021	MDC_P230	MDC_P020	MDC_P028	MDC_P022
Code	Preferred name (Nom préférentiel)	Definition (Définition)	Remark (Remarque)	Definition class (Classe de définition)	Applicable relations (Relations applicables)	Property data element type (Type d'élément de données de propriété)	Condition	Data type (Type de données)
Property_cod e1	VoltageLevel	Les VoltageLevel s ayant ce BaseVoltage.	#aggregation Kind=none	ICID de BaseVoltage	Asso1	INVERSIBLE_P_DET		List(0,n) of class_referen ce_type(ICID of class VoltageLevel)
Property_cod e2	BaseVoltage	La tension de base utilisée pour tous les équipements au sein du VoltageLevel.	#aggregation Kind=none	ICID de la classe VoltageLevel	Asso1	INVERSIBLE_P_DET		class_referen ce_type of(ICID of class BaseVoltage)
Property_cod e3	Equipments	L'association est utilisée dans la hiérarchie de dénomination .	#aggregation Kind=shared	ICID de EquipmentCo ntainer	Asso2	INVERSIBLE_P_DET		List(0,n) of class_referen ce_type(ICID of class Equipment)
Property_cod e4	EquipmentCo ntainer		#aggregation Kind=none	ICID de Equipment	Asso2	INVERSIBLE_P_DET		List(0,1) of class_referen ce_type(ICID of class EquipmentCo ntainer)

5.4.4.2 Nom d'une extrémité d'association

Sachant que le CIM ne nomme pas une association, mais donne un nom à chaque extrémité d'une association, le nom de l'extrémité d'association, par exemple, "VoltageLevel", est transformé en "short name" (nom court) d'une propriété du POM qui est contenu dans une

classe avec le nom préférentiel "BaseVoltage". En même temps, le nom de l'extrémité d'association doit être copié vers "preferred name" (nom préférentiel) de la même propriété du POM comme nom provisoire initial, la première fois qu'il y a transformation du CIM vers le POM.

La multiplicité d'une extrémité d'association à la classe cible, définie comme [mult to] ([mult jusqu'à]) dans le CIM, doit être décrite dans le datatype comme List(0,n) de class_reference_type. Par exemple, l'extrémité d'association "VoltageLevel" décrite dans le premier enregistrement du Tableau 7, la multiplicité à la classe cible, c'est-à-dire à la classe "BaseVoltage", est spécifiée comme " (0,n) " dans le datatype comme List(0,n) de class_reference_type. Pour le deuxième enregistrement du Tableau 7, la multiplicité à la classe cible, c'est-à-dire, à la classe "VoltageLevel", est omise car la multiplicité est "1". Les détails sont expliqués au paragraphe 5.4.4.8.

5.4.4.3 Description

Une "description" qui explique une association doit être transformée en un attribut "Note" (MDC_P007_1.<lang>) de la propriété du POM où <lang> est supposé être "en". En même temps, sachant que le POM exige une "definition"(MDC_P005.<lang>) pour chaque propriété, la note du CIM d'origine doit être copiée vers l'attribut "definition" de la même propriété du POM la première fois qu'il y a transformation du CIM vers le POM. La "definition" dans le POM est mise à jour selon la procédure de maintenance pour les normes de bases de données spécifiée dans le supplément aux Directives ISO/IEC.

5.4.4.4 Remark ("Remarque")

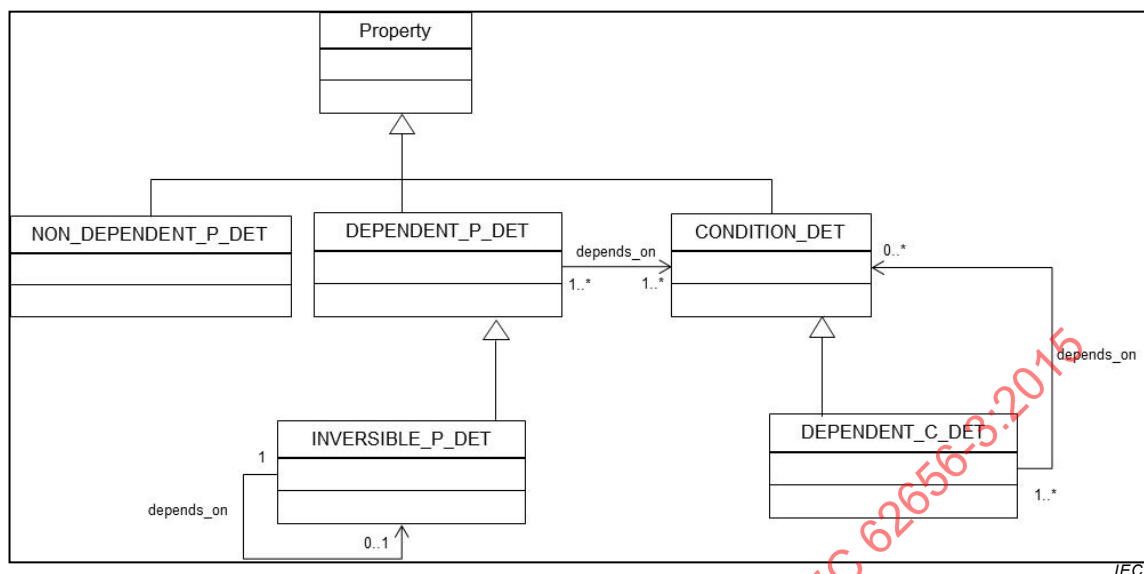
Dans la présente partie de la Norme, "Remark" est utilisé pour décrire l'"aggregationKind" de chaque extrémité d'association. L'"aggregationKind" qui est le même que celui de l'UML, peut avoir une valeur de "none" ("aucune") pour une association normale, ou "shared" ("partagée") pour une agrégation, ou "composition" pour une composition.

Dans la présente partie de la Norme, afin de distinguer la définition d'aggregationKind de l'autre Remark, l'aggregationKind doit être décrit avec le format suivant:

#AggregationKind=none|shared|composition

Les échantillons sont indiqués dans la colonne "Remark" du Tableau 7.

5.4.4.5 Property data element type ("Type d'élément de données de propriété") et condition



Légende

Anglais	Français
Property	Propriété

Figure 4 – Définition de type d'élément de données de propriété (property data element type definition)

Dans la présente norme, "INVERSIBLE_P_DET" est ajouté comme un nouveau "property data element type" ("type d'élément de données de propriété") afin de décrire l'extrémité d'association qui peut avoir une navigabilité. La Figure 4 présente la relation entre la propriété et les sous-types et leurs conditions liées, avec l'ajout de "INVERSIBLE_P_DET". Dans la série IEC 61360 et l'IEC 62656-1, une propriété de "DEPENDENT_P_DET" est supposée dépendre d'une propriété du type "CONDITION_DET". Dans la présente norme, une propriété de "INVERSIBLE_P_DET" peut dépendre d'une autre propriété de "INVERSIBLE_P_DET". C'est pourquoi une propriété issue d'une extrémité d'association peut dépendre d'une propriété issue d'une extrémité d'association du côté opposé. Pour d'autres utilisations de "INVERSIBLE_P_DET", il n'est pas nécessaire que la cardinalité de "depends_on" de "INVERSIBLE_P_DET" soit 0 ou 1, mais elle peut être mise à n'importe quel nombre selon ce qui est nécessaire.

Dans la présente norme, le "INVERSIBLE_P_DET" combiné avec l'attribut de "Condition" est appliqué pour enregistrer la navigabilité d'une extrémité d'association.

Dans l'UML, il y a trois types de navigabilité pour une extrémité d'association:

– Non navigable

NOTE 1 Dans le diagramme de classe UML, une extrémité d'association avec non-navigabilité est représentée par un signe X.

– Navigable

NOTE 2 Dans le diagramme de classe UML, une extrémité d'association avec navigabilité est représentée par une flèche.

– Unspecified (Non spécifié)

NOTE 3 Dans le diagramme de classe UML, une extrémité d'association avec navigabilité non spécifiée est représentée par une ligne simple. Dans le CIM, la plupart des extrémités d'association sont définies avec navigabilité non spécifiée.

Chaque extrémité d'association doit être transformée en propriété du POM avec le "property data element type" comme "INVERSIBLE_P_DET". Pour une association avec deux extrémités d'association navigables, la "Condition" de la propriété (par exemple, Property_A) représentant l'extrémité d'association doit avoir la valeur de l'ID de propriété (par exemple, Property_B) représentant l'extrémité d'association du côté opposé.

5.4.4.6 Applicable relations ("Relations applicables")

Pour une propriété du POM issue d'une extrémité d'association du CIM, cet attribut est utilisé pour décrire l'association qui contient l'extrémité d'association. Comme l'association elle-même est décrite comme une instance de la relation du POM, l'ICID correspondant de la relation doit être décrit dans cet attribut. Les exemples sont indiqués dans le Tableau 7.

5.4.4.7 Native and inherited association ends ("Extrémités d'association natives et héritées")

Dans le CIM, il existe deux sortes d'extrémités d'association pour une classe: les extrémités d'association natives et les extrémités d'association héritées de ses classes supérieures.

- Chaque extrémité d'association native doit être transformée en une propriété du POM qui a sa "Definition class" ("classe de définition") dans la classe où la propriété est définie. En même temps, on doit ajouter la propriété aux "Applicable properties" ("propriétés applicables") (expliquée au 6.2.15) de la classe du POM.
- Les extrémités d'association héritées d'une classe doivent provenir des "Applicable properties" ("propriétés applicables") de sa classe supérieure et de ses classes ancêtres.

5.4.4.8 Data type ("Type de données")

Le POM est doté d'une propriété de "class_reference_type" qui est capable de décrire une association entre deux classes. Par conséquent, une propriété transformée d'une extrémité d'association du CIM dans le POM doit être de "class_reference_type". En outre, pour décrire la multiplicité des instances d'une classe à laquelle est reliée l'extrémité d'association, le POM peut utiliser un ensemble ordonné de propriétés de "class_reference_type", c'est-à-dire que le data type de la propriété doit être décrit comme étant "List(0,n) of class_reference_type(ICID)" où ICID signifie l'ID global de la classe à laquelle est reliée l'extrémité d'association. Lorsque la multiplicité est 1, il n'est pas nécessaire de définir le data type comme "List(1) of class_reference_type(ICID)" mais seulement comme "class_reference_type(ICID)".

Pour une association de type "Composition", il est recommandé d'utiliser le data type de "class_instance_type" au lieu de "class_reference_type" car le premier type impose que les objets composés doivent avoir les mêmes cycles de vie que les objets composites.

5.5 Classe Enumeration ("énumération") du CIM

5.5.1 Généralités

Dans le CIM, les stéréotypes UML de "primitive", "enumeration" et "datatype" sont appliqués. Une classe avec le type "primitive" du stéréotype représente une classe commune, une classe d'un type "enumeration" du stéréotype représente une classe "enumeration" et une classe avec un datatype du stéréotype représente une classe "datatype". Une classe "enumeration" définit toutes les valeurs possibles pour cette énumération et la classe "enumeration" n'a pas de structures hiérarchiques.

Le POM définit un objet comme l'énumération qui est fournie pour décrire les valeurs possibles d'une propriété. Par conséquent, l'énumération du CIM doit être décrite comme une énumération du POM.

Le POM définit un objet comme datatype qui doit être fourni pour décrire le datatype du CIM. Le détail du datatype est expliqué au 5.7.

Tableau 8 – Mise en correspondance d'une énumération du CIM et d'une énumération du POM

Métamodèle UML dans le CIM		Métamodèle POM		
Objet UML	Définition du CIM	Objet du POM	Nom d'attribut	ID d'attribut
Enumeration ("Énumération")	Name ("Nom")	Enumeration ("Énumération")	Short name ("Nom court")	MDC_P004_3.<lang>
	Description		Note	MDC_P007.<lang>

5.5.2 Name ("Nom")

Le nom de la classe "enumeration" du CIM doit être transformé en "short name" de l'énumération du POM. En même temps, le nom de la classe "enumeration" doit être copié vers "Preferred name" de la même énumération du POM comme nom provisoire initial, la première fois qu'il y a transformation du CIM vers le POM. Le "Preferred name" dans le POM est mis à jour selon la procédure de maintenance pour les normes de bases de données spécifiée dans le supplément aux Directives ISO/IEC.

5.5.3 Description

La "Description" de l'attribut de la classe "enumeration" du CIM doit être transformée en l'attribut "Note" d'une énumération du POM. Et lorsque c'est la première fois qu'il y a transformation du CIM vers le POM, la "Note" doit être aussi copiée vers l'attribut "Definition" du POM. La "Note" dans le POM n'est pas modifiée pour conserver l'originalité du CIM, et la "Definition" dans le POM est mise à jour selon la procédure de maintenance pour les normes de bases de données spécifiée dans le supplément aux Directives ISO/IEC.

5.6 Attribut de Enumeration ("énumération") du CIM

5.6.1 Généralités

Dans le CIM, "enumeration" est défini comme une classe et ses valeurs possibles sont définies comme étant ses attributs natifs. Le POM définit un objet comme un terme servant à donner un moyen de définir des termes utilisés à répétition, des constantes ou des valeurs individuelles utilisées dans un modèle de données. Lors d'une transformation du CIM vers le POM, chaque valeur d'énumération possible du CIM doit être définie comme une instance du terme du POM. Comme indiqué au Tableau 9, les attributs d'une énumération doivent être transformés en instances du terme du POM.

Tableau 9 – Mise en correspondance d'un attribut d'énumération du CIM et d'un terme du POM

Métamodèle UML dans le CIM		Métamodèle POM		
Objet UML	Définition du CIM	Objet du POM	Nom d'attribut	ID d'attribut
Attribute of enumeration ("Attribut d'énumération")	Name ("Nom")	Terme	Short name ("Nom court")	MDC_P004_3.<lang>
	Description		Note	MDC_P007.<lang>

5.6.2 Name ("Nom")

Le nom d'un attribut d'énumération doit être transformé en "Short name" du terme du POM. En même temps, le nom de l'attribut d'énumération du CIM doit être copié vers "Preferred name" du même terme du POM comme nom provisoire initial, la première fois qu'il y a

transformation du CIM vers le POM. Le "Preferred name" dans le POM est mis à jour selon la procédure de maintenance pour les normes de bases de données spécifiée dans le supplément aux Directives ISO/IEC.

5.6.3 Description

La "Description" d'un attribut d'énumération doit être transformée en "Note" pour le terme du POM. Et lorsque c'est la première fois qu'il y a une transformation du CIM vers le POM, la "Note" doit être copiée vers l'attribut "Definition" du POM. La "Note" dans le POM ne doit pas être modifiée pour conserver l'originalité du CIM, et la "Definition" dans le POM est mise à jour selon la procédure de maintenance pour les normes de bases de données spécifiée dans le supplément aux Directives ISO/IEC.

5.7 Data type ("Type de données")

5.7.1 Datatype de base du CIM

Dans le modèle CIM, il y a deux sortes de types de données (datatype): le datatype de base "String" (chaîne), "Integer" (entier), "Float" (flottant) et "Boolean" (booléen), qui peuvent être utilisées pour décrire d'autres types de données. L'autre sorte est le datatype (type de données) nommé qui est défini avec le datatype (type de données) de base. Le datatype nommé est défini comme une classe avec un stéréotype de datatype qui a un triplet d'attribut {value, unit, multiplier} (respectivement valeur, unité, multiplicateur).

Tableau 10 – Mise en correspondance du data type de base du CIM et du data type simple du POM

Data type de base du CIM		Data type simple du POM (MDC_P022)
Name ("Nom")	Définition dans le CIM	
Integer	Un nombre entier. La plage n'est pas spécifiée et n'est pas limitée.	INT_MEASURE_TYPE
boolean	Un type avec un espace de valeurs "true" (vrai) et "false" (faux)	BOOLEAN_TYPE
Float	Un nombre en virgule flottante. La plage n'est pas spécifiée et n'est pas limitée.	REAL_MEASURE_TYPE
String	Une chaîne ("String") constituée d'une séquence de caractères de huit bits. Le codage des caractères est UTF-8. La longueur de la chaîne est indéterminée et illimitée.	STRING_TYPE

Dans le POM, il existe aussi deux sortes de types de données: le data type (type de données) simple et le data type (type de données) nommé. Le data type simple peut être utilisé pour définir un data type nommé.

Lors de la transformation du CIM vers le POM, le data type de base doit être mis en correspondance avec le data type simple du POM correspondant. Le Tableau 10 donne les règles de mise en correspondance entre les data types de base du CIM et les data types simples du POM.

Une différence importante entre le data type de base du CIM et le data type du POM est que: le POM établit une distinction entre le data type avec unités et le data type sans unités. Par exemple, il y a REAL_TYPE pour ceux sans unités et REAL_MEASURE_TYPE pour ceux avec unité. Au contraire, le data type de base du CIM n'est pas lié à une unité. L'unité dans le POM étant mise à "1" par défaut, le data type de base relatif au nombre dans le CIM doit donc être transformé en data type simple du POM.

5.7.2 Data types ("types de données") nommés du CIM

5.7.2.1 Généralités

Dans le CIM, un type de données autre que ceux de base est décrit comme une classe. Il doit être transformé en data type (type de données) nommé dans le POM selon les règles indiquées dans le Tableau 11.

5.7.2.2 Name ("Nom")

Le Nom du datatype doit être transformé en "short name" du datatype du POM. En même temps, le nom du type de données du CIM doit être copié vers "Preferred name" du même type de données du POM comme nom provisoire initial, la première fois qu'il y a transformation du CIM vers le POM. Le "Preferred name" dans le POM est mis à jour selon la procédure de maintenance pour les normes de bases de données spécifiée dans le supplément aux Directives ISO/IEC.

5.7.2.3 Description

La Description du type de données du CIM doit être transformée en "Note" du datatype du POM. Et lorsque c'est la première fois qu'il y a transformation du CIM vers le POM, la "Note" doit être copiée vers l'attribut "Definition" du même datatype du POM. La "Note" dans le POM n'est pas modifiée pour conserver l'originalité du CIM, et la "Definition" dans le POM est mise à jour selon la procédure de maintenance pour les normes de bases de données spécifiée dans le supplément aux Directives ISO/IEC.

Tableau 11 – Mise en correspondance du type de données du CIM et d'un type de données nommé dans le POM

Métamodèle UML dans le CIM		Métamodèle POM			Description
Objet UML	Définition du CIM	Objet du POM	Nom d'attribut	ID d'attribut	
Datatype ("type de données")	Name ("Nom")	Datatype ("type de données")	Short name ("nom court")	MDC_P004_3.<lang>	Le nom d'un type de données du CIM doit être décrit avec son attribut.
	Description		Note	MDC_P007_1.<lang>	<lang> dans le POM doit être un code ISO 639 de deux lettres. Le CIM ne donnant que des notes en anglais, lang doit être "en" ici.
			Datatype ("type de données")	MDC_P022	Un data type simple du POM correspondant est décrit ici.
	Unit ("unité")		Unit in text ("Unité en texte")	MDC_P023_1	Cet attribut est utilisé pour décrire l'unité d'un type de données.
	Multiplier ("Multiplicateur")		Alternative units ("Unités alternatives")	MDC_P111	Cet attribut est utilisé pour décrire les unités alternatives d'un datatype ou d'une propriété.

5.7.2.4 Data type ("Type de données")

Cet attribut doit décrire le type de données simple du POM correspondant au type de données de base du CIM.

5.7.2.5 Unit ("Unité")

Le CIM définit une unité comme étant une partie d'un datatype chaque fois que cela est nécessaire. Le POM peut associer une unité SI à chaque datatype. Par conséquent, l'unité du datatype du CIM doit être transformée en l'unité du datatype du POM.

5.7.2.6 Alternative units ("Unités alternatives")

Le CIM définit un multiplicateur pour une unité destiné à imposer un éventuel préfixe. Dans le CIM de l'IEC 61970-301:2013, le multiplicateur d'unité est une énumération nommée unitMultiplier avec des énumérateurs possibles {p,n,micro,m,c,d,k,M,G,T,none} qui fait partie des préfixes SI. Dans chaque datatype nommé, un multiplicateur d'unité UnitMultiplier [0..1] ou "none" est assigné. Ici, "none" représente le fait qu'aucun préfixe pour une unité n'est disponible pour un datatype. UnitMultiplier[0..1] signifie qu'on ne peut sélectionner plus d'un énumérateur de l'unitMultiplier pour une unité. Cependant, il n'y a pas de contrainte claire indiquant quels unitMultipliers peuvent être sélectionnés et lesquels ne sont pas destinés à une unité définie dans un type de données.

Afin de clarifier la définition d'une unité et de son préfixe, dans la présente partie de la Norme, l'attribut de "Alternative units"(MDC_P111) doit être utilisé pour représenter toutes les unités pouvant être sélectionnées pour une unité SI.

5.8 Paquetage

5.8.1 Généralités

Le diagramme des paquetages du CIM définit la portée du nom pour chaque objet dans le CIM. De plus, les paquetages UML peuvent avoir des relations entre eux pour toutes les parties ("part-whole"). Dans le POM, un objet relation est utilisé pour décrire les informations des paquetages. Le Tableau 12 présente la règle générale pour transformer un objet paquetage du CIM en un objet relation du POM. Pour expliquer les règles de transformation, on applique un exemple de définition du paquetage du CIM indiqué à la Figure 5. Les définitions de relation du POM correspondantes sont indiquées dans le Tableau 13.

Tableau 12 – Mise en correspondance d'un paquetage du CIM et d'une relation du POM

Métamodèle UML dans le CIM			Métamodèle POM		Description
Objet UML	Définition du CIM	Objet du POM	Nom d'attribut	ID d'attribut	
Package ("Paquetage")	Name ("Nom")	Relation	Short name ("Nom court")	MDC_P004_3. <lang>	On doit décrire le nom d'un paquetage dans cet attribut du POM.
	Description		Note	MDC_P007_1. <lang>	On doit décrire l'explication d'un paquetage dans cet attribut du POM. <lang> dans le POM doit être un code ISO 639 de deux lettres. Le CIM ne donnant que des notes en anglais, lang doit être "en" ici.
	Version		Version	MDC_P002_1	On doit décrire la version d'un paquetage dans cet attribut.
	Date		Date of current definition ("Date de la définition courante")	MDC_P003_2	On doit décrire la date d'un paquetage dans cet attribut.
			Relation type ("Type de relation")	MDC_P200	Le paquetage du CIM doit être transformé en la relation du POM avec le type de relation "function" (fonction). Autrement, par exemple, pour une relation de collections de classes, la valeur de cet attribut doit être mise à "Predication".
			Domain of the relation ("Domaine de la relation")	MDC_P201	Cet attribut est utilisé pour décrire des éléments d'une relation lorsque relation type(MDC_P200) est "Predication".
	Package including elements ("Paquetage comprenant des éléments")		Domain of the function ("Domaine de la fonction")	MDC_P202	On doit énumérer ici les éléments du paquetage. On doit définir une collection de chaque sorte d'éléments comme une instance de relation. De plus, la valeur de cet attribut est valide seulement lorsque relation type(MDC_P200) est "Function".
	Package relation ("Relation de paquetages")		Codomain of the function ("Co-domaine de la fonction")	MDC_P203	On doit décrire ici l'ICID du paquetage parent.
			Domain element type ("Type d'élément de domaine")	MDC_P208	Cet attribut sert à décrire le "element type" ("type d'élément") disponible dans le MDC_P201. Les valeurs possibles pour cet attribut sont la métapropriété du type d'objet. Par exemple: (1)MDC_C002, pour les classes (2)MDC_C003, pour les propriétés (3)MDC_C006, pour le data type
			Role of the relation ("Rôle de la relation")	MDC_P210	La valeur de cet attribut doit être mise à; (1) "CIM package" ("Paquetage du CIM") alors que cette relation est issue d'un paquetage du CIM, (2) "CIM association" ("Association du CIM") alors que cette relation est issue d'une association du CIM. Sinon la valeur de cet attribut doit être mise à zéro.

5.8.2 Objet relation du POM

Pour l'exemple indiqué à la Figure 5, le paquetage "Generation" inclut les paquetages **Production** et **GenerationDynamics**. Pour chaque paquetage, il y a des objets qui en font partie, tels que classe, propriété, data type, etc.

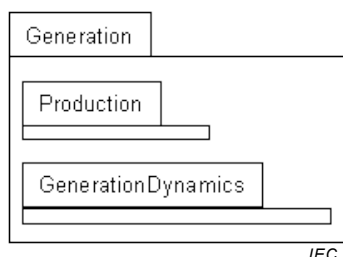


Figure 5 – Exemple de paquetage du CIM
(provenant de la Figure 1 de l'IEC 61970-301:2013)

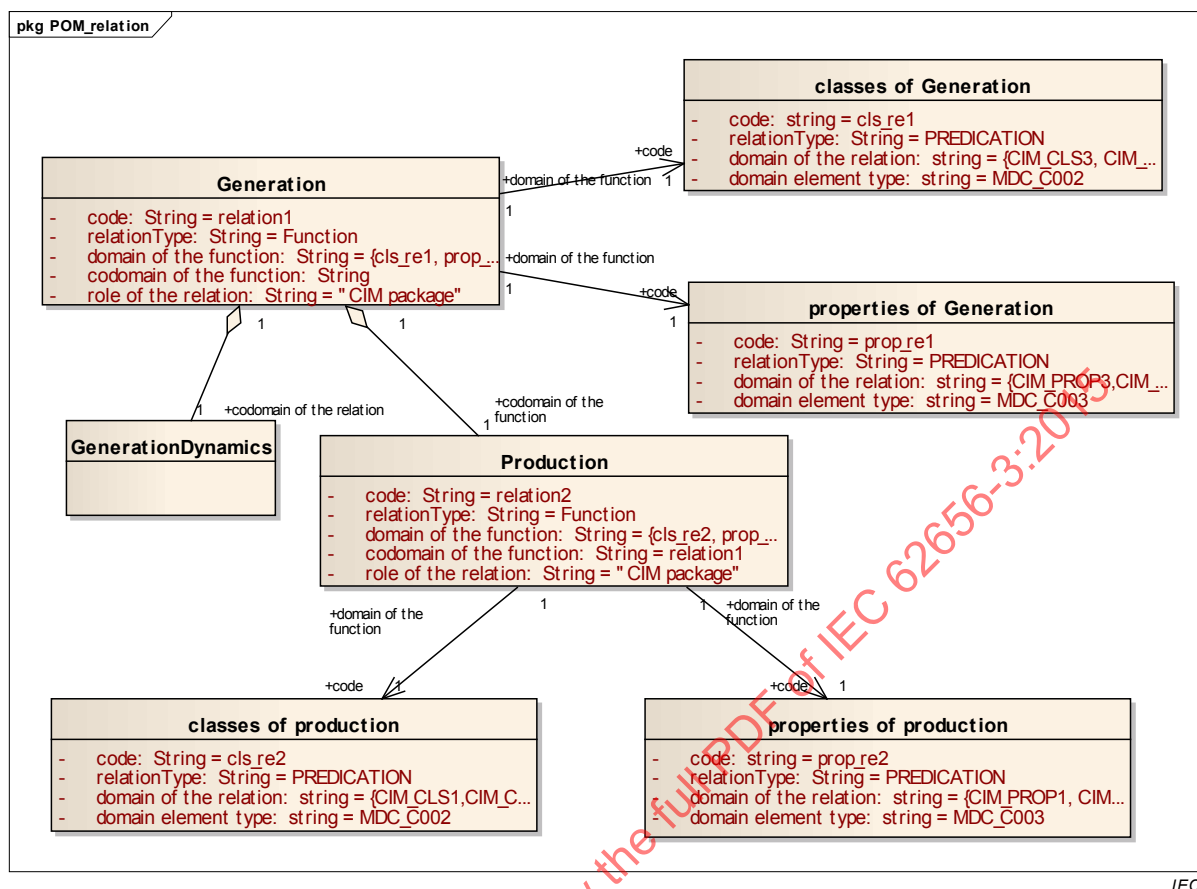
Dans le POM, la relation est une sorte d'ensemble. Par exemple, si "A et B sont liés par une relation R1", cela signifie que A et B appartiennent au même ensemble de R1.

Ainsi, en ce qui concerne les paquetages du CIM, les instances de relation suivantes sont définies en conséquence pour les classes, les propriétés et les datatypes (types de données) d'un paquetage.

- Une instance de relation du POM est définie comme une collection de toutes les classes comprises dans un paquetage.
- Une instance de relation du POM est définie comme une collection de toutes les propriétés comprises dans un paquetage.
- Une instance de relation du POM est définie comme une collection de tous les datatypes (types de données) compris dans un paquetage.

L'objet relation du POM doit utiliser les attributs ci-après pour décrire la relation entre les paquetages CIM, le paquetage du CIM et ses éléments;

- L'Attribut "Domain of the function(MDC_P202)" doit être utilisé pour décrire la possession entre un paquetage et les objets inclus, tels que les classes, les propriétés et les datatypes,
- L'attribut "Codomain of the function(MDC_P203)" doit être utilisé pour décrire la relation de toutes les parties ("part-whole") entre deux paquetages. On doit énumérer l'ICID du paquetage "tout" au "Codomain of the function" du paquetage "partie".
- L'attribut "Role of the relation(MDC_P210)" doit être utilisé pour décrire un type générique de relation. Lorsqu'une relation est utilisée pour décrire un paquetage du CIM, la valeur de cet attribut doit être mise à "CIM package" (paquetage du CIM).



Légende

Anglais	Français
classes of Generation	classes de Generation
properties of Generation	propriétés de Generation
classes of production	classes de production
properties of production	propriétés de production

Figure 6 – Explication du modèle de relation du POM en UML

La Figure 6 explique schématiquement l'utilisation des objets "relation" en UML pour l'exemple de paquetage donné à la Figure 5. Chaque paquetage représenté comme une instance de relation est décrit ici comme une classe UML. Les relations entre les paquetages "Generation", "GenerationDynamics" et "Production" sont décrites par l'attribut "Codomain of the relation" du paquetage "partie".

Dans le même temps, on doit décrire comme une instance de relation les classes possédées par un paquetage. Par exemple, "classes of Generation" (classes de Generation) est défini comme une relation "cls_re1" et associé au paquetage Generation avec l'attribut "Domain of the relation".

5.8.3 Paquetages CIM décrits en relation du POM

5.8.3.1 Généralités

La description de relations des trois paquetages du POM à la Figure 5 est présentée dans le Tableau 13. Chaque paquetage doit être décrit comme une instance de l'objet "relation". Chaque élément utilisé par le paquetage du CIM est expliqué dans ce paragraphe 5.8.3.

Tableau 13 – Paquetage du CIM décrit comme relation du POM

MDC_P001_13	MDC_P004_1. <lang>	MDC_P007_1. <lang>	MDC_P200	MDC_P201	MDC_P202	MDC_P203	MDC_P208	MDC_P210
Code	Preferred name ("Nom préférentiel")	Note	Relation type ("Type de relation")	Domain of the relation ("Domaine de la relation")	Domain of the function ("Domaine de la fonction")	CoDomain of the function ("Co-domaine de la fonction")	Domain element type ("Type d'élément de domaine")	Role of the relation ("Rôle de la relation")
Relation1	Generation	Ce paquetage contient des paquetages qui ont des informations pour Unit Commitment (Gestion de la production) et Economic Dispatch of Hydro and Thermal Generating Units (Dispatching économique des unités de production thermiques et hydrauliques), Load Forecasting (Prévision de la charge), Automatic Generation Control (Conduite automatique de production), et Unit Modeling for Dynamic Training Simulator (Modélisation d'unité pour les simulateurs d'entraînement).	Function («Fonction»)		{Cls_re1,Prop_re1 }			CIM package ("Paquetage du CIM")
Relation2	Production	Le paquetage production est responsable des classes qui décrivent diverses sortes de générateurs. Ces classes donnent également des informations relatives aux coûts de production qui sont utilisées pour allouer de manière économique la demande parmi les unités engagées et calculer les quantités de réserve.	Function («Fonction»)		{Cls_re2,Prop_re2}	Relation1		CIM package ("Paquetage du CIM")

MDC_P001_13	MDC_P004_1. <lang>	MDC_P007_1. <lang>	MDC_P200	MDC_P201	MDC_P202	MDC_P203	MDC_P208	MDC_P210
Code	Preferred name ("Nom préférentiel")	Note	Relation type ("Type de relation")	Domain of the relation ("Domaine de la relation")	Domain of the function ("Domaine de la fonction")	CoDomain of the function ("Co-domaine de la fonction")	Domain element type ("Type d'élément de domaine")	Role of the relation ("Rôle de la relation")
Relation3	generationDynamics	Le paquetage GenerationDynamics contient des moteurs primaires, tels que turbines et chaudières, qui sont nécessaires aux processus de simulation et d'éducation.	Function («Fonction»)		{Cls_re3, Prop_re3}	Relation1		CIM package ("Paquetage du CIM")
Cls_re1	Classes of Generation ("Classes de "Generation")		PREDICATION	{CIM_cls1, CIM_cls2,...}			MDC_C002	
Prop_re1	Properties of Generation ("Propriétés de "Generation")		PREDICATION	{CIM_prop1, CIM_prop2,...}			MDC_C003	
Cls_re2	Classes of production ("Classes de "production")		PREDICATION	{CIM_cls1, CIM_cls12,...}			MDC_C002	
Prop_re2	Properties of production ("Propriétés de "production")		PREDICATION	{CIM_prop11, CIM_prop12,...}			MDC_C003	

5.8.3.2 Name ("Nom")

Le Nom du paquetage doit être transformé en "Short name" de la relation du POM. En même temps, le nom du paquetage du CIM doit être copié vers "Preferred name" de la même relation du POM comme nom provisoire initial, la première fois qu'il y a transformation du CIM vers le POM. Le "Preferred name" dans le POM est mis à jour selon la procédure de maintenance pour les normes de bases de données spécifiée dans le supplément aux Directives ISO/IEC.

5.8.3.3 Description

Le CIM donne une explication relative à un paquetage qui doit être transformé en "Description" de la relation du POM. Et lorsque c'est la première fois qu'il y a transformation du CIM vers le POM, la "Note" doit être copiée vers l'attribut de "Definition" de la même relation du POM. La "Note" dans le POM n'est pas modifiée pour conserver l'originalité du CIM, et la "Définition" dans le POM est mise à jour selon la procédure de maintenance pour les normes de bases de données spécifiée dans le supplément aux Directives ISO/IEC.

5.8.3.4 Relation type ("Type de relation")

Un paquetage du CIM doit être représenté en utilisant un objet relation du POM dont le "Relation type"(MDC_P200) doit être spécifié comme étant "Function". Une relation fonctionnelle signifie que les instances aux extrémités de la relation ont une relation n à 1 . En termes plus simples, pour chaque élément dans un paquetage, il y a un seul paquetage auquel l'élément appartient. De ce fait, la relation qui représente un paquetage prend la forme de "Function"

5.8.3.5 Relation d'inclusion

Dans le CIM, tous les éléments appartiennent à un certain paquetage. Pour chaque élément, cette relation est décrite avec l'attribut "CIM Package"(MDC_P211), synonyme de l'attribut "Segment" de l'IEC 62656-1, en plus de l'expression de la relation par une relation fonctionnelle, dans laquelle les éléments sont énumérés dans l'attribut de "Codomain of the relation" (MDC_P203).

6 Description du CIM au format IEC 62656

6.1 Généralités

Pour représenter l'ontologie du CIM par le POM, chaque objet CIM dans le format du POM doit avoir les attributs correspondants énumérés dans l'Annexe B. L'article 6 approfondit un certain nombre des attributs fondamentaux pour chaque objet. Tous les objets du POM ont les attributs communs suivants:

- **Code**
- **Version number** ("Numéro de version")
- **Revision number** ("Numéro de révision")
- **Date of original definition** ("Date de la définition originale")
- **Date of current version** ("Date de la version actuelle")
- **Preferred name** ("Nom préférentiel")
- **Short name** ("Nom court")
- **Definition** ("Définition")
- **Note**
- **Alternate ID** ("ID alternatif") (**pour CIM UML ID**)
- **Segment** (**pour Paquetage du CIM**)

À l'attribut "Code" de chaque objet est assigné un ICID unique. Les définitions détaillées de ces attributs pour l'objet classe sont expliquées au paragraphe 6.2. Bien que ces attributs soient communs parmi tous les objets, certains de ces attributs sont facultatifs et leurs définitions détaillées pour d'autres objets sont omises. Certains attributs supplémentaires sont également nécessaires dans chaque objet. Les attributs utilisés dans la présente Partie de l'IEC 62656 sont expliqués de façon plus approfondie dans la présente section

De plus, chaque attribut exige plusieurs éléments définis dans la présente section.

- **Destination ID** ("Identificateur de destination"): cet élément doit être utilisé pour décrire le data identifier (identificateur de données (DI)) d'une séquence d'ICID assignée à un attribut dans l'IEC 62656-1.
- **Destination name** ("Nom de destination"): cet élément doit être utilisé pour décrire le nom de l'attribut spécifié par le "Destination ID" donné ci-dessus conformément à l'IEC 62656-1 ou dans la présente norme.
- **Destination definition** ("Définition de destination"): cet élément doit être utilisé pour décrire l'attribut "definition" dans l'IEC 62656-1.
- **Destination value format** ("Format de valeur de destination"): cet élément doit être utilisé pour décrire le format pour la valeur de l'attribut défini dans l'IEC 62656-1.
- **Original CIM definition** ("Définition du CIM d'origine"): cet élément doit être utilisé pour décrire si une définition correspondante de cet attribut est disponible dans le CIM ou non.
- **Requirement** ("Exigence"): cet élément doit être utilisé pour expliquer si cet attribut est "Mandatory" (obligatoire) ou "Optional" ("facultatif") pour un objet.
- **Name of source CIM object** ("Nom de l'objet source du CIM "): cet élément doit être appliqué pour décrire le nom assigné à cet attribut dans le CIM si disponible.
- **Requirement for mapping** ("Exigence relative à la mise en correspondance"): cet élément doit être utilisé pour décrire les notes et les exigences lors de la traduction de la définition du CIM vers cet attribut IEC 62656.

6.2 Class ("Classe")

6.2.1 Généralités

Le Tableau B.1 donne une liste des parties essentielles des attributs possibles pour une classe dans le CIM paqueté, y compris ceux qui sont obligatoires et facultatifs. Le paragraphe 6.2 explique les attributs communs et les attributs essentiels de la classe.

6.2.2 Code

- **Destination ID**: MDC_P001_5
- **Destination name**: Code
- **Destination definition**: même que l'IEC 62656-1
- **Destination value format**: même que l'IEC 62656-1
- **Original CIM definition**: non disponible
- **Requirement**: Mandatory ("Obligatoire")
- **Name of source CIM object**: N/A ("Non disponible")
- **Requirement for mapping**: La valeur doit être décrite selon la spécification ICID de l'IEC 62656-1. Cet attribut n'est pas défini dans la norme CIM d'origine. Cependant, comme il est l'identificateur d'une classe, cet attribut doit être ajouté à chaque classe dans le CIM paqueté.

6.2.3 Version number ("Numéro de version")

- **Destination ID**: MDC_P002_1
- **Destination name**: Version number ("Numéro de version")
- **Destination definition**: même que l'IEC 62656-1

- **Destination value format:** même que l'IEC 62656-1
- **Original CIM definition:** non disponible
- **Name of source CIM object:** Mandatory ("Obligatoire")
- **Name of source CIM object:** N/A ("Non disponible")
- **Requirement for mapping:** La version doit être identique à l'élément VI de l'ICID décrit dans le code. Cet attribut n'est pas défini dans la norme CIM d'origine. Comme il est utilisé pour identifier les mises à jour de la classe, qui ont une influence sur les instances de la classe, il doit être ajouté à la classe du CIM pour la gestion des mises à jour du CIM.

6.2.4 Revision number ("Numéro de révision")

- **Destination ID:** MDC_P002_2
- **Destination name:** Revision number ("Numéro de révision")
- **Destination definition:** même que l'IEC 62656-1
- **Destination value format:** même que l'IEC 62656-1
- **Original CIM Definition:** disponible dans l'IEC 61979-301:2013
- **Requirement:** Mandatory ("Obligatoire")
- **Name of source CIM object:**
- **Requirement for mapping:** La révision ne doit pas altérer les plages d'instances. Cet attribut n'est pas défini dans la norme CIM d'origine. Comme il est utilisé pour identifier les mises à jour de la classe, qui n'ont pas d'influence sur la plage d'instances, il doit être ajouté à la classe du CIM.

6.2.5 Date of original definition ("Date de la définition originale")

- **Destination ID:** MDC_P003_1
- **Destination name:** Date of original definition ("Date de la définition originale")
- **Destination definition:** même que l'IEC 62656-1
- **Destination value format:** même que l'IEC 62656-1
- **Original CIM definition:** non disponible
- **Requirement:** Mandatory ("Obligatoire")
- **Name of source CIM object:** N/A ("Non disponible")
- **Requirement for mapping:** La date de la définition d'origine doit être décrite dans cet attribut. Sa valeur doit être conforme à l'ISO 8601. Le CIM d'origine n'a pas de définition correspondante pour cet attribut.

6.2.6 Date of current version ("Date de la version actuelle")

- **Destination ID:** MDC_P003_2
- **Destination name:** Date of current definition ("Date de la définition courante")
- **Destination definition:** même que l'IEC 62656-1
- **Destination value format:** même que l'IEC 62656-1
- **Original CIM definition:** non disponible
- **Requirement:** Mandatory ("Obligatoire")
- **Name of source CIM object:** N/A ("Non disponible")
- **Requirement for mapping:** La date de la version actuelle doit être décrite dans cet attribut. Sa valeur doit être conforme à l'ISO 8601. Le CIM d'origine n'a pas de définition correspondante pour cet attribut.

6.2.7 Preferred name ("Nom préférentiel")

- **Destination ID:** MDC_P004_1.<lang>
- **Destination name:** Preferred name ("Nom préférentiel")

- **Destination definition:** même que l'IEC 62656-1
- **Destination value format:** même que l'IEC 62656-1
- **Original CIM definition:** non disponible
- **Requirement:** Mandatory ("Obligatoire")
- **Name of source CIM object:** Name ("Nom")
- **Requirement for mapping:** Le CIM d'origine donne un "name" (nom) à chaque objet "class" (classe). La première fois qu'il y a transformation du CIM vers le POM, le nom d'origine du CIM doit être copié vers cet attribut et il est mis à jour selon la procédure de maintenance pour les normes de base de données spécifiée dans le supplément aux Directives ISO/IEC.

6.2.8 Short name ("Nom court")

- **Destination ID:** MDC_P004_3.<lang>
- **Destination name:** Short name ("Nom court")
- **Destination definition:** même que l'IEC 62656-1
- **Destination value format:** même que l'IEC 62656-1
- **Original CIM definition:** disponible dans l'IEC 61970-301:2013
- **Requirement:** Mandatory ("Obligatoire")
- **Name of source CIM object:** N/A ("Non disponible")
- **Requirement for mapping:** Le nom d'une classe du CIM doit être mis en correspondance avec l'attribut "short name" de la classe du POM correspondante.

6.2.9 Definition ("Définition")

- **Destination ID:** MDC_P005.<lang>
- **Destination name:** Definition ("Définition")
- **Destination definition:** même que l'IEC 62656-1
- **Destination value format:** même que l'IEC 62656-1
- **Original CIM definition:** non disponible
- **Requirement:** Mandatory ("Obligatoire")
- **Name of source CIM object:** Description
- **Requirement for mapping:** Le CIM d'origine fournit un champ appelé "Description" pour expliquer chaque "class" ("classe"). Lors de la transformation du CIM vers le POM, la description du CIM doit être copiée tant vers l'attribut "Definition" ("définition") que vers l'attribut "Note" du POM. La description dans le POM n'est pas modifiée pour conserver la cohérence avec le CIM, alors que la définition dans le POM est mise à jour conformément à la procédure de maintenance pour les normes de bases de données conforme à la spécification dans les Directives ISO/IEC.

6.2.10 Note

- **Destination ID:** MDC_P007_1.<lang>
- **Destination name:** Note
- **Destination definition:** même que l'IEC 62656-1
- **Destination value format:** même que l'IEC 62656-1
- **Original CIM definition:** disponible dans l'IEC 61970-301:2013
- **Requirement:** Optional ("Facultatif")
- **Name of source CIM object:** Description
- **Requirement for mapping:** Une description CIM doit être mise en correspondance avec l'attribut Note de la classe du POM correspondante.

6.2.11 Superclass ("Superclasse")

- **Destination ID:** MDC_P010
- **Destination name:** Superclass ("Superclasse")
- **Destination definition:** même que l'IEC 62656-1
- **Destination value format:** même que l'IEC 62656-1
- **Original CIM definition:** disponible dans l'IEC 61970-301:2013
- **Requirement:** Mandatory ("Obligatoire")
- **Name of source CIM object:** super class ("superclasse")
- **Requirement for mapping:** Le CIM d'origine définit une relation de généralisation entre une classe et sa super classe afin de décrire la relation hiérarchique entre deux classes. Le POM définit un attribut appelé "Superclass" ("superclasse") à cet effet. Cependant, le POM utilise un ICID pour la désignation de la Superclasse, plutôt qu'un "name" ("nom"). Par conséquent, dans la transformation de la relation de généralisation du CIM en la "Superclass" correspondante du POM, une mise en correspondance entre un nom de classe et son ICID est requise.

6.2.12 Class type (Type de classe)

- **Destination ID:** MDC_P011
- **Destination name:** class type ("type de classe")
- **Destination definition:** même que l'IEC 62656-1
- **Destination value format:** même que l'IEC 62656-1
- **Original CIM definition:** non disponible
- **Requirement:** Mandatory ("Obligatoire")
- **Name of source CIM object:** N/A ("Non disponible")
- **Requirement for mapping:** Dans le POM, la valeur possible de cet attribut est l'une des suivantes:
 - "ITEM_CLASS";
 - "COMPONENT_CLSS";
 - "MATERIAL_CLASS";
 - "FEATURE_CLASS";
 - "ITEM_CLASS_CASE_OF",
 - "COMPONENT_CLASS_CASE_OF";
 - "MATERIAL_CLASS_CASE_OF";
 - "FEATURE_CLASS_CASE_OF".

Une classe du CIM doit toujours être représentée comme un "ITEM_CLASS".

6.2.13 Alternate ID ("ID alternatif")

- **Destination ID:** MDC_P101
- **Destination name:** Alternate ID (ID alternatif)
- **Destination definition:** CIM UML ID généré pour chaque objet CIM par un outil UML.
- **Destination value format:** même que l'IEC 62656-1
- **Original CIM definition:** N/A ("Non disponible")
- **Requirement:** Mandatory ("Obligatoire")
- **Name of source CIM object:** CIM UML ID

- **Requirement for mapping:** L'UML ID n'est pas disponible dans les normes CIM, mais est généré par un outil UML comme l'identificateur pour chaque classe. Le POM doit utiliser cet attribut pour décrire l'UML ID généré pour chaque classe du CIM.

6.2.14 CIM package ("Paquetage du CIM")

- **Destination ID:** MDC_P211
- **Destination Name:** CIM package ("paquetage du CIM")
- **Destination definition:** le nom de paquetage qui contient une classe
- **Destination value format:** même que l'IEC 62656-1
- **Original CIM definition:** disponible dans l'IEC 61970-301:2013
- **Requirement:** Mandatory ("Obligatoire")
- **Name of source CIM object:** nom de paquetage
- **Requirement for mapping:** Le CIM est divisé en un ensemble de paquetages. Un paquetage est un moyen polyvalent permettant de grouper les éléments de modèle liés et est défini en fonction du domaine d'application et de son but. Chaque objet dans le CIM, excepté "package", appartient à un certain "package". L'attribut "Segment" du POM, auquel est donné un nom synonyme "CIM package" dans la présente norme, est appliqué pour décrire les informations du paquetage du CIM. Un ICID d'un "package" est requis pour cet attribut.

6.2.15 Applicable properties ("Propriétés applicables")

- **Destination ID:** MDC_P014
- **Destination name:** Applicable properties ("Propriétés applicables")
- **Destination definition:** même que l'IEC 62656-1
- **Destination value format:** même que l'IEC 62656-1
- **Original CIM definition:** disponible dans l'IEC 61970-301:2013
- **Requirement:** Optional ("Facultatif")
- **Name of source CIM object:** N/A ("Non disponible")
- **Requirement for mapping:** Le CIM d'origine a une liste de propriétés définies pour chaque classe et toutes les propriétés définies dans une classe peuvent être appliquées pour décrire des instances de la classe. Dans le POM, les propriétés sont également définies dans une classe. Cependant, seules les propriétés qui sont dans l'état "applicable properties" ("propriétés applicables") peuvent être appliquées pour enregistrer les instances de la classe. Autrement, une propriété est dite visible, mais non applicable. Par conséquent, il est nécessaire que les ICID des propriétés CIM qui sont applicables soient ajoutés à l'attribut "Applicable properties" ("propriétés applicables") de chaque classe.

6.3 Property ("Propriété")

6.3.1 Généralités

Une propriété du POM, qui est utilisée pour définir un attribut d'une classe du CIM et une extrémité d'association dans la présente norme doit avoir les attributs énumérés dans le Tableau B.2. Le paragraphe 6.3 approfondit un certain nombre d'attributs sélectionnés pour une propriété dans le CIM paqueté en plus de ceux qui sont communs.

6.3.2 Property data element type ("Type d'élément de données de propriété")

- **Destination ID:** MDC_P020
- **Destination name:** Property data element type ("Type d'élément de données de propriété")
- **Destination definition:** même que l'IEC 62656-1
- **Destination value format:** même que l'IEC 62656-1
- **Original CIM definition:** non disponible

- **Requirement:** Mandatory ("Obligatoire")
- **Name of source CIM object:** N/A ("Non disponible")
- **Requirement for mapping:** Cet attribut dans le POM sert à définir le type de la propriété qui décide si une propriété dépend d'autres propriétés et conditions ou non. Une propriété issue de l'extrémité d'association doit être assignée avec la valeur comme "INVERSIBLE_P_DET". Pour les propriétés des autres modèles CIM, les valeurs doivent être assignées comme "NON_DEPENDENT_P_DET".

6.3.3 Definition class ("Classe de définition")

- **Destination ID:** MDC_P021
- **Destination name:** Definition class ("Classe de définition")
- **Destination definition:** même que l'IEC 62656-1
- **Destination value format:** même que l'IEC 62656-1
- **Original CIM definition:** disponible dans l'IEC 61970-301:2013
- **Requirement:** Mandatory ("Obligatoire")
- **Name of source CIM object:** attribut natif
- **Requirement for mapping:** Le CIM d'origine définit une propriété comme étant une "native property" ("propriété native") ou une "inherited property" ("propriété héritée") pour distinguer l'endroit où elle est définie. Le POM fournit cet attribut pour enregistrer la classe dans laquelle la propriété est définie. Étant donné que chaque propriété du POM requiert l'ICID d'une classe pour cet attribut, au cours de la transformation du CIM vers le POM, une mise en correspondance entre un nom de classe et son ICID est requise.

6.3.4 Data type ("Type de données")

- **Destination ID:** MDC_P022
- **Destination name:** Type de données
- **Destination definition:** même que l'IEC 62656-1
- **Destination value format:** même que l'IEC 62656-1
- **Original CIM definition:** disponible dans l'IEC 61970-301:2013
- **Requirement:** Mandatory ("Obligatoire")
- **Name of source CIM object:** Data type (Type de données)
- **Requirement for mapping:** La plupart des propriétés CIM requièrent un data type nommé lorsqu'elles sont représentées en data types du POM (se référer au paragraphe 6.4). Pour transformer du CIM vers le POM, le nom de data type dans le CIM doit être transformé en cet attribut.

6.3.5 Applicable relations ("Relations applicables")

- **Destination ID:** MDC_P230
- **Destination name:** Applicable relations (Relations applicables)
- **Destination definition:** même que l'IEC 62656-1
- **Destination value format:** M..0
- **Original CIM definition:** disponible dans l'IEC 61970-301:2013
- **Requirement:** Mandatory ("Obligatoire")
- **Name of source CIM object:** N/A ("Non disponible")
- **Requirement for mapping:** Dans le cas d'une propriété issue d'une extrémité d'association, l'ID de relation correspondant à l'association qui la possède doit être décrit dans cet attribut.

6.3.6 Condition

- **Destination ID:** MDC_P028

- **Destination name:** Condition
- **Destination definition:** même que l'IEC 62656-1
- **Destination value format:** M..0
- **Original CIM definition:** disponible dans l'IEC 61970-301:2013
- **Requirement:** Optional ("Facultatif")
- **Name of source CIM object:** N/A ("Non disponible")
- **Requirement for mapping:** Dans le cas d'une propriété issue d'une extrémité d'association d'une association bidirectionnelle, l'ID de propriété représentant l'extrémité d'association opposée doit être décrit dans cet attribut.

6.3.7 Unit in text ("Unité en texte")

- **Destination ID:** MDC_P023_1
- **Destination name:** Unit in text ("Unité en texte")
- **Destination definition:** même que l'IEC 62656-1
- **Destination value format:** même que l'IEC 62656-1
- **Original CIM definition:** disponible dans l'IEC 61970-301:2013
- **Requirement:** Optional ("Facultatif")
- **Name of source CIM object:** Name of unit (Nom de l'unité)
- **Requirement for mapping:** Le CIM définit une unité (ou un ensemble d'unités) de mesure pour chaque data type numérique chaque fois que cela est nécessaire. L'unité pour une propriété doit être en conformité avec l'unité définie par le data type (MDC_P022) de cette propriété, c'est-à-dire la valeur de cet attribut doit être la même que celle qui est donnée au paragraphe 6.4.3.

6.3.8 Alternative units ("Unités alternatives")

- **Destination ID:** MDC_P111
- **Destination name:** Alternative units (Unités alternatives)
- **Destination definition:** même que l'IEC 62656-1
- **Destination value format:** même que l'IEC 62656-1
- **Original CIM definitions:** Dans le CIM il existe des définitions des préfixes possibles pour une unité.
- **Requirement:** Optional ("Facultatif")
- **Name of source CIM object:** UnitMultiplier
- **Requirement for mapping:** Cet attribut est utilisé pour décrire les unités alternatives disponibles d'une unité de mesure. Étant donné que le CIM définit unitMultiplier pour décrire les préfixes possibles pour l'unité de mesure, cet attribut doit être utilisé pour enregistrer les unités alternatives disponibles contraintes par le préfixe.

6.4 Data type ("Type de données")

6.4.1 Généralités

Un data type qui est utilisé pour définir un data type du CIM, excepté ceux de base, doit avoir les attributs énumérés dans le Tableau B.3. Le paragraphe 6.4 approfondit un certain nombre des attributs particuliers en plus de ceux qui sont communs.

6.4.2 Definition class ("Classe de définition")

- **Destination ID:** MDC_P021
- **Destination name:** Definition class («Classe de définition»)
- **Destination definition:** même que l'IEC 62656-1
- **Destination value format:** même que l'IEC 62656-1

- **Original CIM definition:** disponible dans l'IEC 61970-301:2013
- **Requirement:** Mandatory ("Obligatoire")
- **Name of source CIM object:** native attribute (attribut natif)
- **Requirement for mapping:** Un datatype dans le CIM est possédé par un paquetage, pas par une classe. Par conséquent, cet attribut du datatype doit être maintenu nul dans le POM pour chaque datatype du CIM.

6.4.3 Unit in text ("Unité en texte")

- **Destination ID:** MDC_P023_1
- **Destination name:** unit in texte (unité en texte)
- **Destination definition:** même que l'IEC 62656-1
- **Destination value format:** même que l'IEC 62656-1
- **Original CIM definition:** disponible dans l'IEC 61970-301:2013
- **Requirement:** Optional ("Facultatif")
- **Name of source CIM object:** name of unit (nom de l'unité)
- **Requirement for mapping:** Le CIM définit une unité SI à l'intérieur du data type partout où cela est nécessaire. L'unité doit être transformée en l'attribut "unit" du datatype dans le POM.

6.4.4 Alternative units ("Unités alternatives")

- **Destination ID:** MDC_P111
- **Destination name:** alternative units (Unités alternatives)
- **Destination definition:** même que l'IEC 62656-1
- **Destination value format:** même que l'IEC 62656-1
- **Original CIM definitions:** Dans le CIM, il existe des définitions des préfixes possibles pour une unité.
- **Requirement:** Optional (Facultatif)
- **Name of source CIM object:** UnitMultiplier
- **Requirement for mapping:** Cet attribut est utilisé pour décrire les unités alternatives disponibles d'une unité de mesure. Étant donné que le CIM définit unitMultiplier pour décrire les préfixes possibles pour l'unité de mesure, cet attribut doit être utilisé pour enregistrer les unités alternatives disponibles contraintes par le préfixe.

6.4.5 Data type ("Type de données")

- **Destination ID:** MDC_P022
- **Destination name:** Data type (Type de données)
- **Destination definition:** même que l'IEC 62656-1
- **Destination value format:** même que l'IEC 62656-1
- **Original CIM definition:** disponible dans l'IEC 61970-301:2013
- **Requirement:** Mandatory (Obligatoire)
- **Name of source CIM object:** N/D (Non disponible)
- **Requirement for mapping:** Cet attribut est utilisé pour les data types simples du POM correspondant aux data types de base du CIM énumérés dans le Tableau 10.

6.5 Enumeration ("Énumération")

6.5.1 Généralités

Une énumération qui est utilisée pour décrire la classe "enumeration" du CIM doit avoir les attributs énumérés dans le Tableau B.4. Le paragraphe 6.5 approfondit un certain nombre des attributs particuliers pour "enumeration" ("énumération").

6.5.2 Definition class ("Classe de définition")

- **Destination ID:** MDC_P021
- **Destination name:** Definition class ("Classe de définition")
- **Destination definition:** même que l'IEC 62656-1
- **Destination value format:** même que l'IEC 62656-1
- **Original CIM definition:** non disponible
- **Requirement:** Mandatory ("Obligatoire")
- **Name of source CIM object:**
- **Requirement for mapping:** "Enumeration" dans le CIM est défini comme une classe "enumeration". Elle n'est possédée par aucune classe. Cependant, le POM assigne la "Definition class" ("classe de définition") à chaque énumération. Cet attribut étant obligatoire dans le POM, il doit être ajouté à l'Énumération du CIM et la valeur de cet attribut peut être mise à zéro dans le POM.

6.5.3 Enumerated list of terms ("Liste énumérée de termes")

- **Destination ID:** MDC_P043
- **Destination name:** Enumerated list of terms ("Liste énumérée de termes")
- **Destination definition:** même que l'IEC 62656-1
- **Destination value format:** même que l'IEC 62656-1
- **Original CIM definition:** non disponible
- **Requirement:** Mandatory ("Obligatoire")
- **Name of source CIM object:** N/A ("Non disponible")
- **Requirement for mapping:** Les valeurs possibles sont énumérées dans cet attribut pour chaque énumération dans le POM. Au cours de la transformation du CIM vers le POM, toutes les valeurs possibles de chaque énumération doivent être ajoutées à la liste.

6.5.4 Enumerated code list ("Liste de codes énumérés")

- **Destination ID:** MDC_P044
- **Destination name:** Enumerated code list ("Liste de codes énumérés")
- **Destination definition:** même que l'IEC 62656-1
- **Destination value format:** même que l'IEC 62656-1
- **Original CIM definition:** non disponible
- **Requirement:** Mandatory ("Obligatoire")
- **Name of source CIM object:** N/A ("Non disponible")
- **Requirement for mapping:** Les valeurs possibles représentées dans des codes sont énumérées dans cet attribut pour chaque énumération dans le POM. La liste de cet attribut doit correspondre à la liste de MDC_P043. Au cours de la transformation du CIM vers le POM, les codes de toutes les valeurs possibles de chaque énumération doivent être ajoutés à la liste.

6.6 Métaclasse "Term" ("Terme")

6.6.1 Généralités

Les termes qui sont utilisés pour décrire les attributs de chaque énumération du CIM doivent avoir les attributs énumérés dans le Tableau B.5. Le paragraphe 6.6 approfondit l'un des attributs particuliers pour "term".

6.6.2 Preferred letter symbol in text ("Symbole littéral préférentiel en texte")

- **Destination ID:** MDC_P025_1
- **Destination name:** Preferred letter symbol in text ("Symbole littéral préférentiel en texte")

- **Destination definition:** même que l'IEC 62656-1
- **Destination value format:** même que l'IEC 62656-1
- **Original CIM definition:** non disponible
- **Requirement:** Optional ("Facultatif")
- **Name of source CIM object:** enumeration.enumerator
- **Requirement for mapping:** Le POM fournit cet attribut pour décrire le "Short name" ("nom court") de chaque terme. Un symbole en lettres peut être utilisé pour représenter un terme partout où cela est nécessaire. Cet attribut doit être ajouté à chaque terme dans le CIM.

6.7 Relation

6.7.1 Généralités

Dans la présente norme, l'objet "Relation" du POM est utilisé pour décrire trois types des modèles CIM; (1) un paquetage du CIM, (2) les collections des éléments d'un paquetage, et (3) l'association du CIM. L'Annexe G donne une explication détaillée des utilisations de la relation.

Pour décrire les trois types ci-dessus des modèles CIM, la relation du POM doit avoir les attributs énumérés dans le Tableau B.6. Le paragraphe 6.7 approfondit un certain nombre des attributs particuliers.

6.7.2 Relation type ("Type de relation")

- **Destination ID:** MDC_P200
- **Destination name:** Relation type ("Type de relation")
- **Destination definition:** même que l'IEC 62656-1
- **Destination value format:** même que l'IEC 62656-1
- **Original CIM definition:** non disponible
- **Requirement:** Mandatory ("Obligatoire")
- **Name of source CIM object:** N/A ("Non disponible")
- **Requirement for mapping:** Un paquetage dans le CIM doit être transformé en une relation du POM avec un type de relation de "Function". Pour les autres modèles CIM y compris l'association, le type de relation doit être mis à "Predication".

6.7.3 Domain of the relation ("Domaine de la relation")

- **Destination ID:** MDC_P201
- **Destination name:** Domain of the relation ("Domaine de la relation")
- **Destination definition:** même que l'IEC 62656-1
- **Destination value format:** même que l'IEC 62656-1
- **Original CIM definition:** non disponible
- **Requirement:** Optional ("Facultatif")
- **Name of source CIM object:** N/A ("Non disponible")
- **Requirement for mapping:** Il existe deux types de données disponibles dans cet attribut de la présente norme. Un type de données est utilisé pour les collections d'éléments d'un paquetage du CIM. Dans le CIM, chaque paquetage contient un groupe d'objets tels que des classes du CIM et des attributs du CIM. Dans le POM, les objets conteneurs d'un paquetage doivent être définis comme instances d'une relation selon le type d'objet. Cet attribut est utilisé pour décrire la liste des collections d'éléments conteneurs. Pour l'exemple donné dans le Tableau 13, toutes les classes contenues par le paquetage "Generation" doivent être définies comme une instance de relation (par exemple; cls_re1) qui doit donner la valeur de cet attribut aux listes de classes-conteneurs, la valeur de

“Domain element type” à “MDC_C002” et la valeur de “Relation type(MDC_P200)” à “Predication”. Pour d’autres types d’objets CIM contenus par un paquetage, la même règle est appliquée.

L’autre type de données est utilisé pour l’association du CIM. Cet attribut doit être utilisé pour décrire la liste des propriétés qui représentent les extrémités d’association contenues par l’association. Le Tableau 5 en donne un exemple.

6.7.4 Domain of the function ("Domaine de la fonction")

- **Destination ID:** MDC_P202
- **Destination name:** Domain of the function ("Domaine de la fonction")
- **Destination definition:** même que l'IEC 62656-1
- **Destination value format:** même que l'IEC 62656-1
- **Original CIM definition:** non disponible
- **Requirement:** Optional ("Facultatif")
- **Name of source CIM object:** N/A ("Non disponible")
- **Requirement for mapping:** Dans le CIM, chaque paquetage contient un groupe d’objets tels que des classes du CIM et des attributs du CIM. Dans le POM, les objets conteneurs d’un paquetage doivent être définis comme des instances d’une Relation selon le type d’objet (voir l’exemple décrit en 6.7.3). Tous les objets conteneurs doivent être décrits dans cet attribut en mettant la valeur de “RelationType(MDC_P200)” à “function”.

6.7.5 Codomain of the function ("Co-domaine de la fonction")

- **Destination ID:** MDC_P203
- **Destination name:** Codomain of the function ("Co-domaine de la fonction")
- **Destination definition:** même que l'IEC 62656-1
- **Destination value format:** même que l'IEC 62656-1
- **Original CIM definition:** non disponible
- **Requirement:** Optional ("Facultatif")
- **Name of source CIM object:** N/A ("Non disponible")
- **Requirement for mapping:** Dans le CIM, il existe des relations d’inclusion entre les paquetages. L’ICID du paquetage conteneur doit être décrit dans cet attribut d’un paquetage. Par exemple, dans l’IEC 61970-301:2013, le paquetage “Generation” contient les paquetages “Production” et “GenerationDynamics”. Pour la définition des paquetages “Production” et “GenerationDynamics”, l’ICID du paquetage “Generation” doit être décrit dans cet attribut. Voir l’exemple dans le Tableau 13.

6.7.6 Domain element type ("Type d’élément de domaine")

- **Destination ID:** MDC_P208
- **Destination name:** Domain element type ("Type d’élément de domaine")
- **Destination definition:** même que l'IEC 62656-1
- **Destination value format:** même que l'IEC 62656-1
- **Original CIM definition:** non disponible
- **Requirement:** Optional ("Facultatif")
- **Name of source CIM object:** N/A ("Non disponible")
- **Requirement for mapping:** Cet attribut est utilisé pour définir le type d’objet décrit dans l’attribut “Domain of the relation” lorsque cela est possible. La valeur disponible pour cet attribut est le nom de la métapropriété dans le POM. Par exemple: “MDC_C002” représente la collection de classes, “MDC_C003” représente la collection de propriétés, “MDC_C006” représente la collection des data types.

6.7.7 Role of the relation ("Rôle de la relation")

- **Destination ID:** MDC_P210
- **Destination name:** Role of the relation ("Rôle de la relation")
- **Destination definition:** même que l'IEC 62656-1
- **Destination value format:** même que l'IEC 62656-1
- **Original CIM definition:** non disponible
- **Requirement:** Optional ("Facultatif")
- **Name of source CIM object:** N/A ("Non disponible")
- **Requirement for mapping:** Cet attribut est utilisé pour définir le rôle de la relation. Lorsqu'une relation est issue d'un paquetage du CIM, la valeur de cet attribut doit être mise à "CIM package" (paquetage du CIM). Lorsqu'une relation est issue d'une association du CIM, la valeur de cet attribut doit être mise à "CIM association" ("Association du CIM"). Pour les autres relations disponibles dans la présente norme, telles qu'une relation pour une collection de classes, la valeur de cet attribut doit être mise à zéro. Cet attribut peut être utilisé pour distinguer une définition "CIM package", "CIM association" des autres.

7 Schéma XML pour des instances du CIM mises à jour dans le CDD de l'IEC

7.1 Généralités

Un schéma CIM RDF est défini dans l'IEC 61970-501:2006 et correspond au modèle conceptuel du CIM en UML défini dans l'IEC 61970-301:2013, afin de fournir un format d'échange commun et un ensemble d'interfaces d'accès données aux métadonnées relatives aux réseaux d'énergie électrique. Un fichier conforme au schéma est conçu pour être lisible par l'homme et par une machine et il peut être plus facilement étendu que le modèle CIM défini en UML.

Pour un fichier CIM qui est transformé dans le POM, c'est-à-dire, le CIM paqueté, les mises à jour de ses instances qui ne nécessitent pas un changement ou une extension du schéma CIM peuvent être accomplies conformément aux procédures de maintenance pour les normes dans le format de base de données défini dans l'Annexe SL du supplément spécifique à l'IEC des Directives ISO/IEC. Les mises à jour du contenu de l'instance, modélisé à l'origine dans le CIM, doivent être traçables pour l'instance d'origine du CIM. Si aucune extension n'est apportée au schéma CIM, le schéma CIM RDF d'origine doit être utilisé, car il n'y a nul besoin d'enregistrer un quelconque changement ou ajout de schéma apporté par le POM. Cependant, si une extension du schéma CIM se produit, par exemple, les informations de la version et de la révision dans le POM sont jugées indispensables et nécessitent donc d'être exportées, le schéma CIM RDF d'origine a besoin d'être étendu d'une manière qui permette une démarcation claire des frontières entre le schéma CIM RDF d'origine et les extensions du modèle (vis-à-vis du CIM) apportées par le POM. Le détail de cette démarcation est plus approfondi au paragraphe 7.2.

7.2 Principes

Pendant la conception des règles de transformation du CIM paqueté vers un fichier CIM RDF, la description du schéma RDF existant dans l'IEC 61970-501:2006 est enregistrée pour être utilisée sans changement afin que toutes les applications éventuelles utilisant le CIM RDF fonctionnent sans prendre en considération le schéma RDF étendu dans le POM.

Attributes originating in CIM			Attributes extended by POM		
MDC_P004_1.en	CIM_P002	MDC_P007_1.en	MDC_1_6	MDC_P002_1	...
Name	CIM package	Note	Code	version	
Instances of CIM objects	Switch	Wires	A generic device designed to close, or open, or both, one or more electric circuits.	cim_cls_1	001
	Breaker	Wires	A mechanical switching device capable of making, carrying, and breaking currents under normal circuit conditions and also making...	cim_cls_2	001
	Jumper	Wires	A short section of conductor with negligible impedance which can be manually removed and replaced if the circuit is de-energized...	cim_cls_3	001
Extended instances in POM	Battery	N/A		AAA017	001
	Switchgear	N/A		ACC100	001
	...				

IEC

Légende

Anglais	Français
Attributes originating in CIM	Attributs provenant du CIM
Attributes extended by POM	Attributs étendus par le POM
Name	Nom
CIM package	Paquetage du CIM
Instances of CIM objects	Instances d'objets du CIM
Switch	Organe de coupure
Wires	Câbles
A generic device designed to close, or open, or both, one or more electric circuits	Un dispositif générique conçu pour fermer et/ou ouvrir un ou plusieurs circuits électriques
Breaker	Disjoncteur
A mechanical switching device capable of making, carrying, and breaking currents under normal circuit conditions and also making...	Un dispositif de commutation mécanique capable d'établir, de transporter et de couper des courants dans des conditions normales de circuit et également d'établir...
Jumper	Pont/shunt
A short section of conductor with negligible impedance which can be manually removed and replaced if the circuit is de-energized...	Courte section de conducteur avec une impédance négligeable qui peut être retirée et remplacée manuellement si le circuit est mis hors tension...
Extended instances in POM	Instances étendues dans le POM
Battery	Batterie
Switchgear	Appareillage de commutation
N/A	N/A (non disponible)

Figure 7 – Instances de classe du CIM paqueté

La Figure 7 donne un exemple d'instances de classe du CIM paqueté. Noter que le contenu indiqué dans cette figure est une partie intégrante des définitions formelles.

Comme le montrent les lignes de la Figure 7, les instances des objets du CIM paqueté peuvent être divisées en deux types selon les définitions de leur source;

- Les "instances d'objets du CIM" qui sont issues du CIM sont groupées comme étant du type 1,

- Les "instances étendues dans le POM" sont groupées comme étant du type2.

Pour décrire ces deux types d'instances, deux groupes d'attributs sont appliqués, comme le montrent les colonnes de la Figure 7 de manière spécifique;

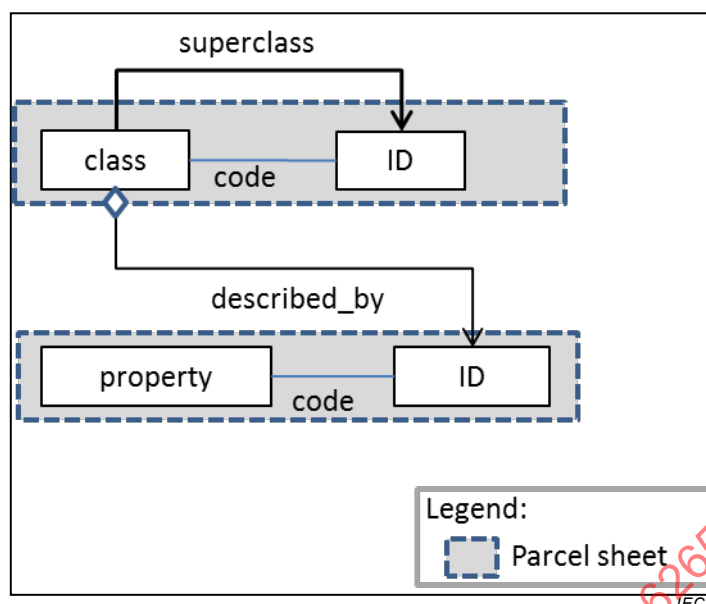
- Les "attributs provenant du CIM" correspondent aux attributs qui ont des définitions du CIM d'origine. Ainsi, le schéma RDF de l'IEC 61970-501:2006 peut être appliqué pour décrire leurs instances. Par exemple, à la Figure 7, les instances en couleur orange, c'est-à-dire les colonnes Name, CIM package, Note (respectivement Nom, Paquetage du CIM et Note) définies pour une classe du CIM, peuvent être décrites avec le schéma RDF de l'IEC 61970-501:2006.
- Les "attributs étendus par le POM" sont ceux qui ne sont disponibles que dans le POM. Pour l'exemple donné à la Figure 7, les colonnes Code, Version représentent des attributs ajoutés par le POM pour une classe. Comme ils n'ont pas de définitions dans le CIM d'origine, l'IEC 61970-501:2006 n'a donc pas de schémas RDF pour décrire leurs instances. Dans ce cas, un schéma XML étendu subsumant les définitions de l'IEC 61970-501:2006 est créé, selon les règles génériques d'extension du RDF.

L'Annexe C illustre des schémas XML en tant qu'informations à valeur informative, pour décrire ce dernier cas qui contient les deux types d'attributs du CIM paqueté. Dans l'Annexe C, un espace de noms supplémentaire: "pom" est fourni pour décrire les "attributs étendus par le POM" d'une instance. Par exemple, dans l'espace de nom "pom", les schémas XML pour Code, Version indiqués à la Figure 7 sont exprimés séparément comme étant <pom:code> et <pom:version>.

L'Annexe D donne des exemples de fichiers XML pour chaque sorte de modèle CIM paqueté en tant qu'informations à valeur informative. Dans ces exemples, les descriptions des "attributs provenant du CIM" viennent en premier, puis les "attributs étendus par le POM" suivent, marqués par l'étiquette <pom:ext>.

8 Mécanisme de contrôle des versions dans le POM pour le CIM paqueté

Dans le CIM paqueté, un objet est toujours identifié par un ID unique au niveau global et les références aux objets sont toujours faites par l'intermédiaire de l'ID. La Figure 8 décrit ce mécanisme de référence.



Légende

Anglais	Français
superclass	superclasse
class	classe
code	code
ID	Identificateur
described_by	décrit par
property	propriété
Legend	Légende
Parcel sheet	Feuille de paquet

Figure 8 – Mécanisme de référence conceptuel avec ID dans le POM

Dans l'IEC 62656-1, comme ID unique au niveau global, un ICID (ID de concept international (International Concept ID)), qui inclut les informations relatives à la version, est utilisé pour identifier chaque objet et les relations de référence entre les objets. Comme décrit dans l'IEC 62656-1, chaque ICID est unique à travers la frontière des organisations dans le format de paquet et il a la séquence suivante:

ICID::= RAI'##DI'##VI

Ici, RAI représente l'identificateur de l'autorité d'enregistrement, DI l'identificateur de données (data identifier) et VI l'identificateur de version (version identifier). Lorsqu'un objet est mis à jour, la version incluse dans l'ICID est mise à jour en conséquence. Avec les descriptions ci-dessus de la structure de l'ICID et des relations de référence, il devient clair de savoir si, oui ou non, un objet est mis à jour avec le numéro de version inclus dans l'ICID et, par conséquent, un contrôle des versions entité par entité peut être réalisé avec le CIM paqueté.

Annexe A (normative)

Règles de mise en correspondance du CIM vers le POM – Règles de mise en correspondance décrites dans la métaclasse "relation" du POM

Les règles de mise en correspondance entre les métamodèles UML disponibles dans le CIM et les métamodèles du POM peuvent être décrites en utilisant la métaclasse "relation" dans le POM. Les métamodèles UML disponibles dans le CIM qui exigent des règles de mise en correspondance sont:

- Class ("Classe")
- Class attribute ("Attribut de classe")
- Association
- Enumeration ("Énumération")
- Enumeration attribute ("Attribut d'énumération")
- Data type ("Type de données")
- Package ("Paquetage")

Les règles de mise en correspondance pour les métamodèles ci-dessus dans le CIM décrits comme relations dans le POM sont énumérées dans les tableaux (Tableaux A.1 à A.8). Dans ces tableaux, les métamodèles UML disponibles dans le CIM sont indiqués dans la colonne "Domain of the function" et les métamodèles POM correspondants sont énumérés dans la colonne "Codomain of the function".

En ce qui concerne les règles de mise en correspondance du métamodèle POM vers le métamodèle UML, outre les cas de relation de mise en correspondance 1:1, il y a les cas de mise en correspondance 1:n des métamodèles POM vers les métamodèles UML. Les règles de mise en correspondance de 1:n pour les métamodèles sont énumérées dans le Tableau A.9. Dans ce tableau, les métamodèles POM sont énumérés dans la colonne 'Domain of the function' et les possibles métamodèles UML sont énumérés dans la colonne "Codomain of the function".

Par exemple, chaque propriété du POM peut être mise en correspondance avec (1) un attribut de la classe du CIM, ou (2) une extrémité d'association du CIM en fonction du "Property data element type" ("type d'élément de données de propriété"). Lorsque la valeur du "Property data element type" est "INVERSIBLE_P_DET", la propriété du POM doit être mise en correspondance avec une extrémité d'association du CIM. En ce qui concerne l'autre propriété du POM avec "NON_DEPENDENT_P_DET", elle doit être mise en correspondance avec un attribut ordinaire de sa classe du CIM correspondante. En conséquence, les attributs de la propriété du POM doivent être mis en correspondance avec les différents métamodèles du CIM. Un autre exemple est la relation du POM. Chaque relation du POM peut être mise en correspondance avec (1) un CIM package (Paquetage du CIM), (2) une CIM association (association du CIM), etc. Selon les règles de mise en correspondance des Tableaux A.1 à A.9, elle est disponible pour transformer le métamodèle CIM paqueté en métamodèle UML.

Tableau A.1 – Règle de mise en correspondance d'une classe du CIM vers une classe du POM

MDC_P001_13	MDC_P004_1.<lang>	MDC_P005.<lang>	MDC_P200	MDC_P202	MDC_P203	MDC_P007_1.<lang>
Code	Preferred name ("Nom préférentiel")	definition ("définition")	Relation type ("Type de relation")	Domain of the function ("Domaine de la fonction")	Codomain of the function ("Co- domaine de la fonction")	Note
cls_re_001	Name mapping for classes ("Mise en correspondance de nom pour des classes")	Mise en correspondance entre le nom d'une classe du CIM et le nom préférentiel d'une classe du POM	Function ("Fonction")	cim.class.name	MDC_P004_3.<lang>	Le nom court d'une classe du POM est désigné par MDC_P004_3.<Lang>
cls_re_002	Description mapping for classes ("Mise en correspondance de description pour des classes")	Mise en correspondance entre la description d'une classe du CIM et la note d'une classe du POM	Function ("Fonction")	cim.class.description	MDC_P007_1.<lang>	La note d'une classe du POM est désignée par MDC_P007_1.<Lang>
cls_re_003	Mapping for generalization ("Mise en correspondance pour généralisation")	Mise en correspondance entre la relation de généralisation des classes du CIM et la superclasse d'une classe du POM	Function ("Fonction")	cim.class.superclass	MDC_P010	La superclasse d'une classe du POM est désignée par MDC_P010

Tableau A.2 – Règle de mise en correspondance d'un attribut de classe du CIM vers une propriété du POM

MDC_P001_13	MDC_P004_1.<lang>	MDC_P005.<lang>	MDC_P200	MDC_P202	MDC_P203	MDC_P007_1.<lang>
Code	Preferred name ("Nom préférentiel")	definition ("définition")	Relation type ("Type de relation")	Domain of the function ("Domaine de la fonction")	Codomain of the function ("Co-domaine de la fonction")	Note
cls_attr_re_001	Name mapping for CIM class attributes ("Mise en correspondance de nom pour les attributs de classe du CIM")	Mise en correspondance entre le nom d'un attribut de classe du CIM et le nom préférentiel d'une propriété du POM	Function ("Fonction")	cim.classAttribute.name	MDC_P004_3.<lang>	Le nom court d'une propriété du POM est désigné par MDC_P004_3.<Lang>
cls_attr_re_002	Mapping for CIM class attribute type ("Mise en correspondance pour le type d'attribut de classe du CIM")	Mise en correspondance entre le type d'un attribut de classe du CIM et le data type d'une propriété du POM	Function ("Fonction")	cim.classAttribute.type	MDC_P022	Le data type d'une propriété du POM est désigné par MDC_P022.
cls_attr_re_003	Description mapping for CIM class attribute ("Mise en correspondance de description pour l'attribut de classe du CIM")	Mise en correspondance entre la description d'un attribut de classe du CIM et la note (MDC_P007_1.<lang>) d'une propriété du POM.	Function ("Fonction")	cim.classAttribute.description	MDC_P007_1.<lang>	La note d'une propriété du POM est désignée par MDC_P007_1.<Lang>
cls_attr_re_004	mapping for CIM class ownership ("Mise en correspondance pour la possession de classe du CIM")	Mise en correspondance entre la possession d'un attribut du CIM et la classe "definition" d'une propriété du POM	Function ("Fonction")	cim.classAttribute.owner ship	MDC_P021	La classe "definition" d'une propriété du POM est désignée par MDC_P021.

Tableau A.3 – Règle de mise en correspondance d'une association du CIM vers une relation du POM

MDC_P001_13	MDC_P004_1.<lang>	MDC_P005.<lang>	MDC_P200	MDC_P202	MDC_P203	MDC_P007_1.<lang>
Code	Preferred name ("Nom préférentiel")	definition ("définition")	Relation type ("Type de relation")	Domain of the function ("Domaine de la fonction")	Codomain of the function ("Co-domaine de la fonction")	Note
association_re_001	Mapping for CIM association end name ("Mise en correspondance pour un nom d'association du CIM")	Mise en correspondance entre le nom d'une association du CIM et le nom préférentiel d'une relation du POM	Function ("Fonction")	cim.association.name	MDC_P004_3.en	Le preferred name ("nom préférentiel") d'une relation du POM est utilisé pour décrire le nom d'une association
association_re_002	Mapping for association end members (Mise en correspondance pour les membres d'extrémité d'association) et "Domain of the relation" ("Domaine de la relation")	Mise en correspondance entre le membre d'extrémité d'association possédé par une association et le "Domain of the relation" d'une relation du POM	Function ("Fonction")	cim.association.associationE nds	MDC_P201	La liste de la propriété représentant les extrémités d'association d'une association doit être décrite dans le "Domain of the relation" d'une relation du POM.

Tableau A.4 – Règle de mise en correspondance d'une extrémité d'association du CIM vers une propriété du POM

MDC_P001_13	MDC_P004_1.<lang>	MDC_P005.<lang>	MDC_P200	MDC_P202	MDC_P203	MDC_P007_1.<lang>
Code	Preferred name ("Nom préférentiel")	definition ("définition")	Relation type ("Type de relation")	Domain of the function ("Domaine de la fonction")	Codomain of the function ("Co- domaine de la fonction")	Note
associationend_re_001	Mapping for CIM associationend name ("Mise en correspondance pour nom d'extrémité d'association du CIM")	Mise en correspondance entre le nom d'une extrémité d'association du CIM et le nom court d'une propriété du POM	Function ("Fonction")	cim.associationend.name	MDC_P004_3.en	Le nom court d'une propriété du POM est désigné par MDC_P004_3.<Lang>
associationend_re_002	Mapping for CIM associationend description ("Mise en correspondance pour description d'extrémité d'association du CIM")	Mise en correspondance entre la description d'une note d'association du CIM et la note d'une propriété du POM	Function ("Fonction")	cim.associationend.description	MDC_P007_1.en	La note d'une propriété du POM est désignée par MDC_P007_1.<Lang>
associationend_re_003	Mapping for CIM associationend ownership ("Mise en correspondance pour la possession d'extrémité d'association du CIM")	Mise en correspondance entre la classe où l'extrémité d'association du CIM est définie et la classe "definition" d'une propriété du POM	Function ("Fonction")	cim.associationend.ownership	MDC_P021	La classe "definition" d'une propriété du POM est désignée par MDC_P021.
associationend_re_004	Mapping for CIM aggregationKind ("Mise en correspondance pour aggregationKind du CIM")	Mise en correspondance entre l'aggregationKind d'une extrémité d'association du CIM et la Remark ("Remarque") d'une propriété du POM	Function ("Fonction")	cim.associationend.aggregationKind	MDC_P007_2.en	L'aggregationKind du CIM d'une propriété du POM est décrit dans la Remark ("Remarque") désignée par MDC_P007_2.en

MDC_P001_13	MDC_P004_1.<lang>	MDC_P005.<lang> definition ("définition")	MDC_P200 Relation type ("Type de relation")	MDC_P202 Domain of the function ("Domaine de la fonction")	MDC_P203 Codomain of the function ("Co-domaine de la fonction")	MDC_P007_1.<lang>
Code	Preferred name ("Nom préférentiel")					Note
associationend_re_005	Mapping for the opposite associationend of a bidirectional association (Mise en correspondance pour l'extrémité d'association opposée d'une association bidirectionnelle)	Mise en correspondance entre l'extrémité d'association opposée d'une association bidirectionnelle et la Condition de la propriété du POM représentant l'extrémité d'association	Function ("Fonction")	cim.associationend.opposite .end	MDC_P028	L'extrémité d'association opposée d'une association bidirectionnelle doit être décrite dans l'attribut MDC_P028(Remark) de la propriété du POM.
associationend_re_006	Mapping for the class to which an associationend is connected (Mise en correspondance pour la classe à laquelle est liée une extrémité d'association)	Mise en correspondance entre la classe à laquelle est liée une extrémité d'association du CIM et la classe définie dans class_reference_type du data type de propriété du POM	Function ("Fonction")	cim.associationend.targetCi ass	MDC_P022	La classe à laquelle une extrémité d'association est liée doit être décrite dans class_reference_type de l'attribut data type(MDC_P022) de la propriété du POM.

Tableau A.5 – Règle de mise en correspondance entre énumération du CIM et énumération du POM

MDC_P001_13	MDC_P004_1.<lang>	MDC_P005.<lang>	MDC_P200	MDC_P202	MDC_P203	MDC_P007_1.<lang>
Code	Preferred name ("Nom préférentiel")	definition ("définition")	Relation type ("Type de relation")	Domain of the function ("Domaine de la fonction")	Codomain of the function ("Co-domaine de la fonction")	Note
enum_re_001	Name mapping for CIM enumeration ("Mise en correspondance de nom pour une énumération du CIM")	Mise en correspondance entre le nom d'une énumération du CIM et le nom préférentiel d'une énumération du POM	Function ("Fonction")	cim.enumeration.name	MDC_P004_3.<lang>	Le nom court d'une énumération du POM est désigné par MDC_P004_3.<lang>
enum_re_002	Description mapping for CIM enumeration ("Mise en correspondance de description pour une énumération du CIM")	Mise en correspondance entre la description d'une énumération du CIM et la note d'une énumération du POM	Function ("Fonction")	cim.enumeration.description	MDC_P007_1.<lang>	La note d'une énumération du POM est désignée par MDC_P007_1.<Lang>

Tableau A.6 – Règle de mise en correspondance entre un attribut d'énumération du CIM et un terme du POM

MDC_P001_13	MDC_P004_1.<lang>	MDC_P005.<lang>	MDC_P200	MDC_P202	MDC_P203	MDC_P007_1.<lang>
Code	Preferred name ("Nom préférentiel")	definition ("définition")	Relation type ("Type de relation")	Domain of the function ("Domaine de la fonction")	Codomain of the function ("Co-domaine de la fonction")	Note
enum_attr_re_001	enumeration attribute name ("nom d'attribut d'énumération")	Mise en correspondance entre le nom d'un attribut d'énumération du CIM et le nom préférentiel d'un terme du POM	Function ("Fonction")	cim.enumeration.attribute.name	MDC_P004_3.<lang>	Le nom court d'un terme du POM est désigné par MDC_P004_3.<lang>
enum_attr_re_002	enumeration attribute description ("description d'attribut d'énumération")	Mise en correspondance entre la description d'une note d'attribut d'énumération du CIM et la note d'un terme du POM	Function ("Fonction")	cim.enumeration.attribute.description	MDC_P007_1.<lang>	La note d'un terme du POM est désignée par MDC_P007_1.<Lang>

Tableau A.7 – Règle de mise en correspondance entre un data type du CIM et un data type du POM

MDC_P001_13	MDC_P004_1.<lang>	MDC_P005.<lang>	MDC_P200	MDC_P202	MDC_P203	MDC_P007_1.<lang>
Code	Preferred name ("Nom préférentiel")	definition ("définition")	Relation type ("Type de relation")	Domain of the function ("Domaine de la fonction")	Codomain of the function ("Co-domaine de la fonction")	Note
datatype_re_001	Name mapping for CIM data type ("Mise en correspondance de nom pour data type du CIM")	Mise en correspondance entre le nom d'un data type du CIM et le nom préférentiel d'un data type du POM	Function ("Fonction")	cim.datatype.name	MDC_P004_3.<lang>	Le nom court d'un data type du POM est désigné par MDC_P004_3.<lang>
datatype_re_002	Description mapping for CIM data type ("Mise en correspondance de description pour data type du CIM")	Mise en correspondance entre la description d'un data type du CIM et le nom préférentiel d'un data type du POM	Function ("Fonction")	cim.datatype.description	MDC_P007_1.<lang>	La note d'un data type du POM est désignée par MDC_P007_1.<lang>
datatype_re_003	Mapping for CIM basic data type ("Mise en correspondance pour data type de base du CIM")	Mise en correspondance entre le data type de base du CIM et le data type simple du POM	Function ("Fonction")	cim.datatype.type	MDC_P022	La relation 1:1 entre le data type de base du CIM et le data type simple du POM est énumérée dans le Tableau 10.
datatype_re_005	Mapping for CIM unit ("Mise en correspondance pour unité du CIM")	Mise en correspondance entre l'unité ("unit") d'un data type du CIM et l'unité en texte ("unit in text") d'un data type du POM	Function ("Fonction")	cim.datatype.unit	MDC_P023_1	L'unité en texte ("unit in text") d'un data type du POM est désignée par MDC_P023_1
datatype_re_006	Mapping for CIM multiplier for unit ("Mise en correspondance pour multiplicateur CIM pour unité")	Mise en correspondance entre le multiplicateur ("multiplier") d'une unité du CIM et les unités alternatives d'une unité du POM	Function ("Fonction")	cim.datatype.multiplier	MDC_P111	Les préfixes variables pour l'unité CIM sont désignés par les unités alternatives (MDC_P111).

Tableau A.8 – Règle de mise en correspondance entre Paquetage du CIM et relation du POM

MDC_P001_13	MDC_P004_1.<lang> Preferred name ("Nom préférentiel")	MDC_P005.<lang> definition ("définition")	MDC_P200 Relation type ("Type de relation")	MDC_P202 Domain of the function ("Domaine de la fonction")	MDC_P203 Codomain of the function ("Co- domaine de la fonction")	MDC_P007_1.<lang> Note
package_re_001	Mapping for CIM Package name ("Mise en correspondance pour un nom de paquetage du CIM")	Mise en correspondance entre le nom de paquetage du CIM et l'attribut nom de relation du POM	Function ("Fonction")	cim.package.name	MDC_P004_3.<lang>	un nom de paquetage doit être mis en correspondance avec le nom court d'une relation dans le POM
package_re_002	Mapping for CIM package description ("Mise en correspondance pour une description du paquetage du CIM")	Mise en correspondance entre la description de paquetage du CIM et la note de relation du POM	Function ("Fonction")	cim.package.description	MDC_P007_1.<lang>	une description pour un paquetage du CIM doit être mise en correspondance avec une note de paquetage du POM.
package_re_003	Mapping for elements of the CIM package ("Mise en correspondance pour les éléments du paquetage du CIM")	Mise en correspondance entre la liste d'éléments et le "domain of the relation" du POM	Function ("Fonction")	cim.package.elements	MDC_P201	Les listes d'éléments doivent être décrites par le MDC_P201 dans le POM d'un paquetage
package_re_004	Mapping for CIM part- whole package ("Mise en correspondance pour un paquetage toutes les parties du CIM")	Mise en correspondance entre la relation toutes les parties du paquetage du CIM et le "codomain of the relation" de la relation du POM	Function ("Fonction")	cim.package.whole	MDC_P202	Le paquetage "tout" doit être énuméré dans l'attribut MDC_P202 de la relation du POM pour le paquetage "partie"
package_re_005	Mapping for CIM package version ("Mise en correspondance pour une version du paquetage du CIM")	Mise en correspondance entre la version du paquetage du CIM et la version de la relation du POM	Function ("Fonction")	cim.package.version	MDC_P002_1	La version pour un paquetage du CIM doit être mise en correspondance avec la version de la relation du POM
package_re_006	Mapping for CIM package date ("Mise en correspondance pour une date du paquetage du CIM")	Mise en correspondance entre la date du paquetage du CIM et la "date of current definition" de la relation du POM	Function ("Fonction")	cim.package.date	MDC_P003_2	La date du paquetage du CIM doit être mise en correspondance avec la date_of_current_definition de la relation du POM.

MDC_P001_13	MDC_P004_1.<lang>	MDC_P005.<lang>	MDC_P200	MDC_P202	MDC_P203	MDC_P007_1.<lang>
Code	Preferred name ("Nom préférentiel")	definition ("définition")	Relation type ("Type de relation")	Domain of the function ("Domaine de la fonction")	Codomain of the function ("Co-domaine de la fonction")	Note
Package_re_007	Mapping for CIM elements type ("Mise en correspondance pour un type d'élément du CIM")	Mise en correspondance entre le type d'élément du CIM, tel que "class", "property", "datatype", et le "domain element type" du POM	Function ("Fonction")	cim.package.element.type	MDC_P208	Les types possibles pour les éléments du CIM sont: (1)MDC_C002 pour les éléments "class" (2)MDC_C003 pour les éléments "property" (3)MDC_C006 pour les éléments "data type"

IECNORM.COM : Click to view the full PDF of IEC 62656-3:2015

Tableau A.9 – Règle de mise en correspondance de propriété du POM vers agrégation/association/attribut de classe du CIM

MDC_P001_13	MDC_P004_1.<lang>	MDC_P005.<lang>	MDC_P200	MDC_P202	MDC_P203	MDC_P007_1.<lang>
Code	Preferred name ("Nom préférentiel")	definition ("définition")	Relation type ("Type de relation")	Domain of the function ("Domaine de la fonction")	Codomain of the function ("Co-domaine de la fonction")	Note
Pom_re_1	Mapping for POM property short name ("Mise en correspondance pour le nom court de propriété du POM")	Mise en correspondance du "short name" d'une propriété du POM vers le nom d'une extrémité d'association du CIM ou du nom d'un attribut de classe	Function ("Fonction")	MDC_P004_3.en	cim.associationend.name cim.classAttribute.name	Le nom court d'une propriété du POM est désigné par MDC_P004_3.en
Pom_re_2	Mapping for POM property note ("Mise en correspondance pour note de propriété du POM")	Mise en correspondance de la note d'une propriété du POM vers la description d'une agrégation du CIM, d'une association du CIM ou d'un attribut de classe	Function ("Fonction")	MDC_P007_1.en	cim.aggregationend.descripti on cim.classAttribute. description	La note d'une propriété du POM est désignée par MDC_P007_1.<lang>
Pom_re_3	Mapping for POM property definition class ("Mise en correspondance pour une classe de définition de propriété du POM")	Mise en correspondance de la "definition class" (classe "définition") d'une propriété du POM vers la possession d'une agrégation du CIM, d'une association du CIM ou d'un attribut de classe	Function ("Fonction")	MDC_P021	cim.associationend.definedcl ass cim.classAttribute.definedclas s	La classe "définition" d'une propriété du POM est désignée par MDC_P021.
Pom_re_4	Mapping for POM relation name ("Mise en correspondance pour le nom de relation du POM")	Mise en correspondance de la relation du POM vers un Paquetage du CIM ou une association du CIM	Function ("Fonction")	MDC_P004_3.en	cim.package.name cim.association.name	Le nom court d'une relation du POM est désigné par MDC_P004_3.en
Pom_re_5	Mapping for POM Relation note ("Mise en correspondance pour note de relation du POM")	Mise en correspondance de la note d'une relation du POM vers la description d'un paquetage du CIM ou la description d'une association du CIM	Function ("Fonction")	MDC_P007_1.en	cim.package.description cim.association.description	La note d'une relation du POM est désignée par MDC_P007_1.<lang>