

INTERNATIONAL STANDARD

NORME INTERNATIONALE



**Energy management system application program interface (EMS-API) –
Part 401: Profile framework**

**Interface de programmation d'application pour système de gestion d'énergie
(EMS-API) –
Partie 401: Cadre de profils**



THIS PUBLICATION IS COPYRIGHT PROTECTED

Copyright © 2022 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 Secretariat
3, rue de Varembe
CH-1211 Geneva 20
Switzerland

Tel.: +41 22 919 02 11
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 corrigendum or an amendment might have been published.

IEC publications search - webstore.iec.ch/advsearchform

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 It also gives information on projects, replaced and withdrawn publications.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 once a month by email.

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: sales@iec.ch.

IEC Products & Services Portal - products.iec.ch

Discover our powerful search engine and read freely all the publications previews. With a subscription you will always have access to up to date content tailored to your needs.

Electropedia - www.electropedia.org

The world's leading online dictionary on electrotechnology, containing more than 22 300 terminological entries in English and French, with equivalent terms in 19 additional languages. Also known as the International Electrotechnical Vocabulary (IEV) online.

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.

Recherche de publications IEC -

webstore.iec.ch/advsearchform

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 une fois par mois par email.

Service Clients - webstore.iec.ch/csc

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

IEC Products & Services Portal - products.iec.ch

Découvrez notre puissant moteur de recherche et consultez gratuitement tous les aperçus des publications. Avec un abonnement, vous aurez toujours accès à un contenu à jour adapté à vos besoins.

Electropedia - www.electropedia.org

Le premier dictionnaire d'électrotechnologie en ligne au monde, avec plus de 22 300 articles terminologiques en anglais et en français, ainsi que les termes équivalents dans 19 langues additionnelles. Egalement appelé Vocabulaire Electrotechnique International (IEV) en ligne.

INTERNATIONAL STANDARD

NORME INTERNATIONALE



**Energy management system application program interface (EMS-API) –
Part 401: Profile framework**

**Interface de programmation d'application pour système de gestion d'énergie
(EMS-API) –
Partie 401: Cadre de profils**

INTERNATIONAL
ELECTROTECHNICAL
COMMISSION

COMMISSION
ELECTROTECHNIQUE
INTERNATIONALE

ICS 33.200

ISBN 978-2-8322-0093-3

**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.....	5
INTRODUCTION.....	7
1 Scope.....	8
2 Normative references	8
3 Terms, definitions and abbreviated terms	9
3.1 Terms and definitions.....	9
3.2 Abbreviated terms.....	10
4 Overview	11
4.1 Profiles and profiling	11
4.2 Relations between Canonical CIM, profiles and datasets	13
4.3 Profiles and business processes	14
5 Profile document structure.....	15
6 Profiling Use cases.....	15
6.1 Overview.....	15
6.2 Class in different profiles with no overlap.....	16
6.3 Include overlapping sets of attributes and roles from the same class in different profiles.....	17
6.4 Include a base class with different sets of attributes or roles.....	17
6.5 Extending an existing class.....	17
6.6 Extending an existing class used differently in different profiles	18
6.7 Different cardinalities	18
6.8 Add a new datatype	19
7 Profiling rules	19
7.1 Overview.....	19
7.2 R0101 Information model class, attribute, role and datatype names	19
7.3 Class rules for classes without stereotype.....	19
7.3.1 R0201 Including a class.....	19
7.3.2 R0202 Adding a new class.....	19
7.3.3 R0203 The description of a class	19
7.3.4 R0204 Name of a class.....	19
7.3.5 R0205 Cardinality of a class	19
7.3.6 R0206 Concrete class.....	20
7.4 Attribute rules	20
7.4.1 R0301 Including an attribute	20
7.4.2 R0302 Adding a new attribute.....	20
7.4.3 R0303 The description of an attribute	20
7.4.4 R0304 The name of an attribute.....	20
7.4.5 R0305 The cardinality of an attribute	20
7.4.6 R0306 The datatype of an attribute.....	20
7.4.7 R0307 Initial value of an attribute	20
7.5 Datatypes	21
7.5.1 R0401 Including a datatype	21
7.5.2 R0402 Adding a new datatype	21
7.5.3 R0403 Primitive datatype.....	21
7.5.4 R0405 CIMDatatype	21
7.5.5 R0405 Compound datatype	21

7.5.6	R0406 Enumeration	21
7.5.7	R0407 The description of a datatype	21
7.5.8	R0408 The name of a datatype	21
7.6	Association rules	22
7.6.1	R0501 Including an association	22
7.6.2	R0502 Adding a new association	22
7.6.3	R0503 The names of the two roles in an association	22
7.6.4	R0504 The cardinality of an association role	22
7.6.5	R0505 Association navigability	22
7.6.6	R0506 The description of an association	22
7.7	Attribute and association restrictions	22
7.7.1	R0601 Overlap between profiles	22
7.7.2	R0602 Same base class with different sets of attributes and associations in different profiles	23
7.8	R0701 Inheritance structure	23
7.9	R0801 Constraints	23
8	Extending Canonical CIM	24
9	Requirements for a profiling tool	24
9.1	Minimum requirements	24
9.2	Extended requirements for OCL rules	25
Annex A (informative)		26
A.1	Mapping of UML to OWL	26
A.2	Units and multipliers issue	26
A.2.1	Description of issue	26
A.2.2	Long term solution	30
A.2.3	Medium term solution	30
A.2.4	Temporary solution by flattening the profiles	31
A.2.5	Temporary solution by fattening the CIMXML data	31
A.2.6	Maintaining information for a flattened profile	32
Figure 1	– Relations between standards and profiling	12
Figure 2	– Relation between Canonical CIM, profiles and datasets	13
Figure 3	– Class in different profiles with no overlap	16
Figure 4	– Class in different profiles with overlap	17
Figure 5	– Same base class with different sets of attributes or roles	17
Figure 6	– Extending a class the same way in all applicable profiles	18
Figure 7	– Extending an existing class used differently in different profiles	18
Figure A.1	– CIMDatatype example in Canonical CIM and SSH profile	27
Figure A.2	– ActivePower from IEC 61970-452:2021 (CIM16)	28
Figure A.3	– Profile for ActivePower including CIMDatatype attributes	28
Figure A.4	– CIMXML example with an ActivePower instance as an identified node	29
Figure A.5	– CIMXML example with an ActivePower blank node serialisation example	29
Figure A.6	– Profile for ActivePower based on CIM version 9 (CIM9)	29

Figure A.7 – CIMXML example with an ActivePower instance according to the original serialisation	29
Figure A.8 – Example of Flattened profile for ActivePower	31
Table 1 – Abbreviated terms	11
Table A.1 – UML and ontology languages	26

IECNORM.COM : Click to view the full PDF of IEC 61970-401:2022

INTERNATIONAL ELECTROTECHNICAL COMMISSION

**ENERGY MANAGEMENT SYSTEM APPLICATION
PROGRAM INTERFACE (EMS-API) –****Part 401: Profile framework****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.

IEC 61970-401 has been prepared by IEC technical committee 57: Power systems management and associated information exchange. It is an International Standard.

This first edition cancels and replaces IEC TS IEC 61970-401 published in 2005. This edition constitutes a technical revision.

This edition includes the following significant technical changes with respect to the previous edition:

- a) The previous edition of IEC TS 61970-401:2005 provided an overview of the Component Interface Specifications (CIS) IEC 61970-402, IEC 61970-403, IEC 61970-404, IEC 61970-405, and IEC 61970-407. IEC 61970-402 to IEC 61970-407 are duplicates of existing OPC interfaces from OPC Foundation and the DAIS/HDA interfaces from OMG. Hence IEC 61970-402 to IEC 61970-407 have been withdrawn and IEC TS 61970-401:2005 no longer serves a purpose.

- b) IEC 61970-401 (this document) does not contain an overview of Component Interface Specifications (CIS) but instead a description of how to create profile specifications that describes dataset contents (or message contents). Hence it has been renamed "Profile framework". The profile specifications IEC 61970-450 (all parts) and IEC 61970-600 (all parts) describe dataset contents. The purpose of this document is to define the rules to be followed in the process of creating profile specifications.

The text of this International Standard is based on the following documents:

Draft	Report on voting
57/2482/FDIS	57/2494/RVD

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

The language used for the development of this International Standard is English.

This document was drafted in accordance with ISO/IEC Directives, Part 2, and developed in accordance with ISO/IEC Directives, Part 1 and ISO/IEC Directives, IEC Supplement, available at www.iec.ch/members_experts/refdocs. The main document types developed by IEC are described in greater detail at www.iec.ch/standardsdev/publications.

A list of all parts in the IEC 61970 series, published under the general title *Energy management system application program interface (EMS-API)*, can be found on the IEC website.

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

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

IMPORTANT – The "colour inside" logo on the cover page of this document 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.

INTRODUCTION

This document is one of the IEC 61970 series that defines message interfaces for network application data exchange.

The IEC 61970-300 series of documents specify a canonical Common Information Model (CIM) describing network application data. The CIM is an information model that represents all the major objects in an electric utility enterprise needed to describe data used by power network applications. The Canonical CIM provides the semantics for IEC 61970-450 (all parts) and IEC 61970-600 (all parts) profile specifications dedicated to specific data exchanges.

This document describes the framework in which profile specifications are created from the Canonical CIM. It describes the structure of profile documents and the rules for selection of information from the Canonical CIM to be included in profile specifications.

The reasons for creating this document are

- 1) The IEC 61970 profiles have for a long time been created using a profiling method not described by an IEC 61970 document.
- 2) The IEC 61970 profiling method has issues that need resolution. Issues and solutions are described in Annex A.

IECNORM.COM : Click to view the full PDF of IEC 61970-401:2022

ENERGY MANAGEMENT SYSTEM APPLICATION PROGRAM INTERFACE (EMS-API) –

Part 401: Profile framework

1 Scope

This document describes how IEC 61970-450 (all parts), IEC 61970-600 (all parts) profile specifications are structured and created. Profile specifications describe a subset of the Canonical CIM dedicated to a specific data exchange. The Canonical CIM is described in IEC 61970-300 (all parts) as well as in IEC 61968-11.

Rules for creation or extension of Canonical CIM are outside the scope of this document.

This document specifies the structure of a profile specification and the rules for selecting subsets of information from the Canonical CIM. It standardizes the operations used to create the profile elements from the Canonical CIM. As Canonical CIM is described in UML the operations are described in terms of UML classes, attributes, and roles.

It is possible to map UML to RDFS or OWL, so any of the languages UML, RDFS or OWL can be used to describe the created profiles. Specification of languages (UML, RDFS or OWL) used to describe profiles as well as how profiles are presented and edited in user interfaces are outside the scope of this document. Languages used to describe profiles are specified in other specifications. Relevant specifications are referenced in Clause 2.

UML supports adding free text that describes further restrictions on UML constructs, e.g. classes, attribute values, association roles and cardinalities. Languages such as OCL and SHACL are dedicated to describing constraints. OCL is used to describe constraints for object data described in UML while SHACL is used to describe constraints on graph data described by RDFS or OWL. OCL is within the scope of this document, but SHACL is not.

This document supports profiles describing data exchanged as CIMXML datasets or messages. The exchange format within the scope is in accordance with IEC 61970-552 but other formats are possible.

Tool interoperability and serialisation formats are outside the scope of this document.

2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes the requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

IEC 61968-11, *Application integration at electric utilities – System interfaces for distribution management – Part 11: Common information model (CIM) extensions for distribution*

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

IEC 61970-300 (all parts), *Energy management system application program interface (EMS-API)*

IEC 61970-501:2006, *Energy management system application program interface (EMS-API) – Part 501: Common Information Model Resource Description Framework (CIM RDF) schema*¹

IEC 61970-552, *Energy management system application program interface (EMS-API) – Part 552: CIMXML Model exchange format*

OMG Unified Modeling Language®, OMG document number: formal/2015-03-01, available at <http://www.omg.org/spec/UML/2.5>

World Wide Web Consortium (W3C), RDF 1.1 Primer from 24 June 2014, available at <https://www.w3.org/TR/rdf11-primer/>

World Wide Web Consortium (W3C), RDF 1.1 Concepts and Abstract Syntax from 25 February 2014, available at <https://www.w3.org/TR/rdf11-concepts/>

World Wide Web Consortium (W3C), RDF 1.1 XML Syntax from 25 February 2014, available at <https://www.w3.org/TR/rdf-syntax-grammar/>

World Wide Web Consortium (W3C), RDF Schema 1.1 from 25 February 2014, available at <https://www.w3.org/TR/rdf-schema/>

World Wide Web Consortium (W3C), OWL 2 Web Ontology Language Primer (Second Edition), W3C Recommendation 11 December 2012, available at <https://www.w3.org/TR/owl2-primer/>

World Wide Web Consortium (W3C), OWL 2 Web Ontology Language Structural Specification and Functional-Style Syntax (Second Edition), W3C Recommendation 11 December 2012, available at <https://www.w3.org/TR/owl2-syntax/>

3 Terms, definitions and abbreviated terms

For the purposes of this document, the terms and definitions given in IEC TS 61970-2 and the following apply.

ISO and IEC maintain terminological databases for use in standardization at the following addresses:

- IEC Electropedia: available at <http://www.electropedia.org/>
- ISO Online browsing platform: available at <http://www.iso.org/obp>

3.1 Terms and definitions

3.1.1

Canonical CIM

agreed semantic information model describing the power system domain; the information model aims at describing the power system domain in a normalized way without overlap

Note 1 to entry: The Canonical CIM is described in UML. UML can be serialized (saved as a document) in the standardized data format XMI for exchange between UML tools. Ecore is another serialization format with capability to describe UML.

¹ This specification is based on the W3C specification RDF Schema 1.0 from early 2000 which has since been revised multiple times.

3.1.2 profile

subset derived from Canonical CIM supporting the exchange of a specific set of data in a specific context

Note 1 to entry: A semantic information model in the same way as Canonical CIM, can be serialized in several different formats:

- XMI that is the serialisation format of UML as described by OMG, see clause 2;
- RDF/XML used to serialize RDFS or OWL;
- RDF/XML as described by IEC 61970-501. IEC 61970-501 contains several custom extensions to the RDFS specification from 1999;
- Ecore for exchange with Eclipse Modelling Framework (EMF).

Note 2 to entry: The serialisation of IEC 61970 profiles is standardised by IEC 61970-501. A collection of profiles supports a specific use case.

3.1.3 profiling

process of creating profiles

3.1.4 dataset

set of data described by a profile which means that the data in a dataset align with the information described by the profile

Note 1 to entry: A profile does not say anything about the data format.

3.1.5 data format

datasets carrying data described by profiles as defined in this document using CIMXML described in IEC 61970-552 and RDF/XML based on RDFS as described in IEC 61970-501

3.2 Abbreviated terms

The abbreviated terms are given in Table 1.

Table 1 – Abbreviated terms

Abbreviated term	Description
CIMXML	CIMXML is an RDF/XML data format and is described in the specification IEC 61970-552.
RDF/XML	A serialization format for RDF and graph data. For more details refer to the W3C specification RDF 1.1 XML Syntax.
RDF	RDF stands for Resource Description Framework and is an information model for graph data consisting of a: – Resource, in CIMXML this is an object identifier. – Predicate, in CIMXML this is an attribute or role name. – data item, in CIMXML this is a value or a resource reference. For an introduction to RDF refer to the W3C specification RDF 1.1 Primer.
RDFS	RDFS stands for RDF Schema and is an extension of RDF with capability to describe a vocabulary for data-modelling. Hence it is an information modelling language but with less capability than UML. For more details refer to the W3C specification RDF Schema 1.1.
OWL	OWL is an ontology language supporting the semantic web, it is a further development and extension of RDFS. For an introduction to OWL refer to the W3C recommendation OWL 2 Web Ontology Language Primer (Second Edition).
UML	UML stands for Unified Modelling Language. For more details refer to the OMG specification formal/2015-03-01.

4 Overview

4.1 Profiles and profiling

Profiling is the process of selecting information from the Canonical CIM and including it in profiles.

The Canonical CIM is described in IEC 61970-300 (all parts) and IEC 61968-11.

The guiding principle for the profiling method is that the information described by a profile is a true subset of the Canonical CIM and retains class, role names, attribute names and datatypes from the Canonical CIM. The datatypes in CIM are described by classes stereotyped Primitive, CIMDatatype, and Compound. CIMDatatype is a composition of three attributes: value, unit, and multiplier while Compound consists of an arbitrary set of attributes.

The main objective is that different datasets (see Clause 3) exchanged using different profiles based on Canonical CIM solely rely on the definitions and basic principles of the Canonical CIM. This is key to make interoperability feasible. This also enables different profiles to relate data between them by using the Canonical CIM as a hub and supports a reader of a dataset or a message to easily find descriptions of elements in both the profiles and the Canonical CIM. The support for relating data in different datasets or messages described by different profiles is needed when data is divided across different datasets governed by different profiles. Such use cases are defined for network models where the network description is separated from the operational conditions of the network (seen as an input) and the results.

There are several languages that can describe profiles, e.g. UML (serialized as XMI), RDFS, Ecore or OWL. UML includes a graphical language that is implemented by UML editors. OWL does not have a graphical language, but several editors exist that support the display and editing of OWL data.

Current profile specifications, IEC 61970-450 (all parts) and IEC 61970-600 (all parts) use UML to describe the information and OCL to further restrict the information. A profile in UML is described by classes, attributes, associations, roles and datatypes, the common way to describe information in UML. The UML language includes the concept of stereotypes and tagged values that enables custom extensions of the UML language. Hence profiling with UML means copying and updating classes, attributes, associations, stereotypes and tagged values from the Canonical CIM. A profile in RDFS or OWL is described by classes and properties but UML stereotypes and tagged values do not have a direct mapping to RDFS/OWL. Profiling using OWL mean creating OWL classes and properties by selecting UML classes, attributes, and roles from Canonical CIM the same way as for profiling with UML. UML stereotypes and tagged values need a specific mapping depending on the meaning of the stereotypes or tagged values. Clause A.1 presents a mapping of the Canonical CIM UML to OWL.

Figure 1 gives an overview of the IEC 61970 specifications in the context of profiling.

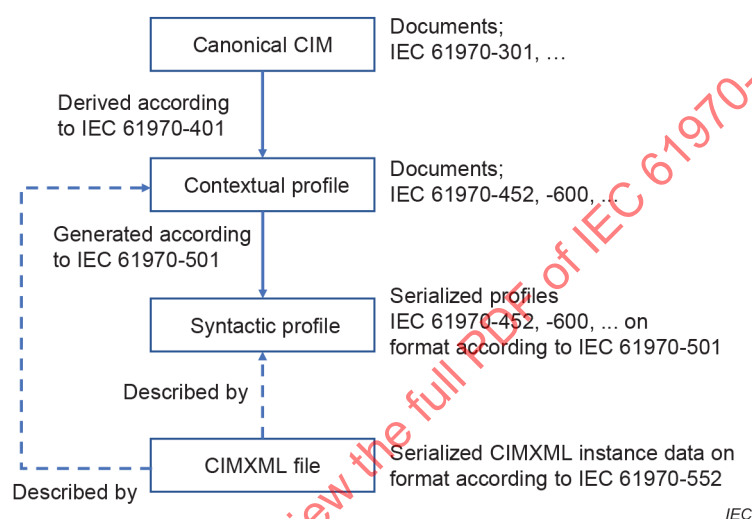


Figure 1 – Relations between standards and profiling

The Canonical CIM is shown at the top in Figure 1. Based on the Canonical CIM and a context for the usage, a contextual profile is created. A serialisation of the contextual profile is a syntactic profile. The serialisation format for contextual profiles is described by IEC 61970-501. IEC 61970-501 is the only graph based syntactic profile. Other syntactic profile exists, e.g. a UML based tool can serialize the profile in the XMI format, a tool based on Eclipse Modelling Framework (EMF) can serialize a profile in Ecore and an OWL based tool can use RDF/XML. All these syntactic profiles describe the same contextual profile but, in a format, specific for the chosen serialisation (UML/XMI, Ecore, RDFS, OWL, etc.). Each possible syntactic profile has its own specification, e.g. like IEC 61970-501.

Datasets on CIMXML format are serialized according to IEC 61970-552 and the data in a CIMXML dataset is described by a profile. The contextual and syntactic profile describe the same information, the only difference being that the syntactic profile is machine readable and can be used in processing or validation of CIMXML datasets.

Tools that process data described by profiles, created according to this document, need a machine-readable serialization of the profiles, the syntactic profile. IEC 61970-501 is an RDFS based serialization intended for this. Profiling tools shall support the generation of profiles in the IEC 61970-501 serialisation format. Any tool, regardless of whether it is based on UML or OWL, will have to serialize profiles into a syntactic profile that is used for exchange with other tools or saved for persistency. IEC 61970-501 does not contain all information needed by profiling tools, e.g. it is possible to generate a IEC 61970-501 syntactic profile from a UML representation but not possible to recreate UML from a profile on IEC 61970-501 format. Hence IEC 61970-501 cannot be used as an exchange format between UML based profiling tools.

Standardized vocabularies, that may be used in addition to Canonical CIM, are available from many sources, e.g.:

- The Dublin Core™ from the Dublin Core Metadata initiative dated 18 December 2006,
- The Profiles Vocabulary (DX-PROF) from W3C, a note dated 18 December 2019,
- Profile Guidance from W3C, a draft dated 24 November 2021.

How such vocabularies may be used together with Canonical CIM is not described and requires further development.

4.2 Relations between Canonical CIM, profiles and datasets

The CIM specifications can be divided in two groups:

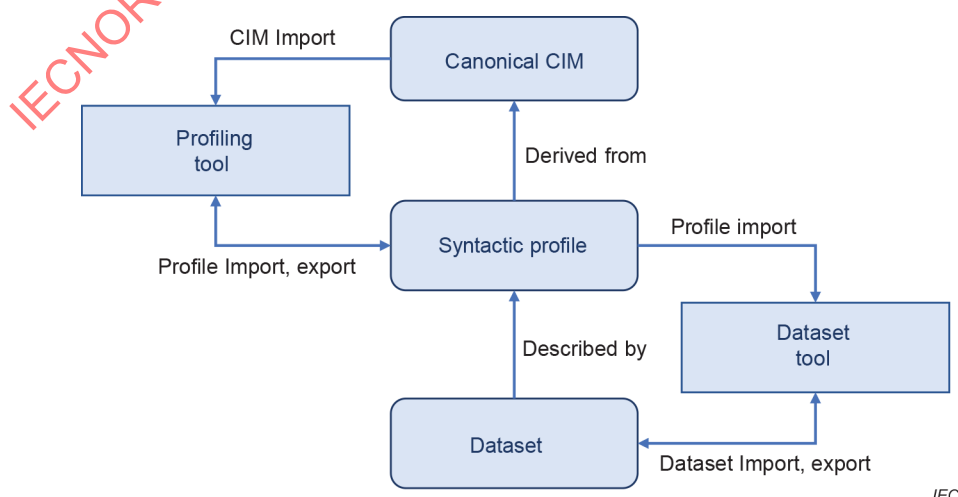
- Canonical CIM,
- Profile derived from Canonical CIM.

The Canonical CIM is an information model for the power system domain standardized by IEC and it is intended to be normalized (information is described only once) without duplicate definitions of the same concept. Canonical CIM is intended to be an information model for the concepts in a domain and not a description of data exchanged by messages.

A profile is a subset of the Canonical CIM intended for a specific data exchange in a specific context, hence a profile is an information model for exchanged data. Information from Canonical CIM is divided into profiles that describe data exchange in different contexts; examples are equipment data that is relatively stable with a low change frequency and power system state that change in times from milliseconds up to hours depending on the use case. The context for equipment data is typically a data modeller tool where the equipment data is maintained while the context for power system state could be real time systems (WAMS, SCADA) or operational planning systems.

IEC 61970 (all parts) support analytical functions used in studies of power systems. A case used in a study is assembled by several datasets described by different profiles, e.g. to run a power flow both equipment data and power system state are needed. Hence the IEC 61970 profiles are dependent on each other and use cases typically involve multiple datasets described by different profiles.

This Subclause (4.2) explains the relationship between the information discussed above and in 4.1. Figure 2 shows the dependencies.



IEC

Figure 2 – Relation between Canonical CIM, profiles and datasets

Figure 2 describes the relations between the Canonical CIM, profiles and datasets. The Canonical CIM at the top of Figure 2 is described in IEC 61970-300 (all parts) and in IEC 61968-11. The profiling process described in this document is represented by the arrow “Derived from” and the “Profiling tool” that is used to create the “Syntactic profiles”. Examples of profiles created using the profiling process are IEC 61970-450 (all parts) and IEC 61970-600 (all parts).

The datasets (bottom of Figure 2) are typically exchanged as CIMXML files but other data formats are possible if they comply with profiles generated according to this document. A dataset is described by a profile (“Described by” in Figure 2) and is serialized as CIMXML files as described in IEC 61970-552.

The “Dataset tool” processes “Datasets”, e.g. export, import, validation, etc. A “Dataset tool” could be hardcoded against a set of profiles but could also import profiles and use them upon processing as indicated in Figure 2 with the arrow “Profile import”. For this to work, a machine-readable version of a profile is needed such as a syntactic profile on RDFS format as described in IEC 61970-501. Serializations for other information modelling languages are also possible, e.g. XML, OWL or Ecore, but IEC 61970 uses RDF based technologies as defined by IEC 61970-501. The ability for a “Dataset tool” to import and use a machine-readable version of a profile (“Profile import” in Figure 2) is sometimes referred to as a model-driven architecture as the tool adapts to the profiles. To support a model-driven usage the “Dataset tool” the “Profiling tool” shall import and export syntactic profiles on IEC 61970-501 format. Example use cases for syntactic profiles are:

- Profiling tools that read, update, and save profiles.
- Validation tools that read CIMXML files and check that they comply with syntactic profiles.
- Data management tools that import and export CIMXML files.
- Analytical functions importing input data and exporting result data.

The syntactic profile language in IEC 61970-501:2006 has custom extensions to RDFS as the RDFS standard had not yet been finalized when IEC 61970-501 was published. RDFS has evolved and has been extended so that IEC 61970-501:2006 custom extensions can be replaced. Also note that IEC 61970-501:2006 does not provide sufficient information to enable tool interoperability: other formats need to be considered for this, e.g. XML that is standardised by OMG and OWL standardized by W3C.

4.3 Profiles and business processes

Datasets are exchanged in the context of a business process which typically means multiple datasets defined by different profiles. The Canonical CIM is typically split over the different profiles. Consequently, the datasets need to be assembled into a complete model whereby each dataset contributes data according to its defining profile.

IEC 61970-450 (all parts) and IEC 61970-600 (all parts) describe data that are used in analytical calculations (e.g. power flow calculations, security analysis, short circuit calculations). Input to such calculations are models including a complete set of data needed to solve the analytical calculation. It is common to divide data in groups. Profiling support grouping by dividing information in multiple profiles where each profile describes one group. Reasons for creating data groups could be different origins and change rates. Power system equipment is stable over time and is only changed when new equipment is added, or old equipment removed. Hence data describing the power system equipment has a low change rate and typically originate from dedicated data management tools. Power system state may change monthly for long term planning, hourly for operational planning, secondly for real time operation and down to millisecond for studies of the dynamic behaviour. Hence power system state data benefit from being exchanged separately from equipment data. A result of this is that data in a model used by analytical functions can be divided in profiles that support exchange of parts of the model as CIMXML files. A consequence is that running an analytical calculation with CIMXML files as input requires an assembly process where the CIMXML files are assembled into a complete

model. The profiling process shall then ensure that the profiles describe the relations not only within a CIMXML file but also between data in different CIMXML files.

Whether an assembled model is complete and valid is determined in a validation process where all the profiles defined for the business process are used. A per profile validation is in many cases not meaningful as information distributed over different profiles cannot be checked until the complete model has been assembled.

5 Profile document structure

The IEC 61970-450 and IEC 61970-600 profile documents shall strictly follow a common structure including the following clauses:

- 1) "Scope" that describes the document application.
- 2) "Normative references" that contains a list of publications that are referred to in the text in such a way that some or all of their content constitutes the requirements of this document.
- 3) "Terms, definitions and abbreviated terms" used in the profile.
- 4) "Application profile specification" that specifies the following
 - a) A reference to the canonical information model as an XMI file name from which the profile is derived.
 - b) A list of the profiles defined in the document. Each profile definition consists of:
 - i) Name, e.g. StateVariables (SV),
 - ii) Version, e.g. 4.,
 - iii) Date of the version,
 - iv) URI that uniquely identifies the profile including its version, e.g. <http://iec.ch/TC57/2013/61970-456/StateVariables/4>.
 - c) Requirements
 - d) Constraints
- 5) "Overview" that describes the context in which this profile is intended to be used and the relation to other profiles. Use cases for which the profile is applicable are also described.
- 6) Use cases that describe where the profile is applicable.
- 7) Profile sections describing the information in the profiles.

6 Profiling Use cases

6.1 Overview

This Subclause 6.1 describes the use cases for how profiles are created from the Canonical CIM. This subclause provides a descriptive overview. The normative use is described in Clause 7.

Profiling is the process where UML classes, attributes, roles and datatypes are selected to be included in a profile. This process follows the rules described in Clause 7.

In the profiling process, a subset of the attributes and roles from a class in Canonical CIM are included in a profile. It is then possible to change attribute and role cardinalities. If new information is needed that is not in Canonical CIM, existing classes can be extended with more attributes or roles as well as new classes added to the Canonical CIM. When adding new attributes new datatypes may also be needed, they are then first added in the Canonical CIM.

Profiling use cases that have been identified can be done as follows.

- a) Include different sets of attributes and roles from the same class in different profiles, see 6.2.
- b) Include overlapping sets of attributes and roles from the same class in different profiles, see 6.3.
- c) Include a base class with different sets of attributes or roles, see 6.4.
- d) Extend an existing class, see 6.5.
- e) Extend an existing class used differently in different profiles, see 6.6.
- f) Different cardinalities, see 6.7.
- g) Add a new datatype, see 6.8.

The above profiling cases are limited to the use cases related to the IEC 61970-450 profiles and modelling capability of UML. When profiling is made additional constraints that goes beyond the capability of UML are often needed. Such constraints could be

- Constraining the value space of an attribute. This could be numeric value constraints, enumeration lists, string length constraints, string content constraints, complex comparisons including attributes from related classes, etc.
- Constraining role cardinalities for subclasses inheriting associations from a more generic class.
- Constraining the occurrence of class instances depending on context.

Constraints like this can be described by the formal languages OCL or SHACL. As OCL or SHACL constraints are not included in the Canonical CIM they are not part of the profiling process but added to the profiles after profiling has been made.

Subclauses 6.2 to 6.8 contain examples from existing profiles, e.g.:

- Steady State Hypothesis (SSH) from IEC 61970-456 and IEC 61970-600;
- Equipment (EQ) IEC 61970-452 and IEC 61970-600.

6.2 Class in different profiles with no overlap

This is the most common use case. A non-overlapping subset of the attributes and roles from a class is included in one or more profiles, see Figure 3.

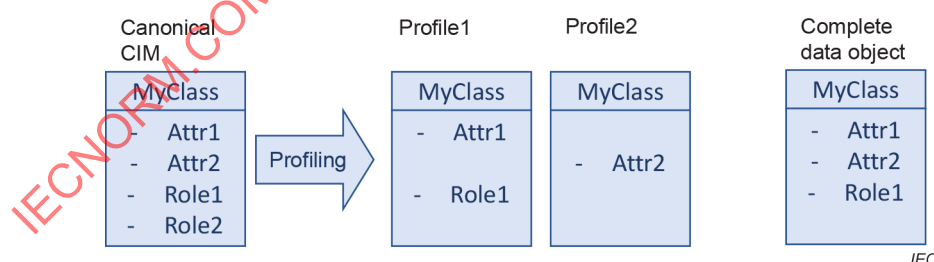


Figure 3 – Class in different profiles with no overlap

An example of MyClass in Figure 3 could be any CIM class with power flow injection data in the SSH profile and static data in the EQ profile, e.g. cim:EnergyConsumer.

A power flow application will then have to collect data from both the EQ and SSH profile datasets to set up the power flow case. A complete object with both EQ and SSH data may then be constructed and described by the “Complete data object” as in Figure 3. Note that inheritance should also be considered but for simplicity it is not shown in Figure 3.

6.3 Include overlapping sets of attributes and roles from the same class in different profiles

This is an uncommon use case that appears in a few cases. Having the same attribute or role in two different profiles can result in ambiguity as to which data to use in datasets for the different profiles. Attributes in the profiles may be given slightly different meanings, e.g. a default value in one profile and current value in another. This way of profiling shall be avoided as it is likely to create confusion. The meaning defined in Canonical CIM shall not be changed.

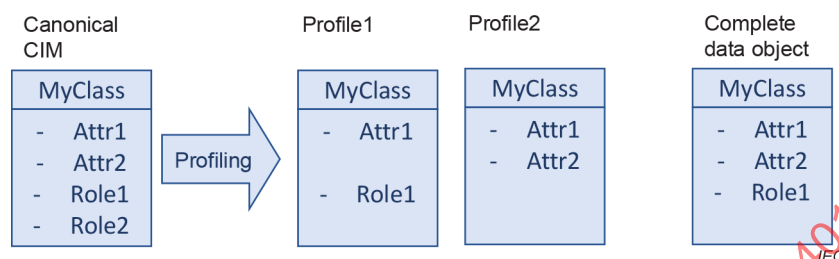


Figure 4 – Class in different profiles with overlap

Figure 4 describes the same case as in Figure 3 with the difference that the same attribute “Attr1” is included in more than one profile. An example is `cim:OperationalLimits` where the limit values appear both in the equipment profile (IEC 61970-452:2021) and the Solved power system state profiles (IEC 61970-456:2018).

6.4 Include a base class with different sets of attributes or roles

This is a case where the same class is included differently in different profiles.

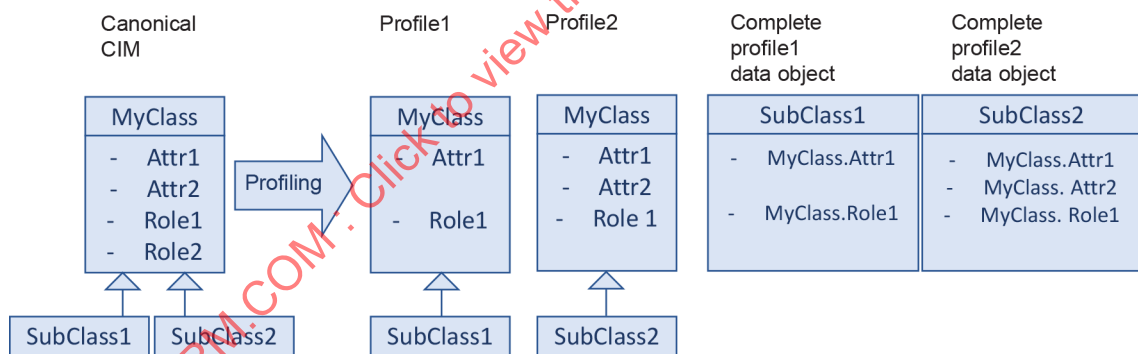


Figure 5 – Same base class with different sets of attributes or roles

Figure 5 shows a case where the “MyClass” is the base class to two other classes “SubClass1” and “SubClass2”. In the profiling, “SubClass1” is put in “Profile1” and “SubClass2” is put in “Profile2”. The attributes and roles from “MyClass” included in the profiles are not exactly the same. An example would be if “MyClass” is `cim:IdentifiedObject`, where `cim:IdentifiedObject.name` and `cim:IdentifiedObject.description` is included in “Profile2” but only `cim:IdentifiedObject.name` in “Profile1”. “SubClass2” could be any class in the EQ profile and “SubClass1” could be `cim:TopologicalNode` in the TP profile.

6.5 Extending an existing class

A class is extended with the requirement that the extension be the same for all profiles. The rules for extending the Canonical CIM are outside the scope of this document.

In the profiling step the extension attributes and roles are moved into a profile version of the class as shown in Figure 6.

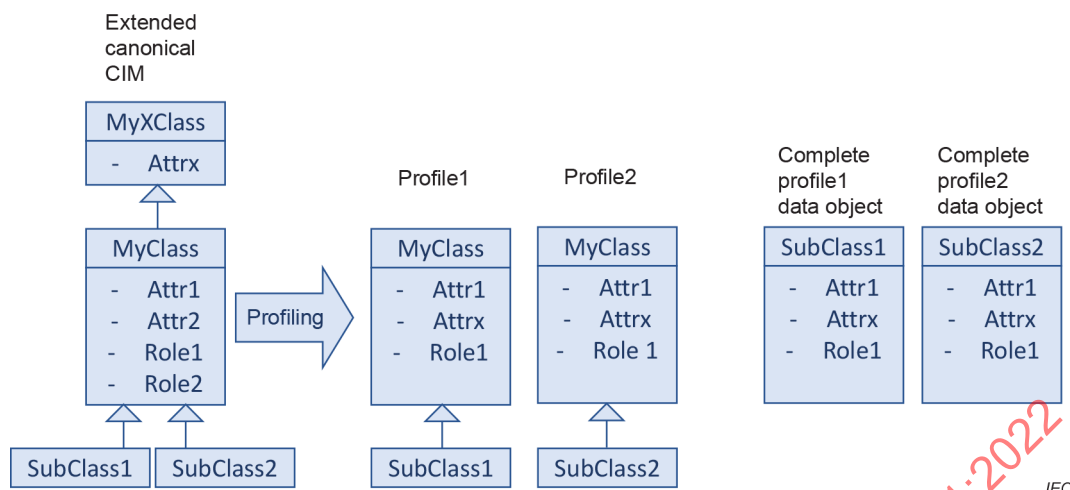


Figure 6 – Extending a class the same way in all applicable profiles

Figure 6 shows how “MyClass” is extended with an attribute “Attrx” inherited from “MyXClass”. “Attrx” is included in “MyClass” for the profiles where “MyClass” is included. An example of this case is when `cim:IdentifiedObject` is extended with a new attribute (e.g. `shortName`) used in all profiles where `cim:IdentifiedObject` is included, e.g. most classes in the EQ profile and `cim:TopologicalNode` in the TP profile.

6.6 Extending an existing class used differently in different profiles

A class is extended with the requirement that the extension be the same for all profiles where it is used. This is the same case as in 6.5 but the extension is included differently in different profiles.

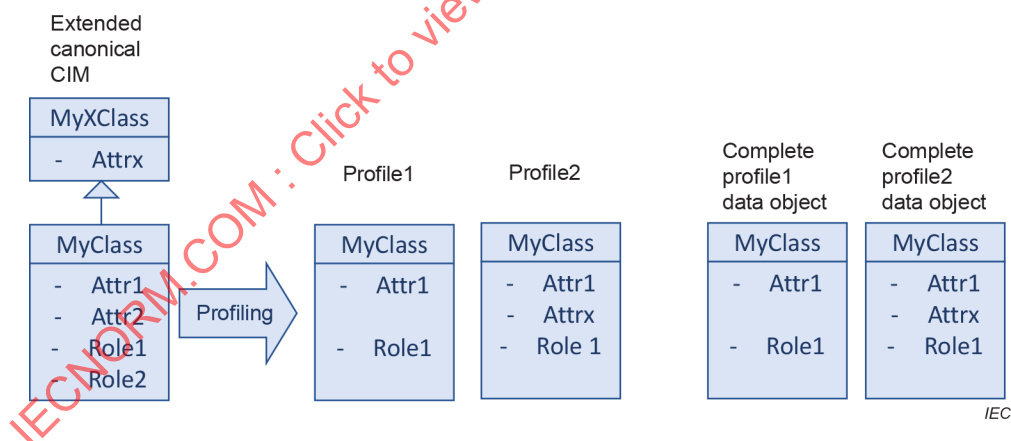


Figure 7 – Extending an existing class used differently in different profiles

In Figure 7 “MyClass” has the extension “Attrx” in “Profile2” but not in “Profile1”. This is the same use case as in 6.4 but for extensions.

An example is `cim:ConnectivityNode` and `cim:TopologicalNode` extended with additional data used in boundary profiles leaving the non-boundary versions of `cim:ConnectivityNode` and `cim:TopologicalNode` unchanged.

6.7 Different cardinalities

In the case where a class is included in different profiles as in 6.4 and 6.6, it is possible to use different cardinalities for the same attribute in the different profiles which means that the profiled classes have the same class name but different attribute or role cardinalities.

This is a violation of object-oriented design as a class with the same name is defined differently in different profiles. An example is if `cim:IdentifiedObject.name` is required in the equipment profile but not in the topology profile as is the case in IEC TS 61970-600. Object-oriented designs require that different classes have different names.

As discussed in 6.1, the IEC 61970-450 series profiles describe CIMXML documents that are assembled into models used by analytical calculations. If then instances with the same class name but with different attribute or role cardinalities appear in the model this would require different class names to distinguish the cardinality differences. As this is not the case implementers need to spend extra effort to work around this, e.g. use different class names within the implementation or use some decoration technique.

6.8 Add a new datatype

When attributes are added it may be that the existing datatypes do not cover the needs. Hence a new datatype needs to be added.

7 Profiling rules

7.1 Overview

This Clause 7 includes rules that shall be followed when creating IEC 61970 profiles. The preceding clauses give the background to the rules.

7.2 R0101 Information model class, attribute, role and datatype names

When profiling, it is not permitted to change the class, attribute, role or datatype names. The reason is to enable direct tracing of the profiled entities back to the Canonical CIM which enables interoperability between the created profiles. This is also a prerequisite for efficient model assembly (see 6.1) as it does not require any mapping between the information in the different profiles.

7.3 Class rules for classes without stereotype

7.3.1 R0201 Including a class

Only classes from the Canonical CIM or extensions based on Canonical CIM can be included in a profile.

7.3.2 R0202 Adding a new class

If a new class is needed the Canonical CIM shall be extended with the class as described in Clause 8. After this extension, the new class is included as described in 6.5.

7.3.3 R0203 The description of a class

The description of the class in the Canonical CIM or extensions based on Canonical CIM shall be included unchanged, and no additions are allowed.

7.3.4 R0204 Name of a class

Changing the name of a class included in a profile is not allowed.

7.3.5 R0205 Cardinality of a class

It shall be possible to specify the cardinality of a class, i.e. the number of instances that can appear for a class.

Examples where it would be useful to specify class cardinality are as follows.

- At least one `cim:ConnectivityNode` for models built with `cim:ConnectivityNodes`.
- Exactly one `cim:ControlArea` is required for a TSO network model.
- At least one `cim:TopologicalIsland` is required for a solved power flow case.

7.3.6 R0206 Concrete class

For a class it is possible to specify if it is allowed to have any instances, i.e. the class is concrete. A class that is not allowed to have any instances shall be declared abstract (a UML meta data attribute).

7.4 Attribute rules

7.4.1 R0301 Including an attribute

For UML an attribute from a class in the Canonical CIM or extensions based on Canonical CIM can be included if the class having the attribute is already included in the profile.

7.4.2 R0302 Adding a new attribute

If a new attribute is needed for an already existing class in the Canonical CIM, the class shall be extended with the new attribute before including it in the profile as described in Clause 8.

7.4.3 R0303 The description of an attribute

The description of an attribute in the Canonical CIM or extensions with additional attributes based on Canonical CIM shall be included unchanged.

7.4.4 R0304 The name of an attribute

It is not allowed to change the name of an attribute included in a profile.

7.4.5 R0305 The cardinality of an attribute

The cardinality of an attribute can be made more restrictive, i.e. cardinality changed from 0..1 to 1..1. The profile shall not make cardinality less restrictive as defined in Canonical CIM or extended Canonical CIM.

7.4.6 R0306 The datatype of an attribute

The datatypes are defined in the Canonical CIM or an extended Canonical CIM. Datatypes are classified into different types by using a stereotyped class, such as one of Primitive, CIMDatatype, Compound or enumeration.

The datatype of an attribute in the Canonical CIM or extensions based on Canonical CIM shall not be changed in a profile.

7.4.7 R0307 Initial value of an attribute

If an attribute has an initial value in the Canonical CIM or in extensions based on Canonical CIM it shall not be overridden in the profile.

If an attribute is without an initial value in the Canonical CIM or extensions based on Canonical CIM it can be specified in the profile.

7.5 Datatypes

7.5.1 R0401 Including a datatype

A datatype used by an attribute in a profile shall be included in the profile.

7.5.2 R0402 Adding a new datatype

If a new datatype is needed, the Canonical CIM shall be extended with the new datatype before including the new datatype in the profile.

7.5.3 R0403 Primitive datatype

A Primitive datatype used in a profile shall be included unchanged.

The value space of an existing Primitive datatype can be further restricted by use of additional constraints described in OCL or SHACL, e.g. string values can be restricted by regular expressions or lists of values.

7.5.4 R0405 CIMDatatype

A CIMDatatype stereotyped class has three attributes: value, unit, and multiplier. A profile may decide that the unit and multiplier are determined for the profile or the context for the exchange, so they are the same for all instances of a particular CIMDatatype stereotyped class. If so the attributes value, unit, and multiplier are removed and the CIMDatatype stereotyped class is assigned the datatype from the value attribute in the CIMDatatype stereotyped class.

Hence a CIMDatatype stereotyped class can be included in a profile in two ways.

- a) The CIMDatatype stereotyped class is flattened by excluding the attributes value, unit, and multiplier and given the datatype from the value attribute value in the CIMDatatype stereotyped class.
- b) The CIMDatatype stereotyped class attributes value, unit and multiplier are included as is.

Which of the profiling methods in a) and b) is used is determined per profile or context for the data exchange.

The value space of an existing CIMDatatype may be further restricted by use of additional constraints described in OCL or SHACL.

7.5.5 R0405 Compound datatype

A compound datatype follows the same rules as a class.

7.5.6 R0406 Enumeration

An enumeration datatype used in a profile shall be included unchanged to avoid interoperability issues between profiles using different subsets of the enumeration in Canonical CIM.

7.5.7 R0407 The description of a datatype

The description of a datatype in the Canonical CIM or extensions based on Canonical CIM shall be included unchanged.

7.5.8 R0408 The name of a datatype

It is not allowed to change the name of a datatype included in a profile.

7.6 Association rules

7.6.1 R0501 Including an association

An association in the Canonical CIM or extensions based on Canonical CIM can be included if both of the classes referenced by the association roles are present in:

- a) the profile.
- b) two different profiles and both profiles are included in the data exchange.

A check (tool dependent) shall be carried out to verify that the set of profiles covers a complete information model including a full description of the two classes in an association across profiles.

7.6.2 R0502 Adding a new association

If a new association is needed, the Canonical CIM shall be extended with the new association before adding the association in the profile.

7.6.3 R0503 The names of the two roles in an association

The names for both roles in an association shall be present in Canonical CIM. If this is not the case, this is an error that shall be corrected before profiling is done.

The names of roles in an association shall not be changed in a profile.

7.6.4 R0504 The cardinality of an association role

The cardinality of an association role can be made more restrictive in a profile e.g. cardinality changed from 0..1 to 1..1. The profile shall not make cardinality less restrictive as defined in Canonical CIM or extended Canonical CIM.

7.6.5 R0505 Association navigability

An association in Canonical CIM does not specify the navigability for the roles but a profile shall add this information to specify which of the two roles shall be exchanged. This avoids exchanging redundant information.

The navigability shall always be specified for an association.

Note that IEC 61970-501 does not include the navigability information. IEC 61970-501 needs to be updated with a description on how navigability is described. Some other formats describe the navigability by excluding the role in the non-navigable direction resulting in lost cardinality information.

7.6.6 R0506 The description of an association

The description of an association in the Canonical CIM or extensions based on Canonical CIM shall be included unchanged.

7.7 Attribute and association restrictions

7.7.1 R0601 Overlap between profiles

Overlapping profiles include the same attribute or role. This means that the same information can be exchanged using different profiles. This is described by the use case in 6.3.

Profiles with overlapping attributes or associations are not allowed.

7.7.2 R0602 Same base class with different sets of attributes and associations in different profiles

As described by use cases in 6.4 and 6.6 the same base class can be included in different profiles with different sets of attributes and associations.

If the intent of the base class is to provide the same kind of information to a profile it shall be profiled the same way in all profiles where it is included, i.e. exactly the same set of attributes and associations shall be included in all profiles. If this is not the case the class, despite having the same name, will be different, e.g. have different:

- sets of attributes,
- sets of associations,
- cardinalities for the same attribute or association role.

Differences as described in the bulleted list above are not allowed in different profiles.

7.8 R0701 Inheritance structure

The inheritance structure from the Canonical CIM or extensions based on Canonical CIM shall be kept unchanged in a profile.

7.9 R0801 Constraints

The UML language lacks the capability to describe constraints other than what is described by cardinalities. The Object Constraint Language (OCL) is currently used to describe constraints on profiles. The rules may use attributes or roles in statements that describe the constraint.

Common cases where additional constraints are desired are as follows.

- Numerical values where only values in a range are meaningful, e.g. a resistance that is negative means generation of active power which is not meaningful.
- A class high up in an inheritance hierarchy has an association that is meaningful only for a subset of the derived classes. The containment structure defined by the association `cim:Equipment-cim:EquipmentContainer` can be restricted for the subtypes, e.g. `cim:Line` can be restricted to only contain `cim:ACLineSegments`.

OCL can be used to implement restrictions described in the bulleted list above. OCL is similar to a programming language where an OCL statement is linked to a class in a profile. This means that when a restriction is implemented in OCL a decision on what class to place it on is needed. In the above example with the containment many design choices are possible.

- 1) Place the OCL code on the leaf classes to `cim:Equipment`.
- 2) Place the OCL code on the leaf classes to `cim:EquipmentContainer`.
- 3) Mixing 1 and 2 is possible but is a poor design choice.
- 4) Place the OCL code on `cim:Equipment`.
- 5) Place the OCL code on `cim:EquipmentContainer`.
- 6) Mixing 4 and 5 is possible but is a poor design choice.
- 7) Mixing 1 and 4 or 2 and 5 is possible and can make sense in some cases.

The numbered list above implies that OCL allows for many design choices, hence a restriction shall always be described in (English) language before the corresponding OCL rules are created as OCL rule creation is similar to programming.

Hence the following applies:

- Restrictions shall be described in (English) text on the restricted UML artefact (Class, Attribute or Role) either in Canonical CIM or in a profile.
- A restriction described in (English) text shall have a translation to OCL that can be implemented on one or more UML classes.

It is also possible to restrict data by use of XML Schema (xsd) or XSLT scripts with regular expressions. These technologies are well suited for syntax checks and are an alternative to OCL in the case of syntax checks.

8 Extending Canonical CIM

Canonical CIM can be extended in two ways:

- 1) By adding a new class where associations and attributes are mixed into an existing class with multiple inheritance. This will leave the existing class unchanged in the sense that its attributes and associations in Canonical CIM are unchanged. Figure 6 in 6.5 shows an example of this.
- 2) By subtyping an existing class. This method has not been used to extend Canonical CIM yet.

Extension classes in the above list shall be included in a separate package and not be included in the Canonical CIM packages. This makes it easy to distinguish them from the Canonical CIM classes.

If a base class is extended according to case 1), the rules in 7.7.2 shall be observed meaning that the base class shall be profiled the same way in all profiles where it is included. When the extended class is included in a profile, the mix in inheritance is removed. The attributes and association roles are instead included directly in the extended class as shown in Figure 6.

If the extension is intended to describe different types of objects, then extension according to case 2) above is recommended. However, making extensions this way creates the risk of conflicting with later extensions of Canonical CIM. The following example illustrates this

- The class `cim:GeneratingUnit` in Canonical CIM is extended according to case 2) with new attributes supporting market applications and the new class is called `cim:GeneratingUnitMkt`.
- The `cim:GeneratingUnit` has been subtyped into the classes `cim:ThermalGeneratingUnit` and `cim:HydroGeneratingUnit`. The extension of `cim:GeneratingUnitMkt` in the above bullet is not available to these subtypes as they are in different inheritance paths. But if the application data is instead added to `cim:GeneratingUnit` according to case 1) it will be available to `cim:ThermalGeneratingUnit` and `cim:HydroGeneratingUnit`. The `cim:GeneratingUnit` is subtyped into other classes and it is now realized that the attributes in `cim:GeneratingUnitMkt` are also needed for these subclasses.

This is one example of how the extension of Canonical CIM shall not be done. The topic of how to carry out information modelling is wide, and outside the scope of this document.

9 Requirements for a profiling tool

9.1 Minimum requirements

A profiling tool shall comply with the following minimum requirements. A tool shall:

- R1001 have a user interface that support the functions listed below in this bulleted list.
- R1002 be able to include classes, attributes, associations and datatypes from Canonical CIM or an extended Canonical CIM;

- R1003 be able to read and write a profile according to the existing format specifications , e.g. RDF/XML according to IEC 61970-501:2006;
- R1004 in the case of a profile in UML, be able to export and import UML XMI for exchange with commercial UML tools.
- R1005 be able to restrict cardinalities and initial values of a class, an attribute, an association, a datatype;
- R1006 be able to flatten a CIMDatatype stereotyped class by removing the attributes value, unit and multiplier as described in 7.5.4;
- R1007 be able to visualise profiles graphically in UML or other technology;
- R1008 support synchronisation of a profile with an updated Canonical CIM to support evolution on the profile in parallel with Canonical CIM. The CIM namespace is the same in this case;
- R1009 support migration of profiles from one version of Canonical CIM to another. The CIM namespaces are different in this case;
- R1010 be able to specify the direction of associations;
- R1011 retain the meaning of stereotypes and tagged values.

9.2 Extended requirements for OCL rules

The extended requirements describe additions to the minimum requirements in 9.1. The extended requirements are optional and cover requirements on tools managing OCL rules.

Based on experiences from EMF tools that support maintaining OCL rules and validating data with OCL rules the following extended requirements shall be supported:

- R1101 a tool shall have a user interface that shows the information model with OCL rules and supports the functions listed below in this bulleted list.
- R1102 serialize profiles to the Ecore XML to support import and validation with EMF.
- R1104 editing of the OCL rules is linked to the classes in the information model.
- R1105 context sensitive editing of OCL rules such that errors are reported while typing.
- R1106 export of OCL rules from an information model as a separate file on Ecore XML format.
- R1107 import of OCL rules to an information model from any number of separate files containing OCL rules on Ecore XML format.
- R1108 import CIMXML instance files in a model and validate them with the information model extended with OCL rules.

R1109 creation of XML Schemas for syntax checks.

Annex A (informative)

A.1 Mapping of UML to OWL

A profile can be defined in UML or OWL. Hence the two languages have constructs that are capable of describing the same semantics. It is then important to know how these constructs map to each other, see Table A.1.

Table A.1 – UML and ontology languages

CIM UML elements	OWL elements
Package	Ontology
Class	Class
Class stereotyped as "Compound"	Class
Simple valued attribute	Data property
Class attribute or role	Object property
Association	Object properties with inverse relation
Class stereotyped as Primitive	Equivalenced with XML Schema types
Class stereotyped as CIMDatatype	The CIMDataType value attribute are mapped to XML Schema types and the internal CIMdatatype attributes value, unit and multiplier are removed (flattened)
Role cardinality	Restriction on Object property (minQualifiedCardinality and maxQualifiedCardinality)
Inheritance between classes	Subtypes
Enumeration literal	An instance
Enumeration class	A class describing the enumeration member instances

Most of the mappings in Table A.1 are straightforward except for the stereotypes where additional assumptions need to be made.

Table A.1 includes constraints as UML does not have a formal constraint language.

OCL is a constraint language that is defined for object data and SHACL is a constraint language defined for graph data. This annex does not describe a mapping between OCL and SHACL.

Mapping of stereotypes and tagged values are made case by case. The stereotypes Primitive and CIMDatatype are described in Table A.1, other stereotypes will need their own rows in Table A.1.

Clause A.2 discusses the units and multiplier issue. Table A.1 does not include rows for this issue as it is not related to the mapping of UML to OWL. The units and multiplier issue is rather an incompatibility between two different profiles.

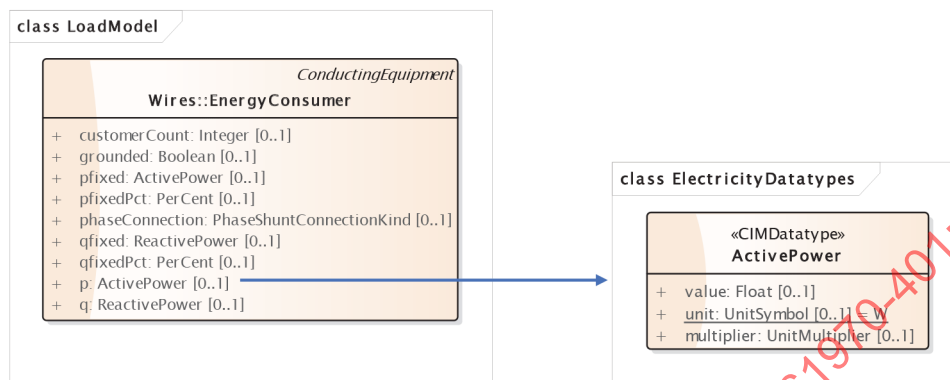
A.2 Units and multipliers issue

A.2.1 Description of issue

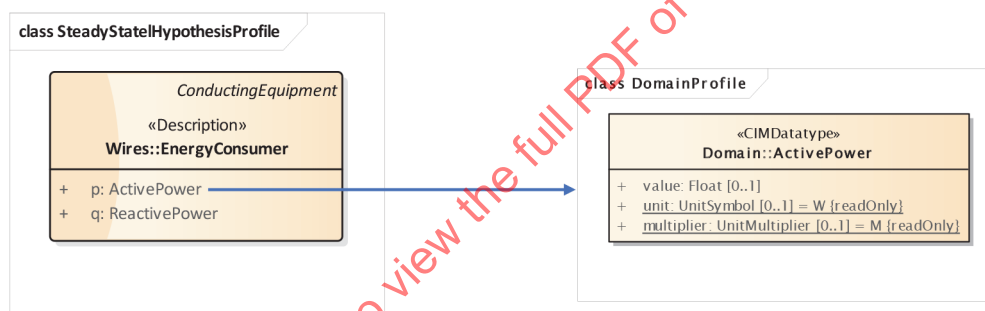
The issues described in this Subclause A.2.1 are exemplified with syntactic profile examples showed by figures based on IEC 61970-501:2006.

Canonical CIM have CIMDatatype stereotyped classes with attributes for value, unit, and multiplier. There is one such class per quantity, e.g. active power, voltage, etc. In Canonical CIM an attribute of a particular quantity is given a datatype described by a class stereotyped CIMDatatype, e.g. an active power attribute “p” is given a datatype described by the class “ActivePower” as exemplified in Figure A.1.

Canonical CIM



SSH Profile



IEC

Figure A.1 – CIMDatatype example in Canonical CIM and SSH profile

Figure A.1 shows the UML for Canonical CIM and the IEC 61970-456 profile (SSH) when viewed in a UML tool.

The IEC 61970-456 profiles include the CIMDatatype stereotyped classes as given in Canonical CIM. The way in which the ActivePower CIMDatatype from Canonical CIM appears in IEC 61970-452:2021 SSH profile is shown in Figure A.2.

5.2.178 ActivePower datatype

Product of RMS value of the voltage and the RMS value of the in-phase component of the current.

Table 306 shows all attributes of ActivePower.

Table 306 – Attributes of CoreEquipmentProfile::ActivePower

name	mult	type	description
value	0..1	Float	
multiplier	0..1	UnitMultiplier	(const=M)
unit	0..1	UnitSymbol	(const=W)

Figure A.2 – ActivePower from IEC 61970-452:2021 (CIM16)

Figure A.2 shows a tabular representation of the SSH profile to the lower right in Figure A.1.

Figure A.3 shows the profile for the ActivePower example according to IEC 61970-501. This matches with the IEC 61970-450 specifications shown in Figure A.2 and the lower SSH part in Figure A.1.

```

<rdf:Description rdf:about="#ActivePower">
  <rdfs:label xml:lang="en">ActivePower</rdfs:label>
  <rdfs:comment rdf:parseType="Literal">Product of RMS value of the voltage and the RMS value of the in-phase
  <cims:stereotype>CIMDatatype</cims:stereotype>
  <cims:belongsToCategory rdf:resource="#Package_Domain"/>
  <rdf:type rdf:resource="http://www.w3.org/2000/01/rdf-schema#Class"/>
</rdf:Description>
<rdf:Description rdf:about="#ActivePower.value">
  <cims:stereotype rdf:resource="http://iec.ch/TC57/NonStandard/UML#attribute"/>
  <rdfs:label xml:lang="en">value</rdfs:label>
  <rdfs:domain rdf:resource="#ActivePower"/>
  <cims:dataType rdf:resource="#Float"/>
  <cims:multiplicity rdf:resource="http://iec.ch/TC57/1999/rdf-schema-extensions-19990926#M:0..1"/>
  <rdf:type rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Property"/>
</rdf:Description>
<rdf:Description rdf:about="#ActivePower.unit">
  <cims:stereotype rdf:resource="http://iec.ch/TC57/NonStandard/UML#attribute"/>
  <rdfs:label xml:lang="en">unit</rdfs:label>
  <rdfs:domain rdf:resource="#ActivePower"/>
  <rdfs:range rdf:resource="#UnitSymbol"/>
  <cims:multiplicity rdf:resource="http://iec.ch/TC57/1999/rdf-schema-extensions-19990926#M:0..1"/>
  <cims:isFixed rdf:Literal="W"/>
  <rdf:type rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Property"/>
</rdf:Description>
<rdf:Description rdf:about="#ActivePower.multiplier">
  <cims:stereotype rdf:resource="http://iec.ch/TC57/NonStandard/UML#attribute"/>
  <rdfs:label xml:lang="en">multiplier</rdfs:label>
  <rdfs:domain rdf:resource="#ActivePower"/>
  <rdfs:range rdf:resource="#UnitMultiplier"/>
  <cims:multiplicity rdf:resource="http://iec.ch/TC57/1999/rdf-schema-extensions-19990926#M:0..1"/>
  <cims:isFixed rdf:Literal="M"/>
  <rdf:type rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Property"/>
</rdf:Description>

```

IEC

Figure A.3 – Profile for ActivePower including CIMDatatype attributes

Note that Figure A.3 describes the attributes value, unit and multiplier.

CIMXML data (according to IEC 61970-552) for the active ActivePower example in Figure A.3 should include attributes value, unit, and multiplier. Figure A.4 and Figure A.5 show examples of how an instance of ActivePower with the value attribute appears in a CIMXML file.

```

<cim:ConformLoad rdf:about="#_8395d581-ba01-41a8-8edf-c84956187f0c">
|   <cim:EnergyConsumer.p rdf:resource="#_8395d581-ba01-41a8-8edf-a94856187f0c"/>
</cim:ConformLoad>
<cim:ActivePower rdf:ID="_8395d581-ba01-41a8-8edf-a94856187f0c">
|   <cim:ActivePower.value>5.138035</cim:ActivePower.value>
</cim:ActivePower>

```

IEC

Figure A.4 – CIMXML example with an ActivePower instance as an identified node

Figure A.5 shows how the ActivePower is included as a blank node object without an rdf:ID.

```

<cim:ConformLoad rdf:about="#_8395d581-ba01-41a8-8edf-c84956187f0c">
|   <cim:EnergyConsumer.p>
|       <cim:ActivePower>
|           <cim:ActivePower.value>5.138035</cim:ActivePower.value>
|       </cim:ActivePower>
|   </cim:EnergyConsumer.p>

```

IEC

Figure A.5 – CIMXML example with an ActivePower blank node serialisation example

Note that both Figure A.4 and Figure A.5 include the value attribute according to the profile.

Since the very beginning of CIMXML and IEC 61970-552 the profiles describing CIMDatatypes do not include value, unit and multiplier attributes, the attributes were removed in the profiling (the profiling tool used was CIMTool). The profile for CIMDatatypes, according to IEC 61970-501, is then without the value, unit and multiplier attributes.

Figure A.6 shows the ActivePower example in an IEC 61970-501 compliant profile from an early version of CIM10 where the attributes value, unit and multiplier are not present.

```

<rdf:Description rdf:about="#ActivePower">
|   <rdfs:comment>Product of RMS value of the voltage and the RMS value of the in-phase
|   <rdfs:label>ActivePower</rdfs:label>
|   <rdfs:type rdf:resource="http://www.w3.org/2000/01/rdf-schema#Class"/>
</rdf:Description>

```

IEC

Figure A.6 – Profile for ActivePower based on CIM version 9 (CIM9)

Figure A.7 shows the ActivePower example in an IEC 61970-501 compliant profile from an early version of CIM10 where the attributes value, unit and multiplier are not present.

```

<cim:ConformLoad rdf:about="#_8395d581-ba01-41a8-8edf-c84956187f0c">
|   <cim:EnergyConsumer.p>5.138035</cim:EnergyConsumer.p>|
</cim:ConformLoad>

```

IEC

Figure A.7 – CIMXML example with an ActivePower instance according to the original serialisation

Despite the fact that the IEC 61970-450 series profiles from CIM10 describe CIMDatatypes with attributes value and unit (the multiplier was added later), according to the example in Figure A.2, the serialisation still does not include the attributes value, unit and multiplier. This means that instances of CIMDatatypes appear in CIMXML files as shown by the example in Figure A.7 and not as required by the current profiles as shown in Figure A.4 or Figure A.5. Hence current

IEC 61970-450 series profiles do not correctly represent exchanged instances of CIMDatatype stereotyped classes.

The CIMDatatype stereotyped classes were added in CIM10 in the year 2006 and were then included in the IEC 61970-450 series documents, as exemplified in Figure A.2. But the CIMXML file serialisation from before CIM10 was kept, leading to a mismatch between the profiles and the data in the CIMXML files. This means that tools validating CIMXML files will have to make one of the following changes

- Making the profile compliant with the data by removing the attributes value, unit and multiplier as shown by the example in Figure A.6. This is labelled as “flattening the profile”.
- Making the data compliant with the profile by adding at least the value attribute as shown by the examples in Figure A.4 and Figure A.5. This is labelled as “fattening the data”.

The above solutions are required if off-the-shelf third party software (e.g. Eclipse Modelling Framework or graph/RDF/OWL based software) are used, as compliance between the information model (the profiles) and data (CIMXML) is an absolute requirement.

The above solutions to make data and profiles compatible are not described in any standards, which may create difficulties for implementers.

An issue with the profiles before CIM10 is that the datatype for scalar data is not well described. If a datatype is described by a class in UML the datatype for the class needs also to be described. Say a class ActivePower describes an active power value, then the datatype should be a float number. This was solved by adding the value attribute having the datatype, in the ActivePower example, a float number. But the incompatibility this created between the profile and CIMXML data was left unmanaged.

A.2.2 Long term solution

For exchanges as in IEC 61970-450 and IEC 61970-600 profiles the multipliers are the same in all profiles and there is no need to specify them per profile and even less to specify them per instance. A solution is to specify the multipliers once for the context or to simply use the base SI units thereby not needing the multipliers.

The base SI units are specified in ISO/IEC 80 000 and for this reason they need not be repeated in CIM. Given this CIMDatatype, stereotyped classes will only be included as a datatype in profiles. The datatype could then be described by a tagged value in UML or by inheritance with a class describing the datatype, e.g. a class Float that defines a float number.

A.2.3 Medium term solution

The issue is solved by “flattening” the CIMDatatype stereotyped classes in the profiling step when the contextual profile is created as exemplified for the ActivePower in Figure A.8.

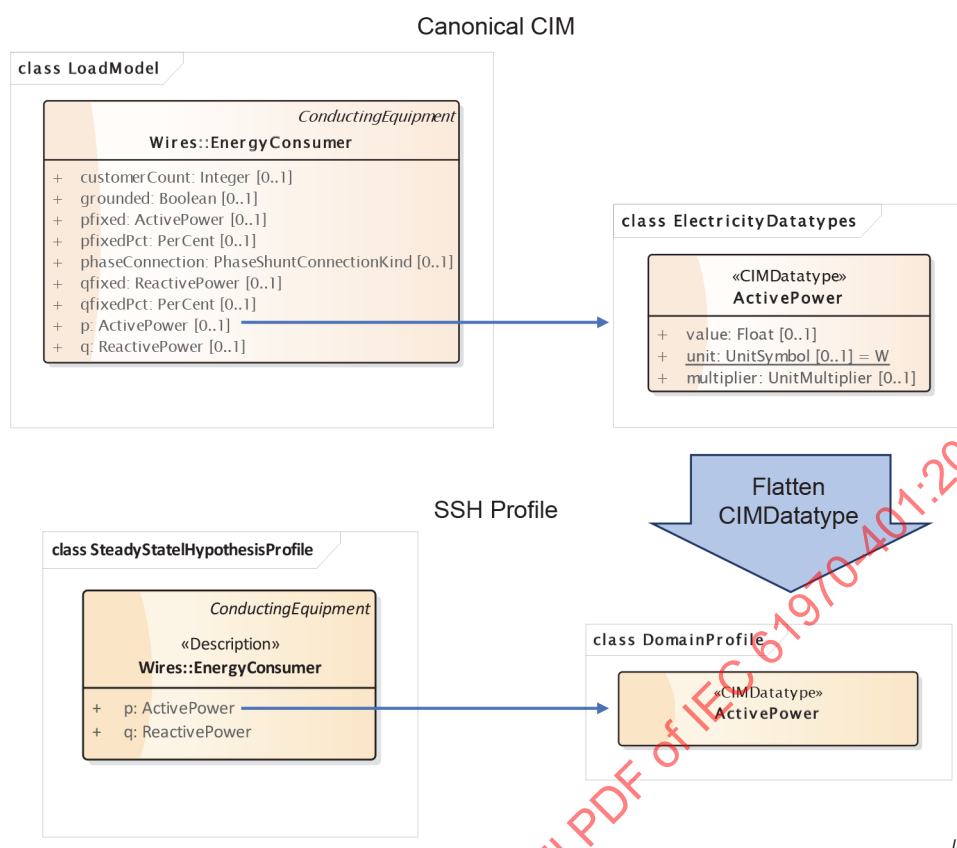


Figure A.8 – Example of Flattened profile for ActivePower

A consequence of this is that all IEC 61970-450 series profiles shall be revised so that the attributes value, unit, and multiplier exemplified in Figure A.2 disappear.

This is a solution that restores the situation to as it was before the CIMDatatype stereotyped classes were introduced in the year 2006. This means that the current mismatch between the profiles and the CIMXML files remains until this change has been implemented in the IEC 61970-450 series documents. Two temporary solutions that are already in use in implementations are described in A.2.4 and A.2.5.

This solution can also be combined with the long-term solution described in A.2.2.

A.2.4 Temporary solution by flattening the profiles

As the attributes value, unit, and multiplier are kept in the IEC 61970-450 series profiles, a temporary solution is to flatten the profile by removing the value, unit and multiplier attributes from the CIMDatatype classes exemplified in Figure A.3, which results in a tweaked profile with CIMDatatype stereotyped class exemplified in Figure A.6. Tools using this solution need to flatten the profiles before processing the data from the CIMXML files.

A.2.5 Temporary solution by fattening the CIMXML data

Another temporary solution is to fatten the CIMXML file by adding the value attribute resulting in a tweaked CIMXML file exemplified in Figure A.4 or Figure A.5. It is also possible to add the unit and multiplier attributes if wanted. Tools using this solution need to fatten the data from the CIMXML files before processing it.

A.2.6 Maintaining information for a flattened profile

Flattening the CIMDatatypes by removing the attributes creates a simple class as exemplified in Figure A.8. Figure A.6 shows the corresponding serialisation according to IEC 61970-501:2006. This means that the datatype, unit and multiplier information is lost. It is possible to keep this information in the following ways

- 1) Change the datatype of the attribute referring to a CIMDatatype to the datatype of the value attribute in the CIMDatatype.
- 2) Let the new class inherit a primitive datatype and stereotype the class as primitive. Add tagged values for the unit and multiplier.
- 3) Keep the CIMDatatype, remove the value attribute and inherit the Primitive datatype for the removed value attribute in the same way as for option 2 but keep the unit and multiplier attributes.

Option 1 is not recommended as the datatype of every attribute in a profile needs to be updated.

Option 2 is a simple change that only affects the datatypes. It is also easy for tools to acquire the datatype by inspecting the base class of the flattened CIMDatatype. For the example in Figure A.8 this means that the ActivePower class to the lower right will be stereotyped Primitive and inherit the Primitive datatype Float. For the unit and multiplier there will be one tagged value each. The benefit of this solution is that the datatype information is clearly visible in diagrams. The downside is that tagged values in UML/XMI do not have a direct mapping to other formats, e.g. RDFS/OWL or Ecore. A possibility is to use custom extensions in the same way as IEC 61970-501:2006 has extended RDFS.

Option 3 intends to avoid the need for custom extensions. If the unit and multiplier are the same for a CIMDatatype in a profile or a set of profiles, the unit and multiplier attributes need not be exchanged in datasets (given that they are optional, which they are), instead the initial values available in the profile are used.

Where the unit and multiplier are different for different instances in a dataset, the CIMDatatype is not flattened, and all three attributes value, unit and multiplier are exchanged in the dataset. A case where this is appropriate is for the CIMDatatype `cim:Money`. An example of the elements in an instance of `cim:Money` could be

```
<cim:Money>
  <cim:Money.value>10</cim:Money.value>
  <cim:Money.unit>EUR</cim:Money.unit>
  <cim:Money.multiplier>k</cim:Money.multiplier>
</cim:Money>
```

IECNORM.COM : Click to view the full PDF of IEC 61970-401:2022

SOMMAIRE

AVANT-PROPOS	37
INTRODUCTION.....	39
1 Domaine d'application	40
2 Références normatives	40
3 Termes, définitions et termes abrégés	41
3.1 Termes et définitions	42
3.2 Termes abrégés	42
4 Vue d'ensemble	43
4.1 Profils et profilage.....	43
4.2 Relations entre CIM canonique, profils et ensembles de données	45
4.3 Profils et processus métier	47
5 Structure des documents de profils	48
6 Cas d'utilisation de profilage.....	48
6.1 Vue d'ensemble	48
6.2 Classe dans des profils différents sans recouvrement.....	50
6.3 Inclusion d'ensembles de recouvrement d'attributs et de rôles de la même classe dans des profils différents	50
6.4 Inclusion d'une classe de base avec différents ensembles d'attributs ou de rôles	51
6.5 Extension d'une classe existante	51
6.6 Extension d'une classe existante utilisée de manière différente dans des profils différents	52
6.7 Cardinalités différentes	52
6.8 Ajout d'un nouveau type de données.....	53
7 Règles de profilage.....	53
7.1 Vue d'ensemble	53
7.2 R0101 Noms de classe, d'attribut, de rôle et de type de données d'un modèle d'information.....	53
7.3 Règles applicables aux classes sans stéréotype	53
7.3.1 R0201 Intégration d'une classe.....	53
7.3.2 R0202 Ajout d'une nouvelle classe	53
7.3.3 R0203 Description d'une classe.....	53
7.3.4 R0204 Nom d'une classe	53
7.3.5 R0205 Cardinalité d'une classe.....	53
7.3.6 R0206 Classe concrète.....	54
7.4 Règles applicables aux attributs.....	54
7.4.1 R0301 Intégration d'un attribut.....	54
7.4.2 R0302 Ajout d'un nouvel attribut.....	54
7.4.3 R0303 Description d'un attribut.....	54
7.4.4 R0304 Nom d'un attribut	54
7.4.5 R0305 Cardinalité d'un attribut	54
7.4.6 R0306 Type de données d'un attribut	54
7.4.7 R0307 Valeur initiale d'un attribut	54
7.5 Types de données	54
7.5.1 R0401 Intégration d'un type de données	54
7.5.2 R0402 Ajout d'un nouveau type de données	55
7.5.3 R0403 Type de données Primitive	55

7.5.4	R0405 CIMDatatype	55
7.5.5	R0405 Type de données Compound	55
7.5.6	R0406 Enumeration	55
7.5.7	R0407 Description d'un type de données	55
7.5.8	R0408 Nom d'un type de données	55
7.6	Règles d'association	56
7.6.1	R0501 Intégration d'une association	56
7.6.2	R0502 Ajout d'une nouvelle association	56
7.6.3	R0503 Noms des deux rôles dans une association	56
7.6.4	R0504 Cardinalité d'un rôle d'association	56
7.6.5	R0505 Navigabilité d'association	56
7.6.6	R0506 Description d'une association	56
7.7	Restrictions concernant les attributs et les associations	56
7.7.1	R0601 Recouvrement entre les profils	56
7.7.2	R0602 Classe de base identique avec différents ensembles d'attributs et d'associations dans des profils différents	57
7.8	R0701 Structure d'héritage	57
7.9	R0801 Contraintes	57
8	Extension du CIM canonique	58
9	Exigences concernant un outil de profilage	59
9.1	Exigences minimales	59
9.2	Exigences étendues pour les règles OCL	59
Annexe A	(informative)	60
A.1	Mise en correspondance entre les langages UML et OWL	60
A.2	Problème lié aux unités et aux multiplicateurs	61
A.2.1	Description du problème	61
A.2.2	Solution à long terme	64
A.2.3	Solution à moyen terme	65
A.2.4	Solution temporaire par aplatissement des profils	66
A.2.5	Solution temporaire par aplatissement des données CIMXML	66
A.2.6	Maintien des informations pour un profil aplati	66
Figure 1	– Relations entre normes et profilage	44
Figure 2	– Relations entre CIM canonique, profils et ensembles de données	46
Figure 3	– Classe dans des profils différents sans recouvrement	50
Figure 4	– Classe dans des profils différents avec recouvrement	50
Figure 5	– Classe de base identique avec différents ensembles d'attributs ou de rôles	51
Figure 6	– Extension identique d'une classe dans tous les profils applicables	51
Figure 7	– Extension d'une classe existante utilisée de manière différente dans des profils différents	52
Figure A.1	– Exemple de CIMDatatype avec CIM canonique et profil SSH	61
Figure A.2	– ActivePower de l'IEC 61970-452:2021 (CIM16)	62
Figure A.3	– Profil pour ActivePower incluant les attributs CIMDatatype	62
Figure A.4	– Exemple CIMXML avec une instance ActivePower en tant que nœud identifié	63
Figure A.5	– Exemple CIMXML avec un exemple de sérialisation de nœud vide ActivePower	63
Figure A.6	– Profil pour ActivePower fondé sur la version 9 du CIM (CIM9)	63

Figure A.7 – Exemple CIMXML avec une instance ActivePower selon la sérialisation d'origine.....	64
Figure A.8 – Exemple de profil aplati pour ActivePower	65
Tableau 1 – Termes abrégés	43
Tableau A.1 – Langages UML et d'ontologie	60

IECNORM.COM : Click to view the full PDF of IEC 61970-401:2022

COMMISSION ÉLECTROTECHNIQUE INTERNATIONALE

**INTERFACE DE PROGRAMMATION D'APPLICATION
POUR SYSTÈME DE GESTION D'ÉNERGIE (EMS-API) –****Partie 401: Cadre de profils****AVANT-PROPOS**

- 1) La Commission Électrotechnique 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. À 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 du présent document 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.

L'IEC 61970-401 a été établie par le comité d'études 57 de l'IEC: Gestion des systèmes de puissance et échanges d'informations associés. Il s'agit d'une Norme internationale.

Cette première édition annule et remplace l'IEC TS 61970-401 parue en 2005. Cette édition constitue une révision technique.

Cette édition inclut les modifications techniques majeures suivantes par rapport à l'édition précédente:

- a) l'édition précédente de l'IEC TS 61970-401:2005 a donné une vue d'ensemble des spécifications d'interface de composants (CIS - *Component Interface Specifications*) (IEC 61970-402, IEC 61970-403, IEC 61970-404, IEC 61970-405 et IEC 61970-407). Les normes IEC 61970-402 à IEC 61970-407 constituent des duplications d'interfaces OPC existantes de OPC Foundation et des interfaces DAIS/HDA d'OMG. Par conséquent, les normes IEC 61970-402 à IEC 61970-407 ont été supprimées et l'IEC TS 61970-401:2005 n'a plus d'utilité;
- b) l'IEC 61970-401 (le présent document) ne comporte pas de vue d'ensemble des spécifications d'interface de composants (CIS), mais en revanche une description du mode de création des spécifications de profils qui spécifie des contenus d'ensembles de données (ou des contenus de messages). La présente norme a par conséquent été renommée "Cadre de profils". Les spécifications de profils IEC 61970-450 (toutes les parties) et IEC 61970-600 (toutes les parties) décrivent des contenus d'ensembles de données. L'objet du présent document est de définir les règles à suivre dans le processus de création de spécifications de profils.

Le texte de cette Norme internationale est issu des documents suivants:

Projet	Rapport de vote
57/2482/FDIS	57/2494/RVD

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

La langue employée pour l'élaboration de cette Norme internationale est l'anglais.

Ce document a été rédigé selon les Directives ISO/IEC, Partie 2, il a été développé selon les Directives ISO/IEC, Partie 1 et les Directives ISO/IEC, Supplément IEC, disponibles sous www.iec.ch/members_experts/refdocs. Les principaux types de documents développés par l'IEC sont décrits plus en détail sous www.iec.ch/standardsdev/publications.

Une liste de toutes les parties de la série IEC 61970, publiée sous le titre général *Interface de programmation d'application pour système de gestion d'énergie (EMS-API)*, se trouve sur le site web de l'IEC.

Le comité a décidé que le contenu de ce document 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 au document recherché. À cette date, le document sera

- reconduit,
- supprimé,
- remplacé par une édition révisée, ou
- amendé.

IMPORTANT – Le logo "colour inside" qui se trouve sur la page de couverture de ce document 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.

INTRODUCTION

Le présent document constitue une norme de la série IEC 61970 qui définit des interfaces de messagerie pour l'échange de données d'application réseau.

La série de documents IEC 61970-300 spécifie un modèle d'information commun (CIM – *Common Information Model*) canonique qui décrit les données d'application réseau. Le CIM est un modèle d'information qui représente tous les objets principaux d'une entreprise de service public de distribution d'électricité nécessaires pour décrire les données utilisées par les applications de réseaux électriques. Le CIM canonique fournit la sémantique pour les spécifications de profils IEC 61970-450 (toutes les parties) et IEC 61970-600 (toutes les parties) dédiées aux échanges de données spécifiques.

Le présent document décrit le cadre de création des spécifications de profils à partir du CIM canonique. Il décrit également la structure des documents de profils, ainsi que les règles de sélection des informations issues du CIM canonique, à intégrer dans les spécifications de profils.

Les motifs de création du présent document sont les suivants:

- 1) une méthode de profilage non décrite par un document IEC 61970 permet de créer depuis bien longtemps les profils IEC 61970;
- 2) la méthode de profilage IEC 61970 présente des problèmes qu'il est nécessaire de résoudre. Les problèmes et les solutions sont décrits à l'Annexe A.

IECNORM.COM : Click to view the full PDF of IEC 61970-401:2022

INTERFACE DE PROGRAMMATION D'APPLICATION POUR SYSTÈME DE GESTION D'ÉNERGIE (EMS-API) –

Partie 401: Cadre de profils

1 Domaine d'application

Le présent document décrit le mode de structuration et de création des spécifications de profils IEC 61970-450 (toutes les parties) et IEC 61970-600 (toutes les parties). Les spécifications de profils décrivent un sous-ensemble du CIM canonique dédié à un échange de données spécifiques. Le CIM canonique est décrit dans l'IEC 61970-300 (toutes les parties), ainsi que dans l'IEC 61968-11.

Les règles de création ou d'extension du CIM canonique ne relèvent pas du domaine d'application du présent document.

Le présent document définit la structure d'une spécification de profil, ainsi que les règles de sélection des sous-ensembles d'informations à partir du CIM canonique. Il normalise les opérations qui permettent de créer les éléments de profils à partir du CIM canonique. Étant donné que le CIM canonique est décrit en langage UML, les opérations sont décrites en matière de classes, d'attributs et de rôles UML.

Du fait d'une mise en correspondance possible entre le langage UML et le langage RDFS ou OWL, l'un des langages UML, RDFS ou OWL peut donc être utilisé pour décrire les profils créés. La spécification des langages (UML, RDFS ou OWL) utilisés pour décrire les profils, ainsi que le mode de présentation et d'édition des profils dans les interfaces utilisateurs ne relèvent pas du domaine d'application du présent document. Les langages qui servent à décrire les profils sont définis dans d'autres spécifications. Les spécifications pertinentes sont référencées à l'Article 2.

Le langage UML prend en charge l'ajout d'un texte libre qui décrit des restrictions supplémentaires concernant les constructions UML, par exemple, classes, valeurs d'attributs, rôles d'association et cardinalités. Les langages tels que OCL et SHACL sont dédiés à la description des contraintes. Le langage OCL sert à décrire les contraintes relatives aux données d'objets décrites en langage UML, tandis que le langage SHACL sert à décrire les contraintes relatives aux données graphiques décrites par le langage RDFS ou OWL. Le langage OCL relève du domaine d'application du présent document, contrairement au langage SHACL.

Le présent document prend en charge les profils de description des données échangées en tant qu'ensembles de données ou messages CIMXML. Le format d'échange qui relève du domaine d'application est conforme à l'IEC 61970-552, mais d'autres formats sont possibles.

L'interopérabilité des outils et les formats de sérialisation ne relèvent pas du domaine d'application du présent document.

2 Références normatives

Les documents suivants sont cités dans le texte de sorte qu'ils constituent, pour tout ou partie de leur contenu, des exigences du présent document. 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 61968-11, *Intégration d'applications pour les services électriques – Interfaces système pour la gestion de distribution – Partie 11: Extensions du modèle d'information commun (CIM) pour la distribution*

IEC TS 61970-2, *Energy management system application program interface (EMS-API) – Part 2: Glossary* (disponible en anglais seulement)

IEC 61970-300 (toutes les parties), *Energy management system application program interface (EMS-API)* (disponible en anglais seulement)

IEC 61970-501:2006, *Energy management system application program interface (EMS-API) – Part 501: Common Information Model Resource Description Framework (CIM RDF) schema* (disponible en anglais seulement)¹

IEC 61970-552, *Interface de programmation d'application pour système de gestion d'énergie (EMS-API) – Partie 552: Format d'échange de modèle CIMXML*

OMG Unified Modeling Language®, OMG document number: formal/2015-03-01, disponible à l'adresse <http://www.omg.org/spec/UML/2.5>

World Wide Web Consortium (W3C), RDF 1.1 Primer (24 juin 2014), disponible à l'adresse <https://www.w3.org/TR/rdf11-primer/>

World Wide Web Consortium (W3C), RDF 1.1 Concepts and Abstract Syntax (25 février 2014), disponible à l'adresse <https://www.w3.org/TR/rdf11-concepts/>

World Wide Web Consortium (W3C), RDF 1.1 XML Syntax (25 février 2014), disponible à l'adresse <https://www.w3.org/TR/rdf-syntax-grammar/>

World Wide Web Consortium (W3C), RDF Schema 1.1 (25 février 2014), disponible à l'adresse <https://www.w3.org/TR/rdf-schema/>

World Wide Web Consortium (W3C), OWL 2 Web Ontology Language Primer (Second Edition), W3C Recommendation (11 décembre 2012), disponible à l'adresse <https://www.w3.org/TR/owl2-primer/>

World Wide Web Consortium (W3C), OWL 2 Web Ontology Language Structural Specification and Functional-Style Syntax (Second Edition), W3C Recommendation (11 décembre 2012), disponible à l'adresse <https://www.w3.org/TR/owl2-syntax/>

3 Termes, définitions et termes abrégés

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

L'ISO et l'IEC tiennent à jour des bases de données terminologiques destinées à être utilisées en normalisation, consultables aux adresses suivantes:

- IEC Electropedia: disponible à l'adresse <http://www.electropedia.org/>
- ISO Online browsing platform: disponible à l'adresse <http://www.iso.org/obp>

¹ La présente spécification est fondée sur le schéma RDF 1.0 de spécification W3C du début des années 2000 qui a fait l'objet de nombreuses révisions depuis cette période.

3.1 Termes et définitions

3.1.1

CIM canonique

modèle convenu d'information sémantique qui décrit le domaine des réseaux électriques. L'objectif du modèle d'information est de décrire le domaine des réseaux électriques selon un processus normalisé sans recouvrement

Note 1 à l'article: Le CIM canonique est décrit en langage UML. Le langage UML peut être sérialisé (sauvegardé sous forme de document) au format de données normalisé XMI pour un échange entre des outils UML. Ecore est un autre format de sérialisation avec une capacité de description du langage UML.

3.1.2

profil

sous-ensemble issu du CIM canonique qui prend en charge l'échange d'un ensemble spécifique de données dans un contexte spécifique

Note 1 à l'article: Un modèle d'information sémantique tout comme le CIM canonique peut être sérialisé sous plusieurs formats différents:

- XMI qui est le format de sérialisation du langage UML comme décrit par OMG, voir l'Article 2;
- RDF/XML utilisé pour sérialiser le langage RDFS ou OWL;
- RDF/XML comme décrit par l'IEC 61970-501. L'IEC 61970-501 contient plusieurs extensions personnalisées selon la spécification RDFS de 1999;
- Ecore pour l'échange avec le cadre de modélisation Eclipse (EMF - *Eclipse Modelling Framework*).

Note 2 à l'article: La normalisation de la sérialisation des profils IEC 61970 observe les principes de l'IEC 61970-501. Un ensemble de profils prend en charge un cas d'utilisation spécifique.

3.1.3

profilage

processus de création de profils

3.1.4

ensemble de données

ensemble de données décrit par un profil signifiant que les données dudit ensemble s'alignent sur les informations décrites par le profil

Note 1 à l'article: Un profil ne traite pas du format de données.

3.1.5

format de données

ensembles de données qui contiennent les données décrites par des profils tels que définis dans le présent document, qui utilisent le langage CIMXML décrit dans l'IEC 61970-552 et le langage RDF/ XML fondé sur le schéma RDFS comme décrit dans l'IEC 61970-501

3.2 Termes abrégés

Les termes abrégés sont indiqués dans le Tableau 1.

Tableau 1 – Termes abrégés

Terme abrégé	Description
CIMXML	format de données RDF/XML décrit dans la spécification IEC 61970-552.
RDF/XML	format de sérialisation de RDF et des données graphiques. Pour des informations plus détaillées, se reporter au document W3C specification RDF 1.1 XML Syntax.
RDF	cadre de description des ressources qui constitue un modèle d'information pour les données graphiques, composé: – d'une ressource (dans le format CIMXML, il s'agit d'un identificateur d'objet); – d'un prédicat (dans le format CIMXML, il s'agit d'un nom d'attribut ou de rôle); – d'un élément de données (dans le format CIMXML, il s'agit d'une valeur ou d'une référence de ressource). Pour une introduction au cadre RDF, se reporter au document W3C specification RDF 1.1 Primer.
RDFS	schéma RDF qui constitue une extension du cadre RDF avec une capacité de description d'un vocabulaire dédié à la modélisation des données. Par conséquent, il s'agit d'un langage de modélisation de l'information avec toutefois une capacité inférieure à celle du langage UML. Pour des informations plus détaillées, se reporter au document W3C specification RDF Schema 1.1.
OWL	langage d'ontologie qui prend en charge la toile sémantique. Il constitue un développement approfondi et une extension du schéma RDFS. Pour une introduction au langage OWL, se reporter au document W3C recommendation OWL 2 Web Ontology Language Primer (Second Edition).
UML	langage de modélisation unifié Pour des informations plus détaillées, se reporter au document OMG specification formal/2015-03-01.

4 Vue d'ensemble

4.1 Profils et profilage

Le profilage est le processus de sélection d'informations du CIM canonique et d'intégration desdites informations dans des profils.

Le CIM canonique est décrit dans l'IEC 61970-300 (toutes les parties) et dans l'IEC 61968-11.

Le principe directif de la méthode de profilage consiste en ce que les informations décrites par un profil constituent un sous-ensemble vrai du CIM canonique et conservent les noms de classes, de rôles et d'attributs, ainsi que les types de données dudit CIM. Les types de données selon le modèle CIM sont décrits par les classes stéréotypées Primitive, CIMDatatype et Compound. CIMDatatype est une composition de trois attributs (valeur, unité et multiplicateur), tandis que Compound est composée d'un ensemble arbitraire d'attributs.

L'objectif principal consiste en ce que les différents ensembles de données (voir l'Article 3) échangés au moyen de profils différents fondés sur le CIM canonique reposent seulement sur les définitions et les principes de base dudit CIM. Il s'agit d'un élément clé de la faisabilité de l'interopérabilité. Cette disposition permet également à des profils différents de lier les données entre elles au moyen du CIM canonique comme un concentrateur. Elle permet aussi au lecteur d'un ensemble de données ou d'un message d'identifier facilement les descriptions d'éléments tant dans les profils que dans le CIM canonique. La prise en charge de la liaison des données dans différents ensembles de données ou différents messages décrits par des profils différents est nécessaire lorsque les données sont réparties entre différents ensembles de données régis par des profils différents. Ces cas d'utilisation sont définis pour les modèles de réseaux qui dissocient la description du réseau de ses conditions de fonctionnement (considérées comme un élément d'entrée) et des résultats.

Il existe plusieurs langages qui peuvent décrire les profils, par exemple, UML (sérialisé en tant que langage XMI), RDFS, Ecore ou OWL. Le langage UML inclut un langage graphique mis en œuvre par les responsables d'édition UML. Le langage OWL n'est pas un langage graphique, mais il existe plusieurs responsables d'édition qui prennent en charge l'affichage et l'édition des données OWL.

Les spécifications de profils actuelles, IEC 61970-450 (toutes les parties) et IEC 61970-600 (toutes les parties) utilisent le langage UML pour décrire les informations et le langage OCL pour réduire davantage ces informations. Un profil en langage UML est décrit par des classes, des attributs, des associations, des rôles et des types de données, qui représentent le mode commun de description des informations dans ce langage. Le langage UML inclut le concept de stéréotypes et de valeurs étiquetées qui permet des extensions personnalisées de ce langage. Par conséquent, le profilage avec le langage UML désigne la reproduction et la mise à jour de classes, d'attributs, d'associations, de stéréotypes et de valeurs étiquetées à partir du CIM canonique. Un profil en langage RDFS ou OWL est décrit par des classes et des propriétés, mais les stéréotypes UML et les valeurs étiquetées n'ont pas une mise en correspondance directe avec les langages RDFS/OWL. Le profilage avec le langage OWL désigne la création de classes et de propriétés OWL par un mode de sélection de classes, d'attributs et de rôles UML à partir du CIM canonique qui est identique au mode de profilage avec le langage UML. Les stéréotypes UML ou les valeurs étiquetées nécessitent une mise en correspondance spécifique selon leur signification. L'Article A.1 présente une mise en correspondance du langage UML CIM canonique et du langage OWL.

La Figure 1 donne une vue d'ensemble des spécifications IEC 61970 dans le contexte du profilage.

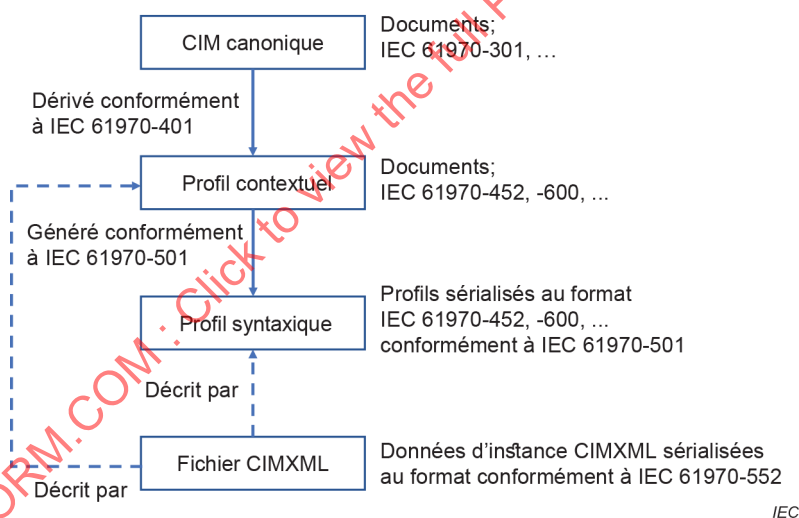


Figure 1 – Relations entre normes et profilage

Le CIM canonique est présenté en haut de la Figure 1. Un profil contextuel est créé sur la base du CIM canonique et d'un contexte d'utilisation donné. Un profil syntaxique constitue une sérialisation du profil contextuel. Le format de sérialisation des profils contextuels est décrit par l'IEC 61970-501. Le profil IEC 61970-501 est le seul profil syntaxique fondé sur un graphique. Un autre profil syntaxique existe. Par exemple, un outil UML peut sérialiser le profil au format XMI, un outil fondé sur le cadre de modélisation Eclipse (EMF) peut sérialiser un profil au format Ecore et un outil OWL peut utiliser le langage RDF/XML. Tous ces profils syntaxiques décrivent le même profil contextuel, mais dans un format spécifique à la sérialisation choisie (UML/XMI, Ecore, RDFS, OWL, etc.). Chaque profil syntaxique potentiel a sa propre spécification, par exemple, comme le profil IEC 61970-501.

La sérialisation des ensembles de données au format CIMXML s'effectue conformément à l'IEC 61970-552 et les données dans un ensemble de données CIMXML sont décrites par un profil. Le profil contextuel et le profil syntaxique décrivent les mêmes informations, à la seule différence que le profil syntaxique est lisible par une machine et peut être utilisé dans le traitement ou la validation des ensembles de données CIMXML.

Il est nécessaire que les outils, qui traitent des données décrites par les profils créés selon le présent document, disposent d'une sérialisation des profils lisible par une machine, appelée profil syntaxique. L'IEC 61970-501 constitue une sérialisation fondée sur le langage RDFS destinée à l'objectif susmentionné. Les outils de profilage doivent prendre en charge la génération de profils au format de sérialisation de l'IEC 61970-501. Tout outil, indépendamment du fait qu'il soit fondé sur le langage UML ou OWL, doit sérialiser les profils dans un profil syntaxique utilisé pour un échange avec d'autres outils ou sauvegardé à des fins de pérennité. L'IEC 61970-501 ne comporte pas toutes les informations nécessaires pour les outils de profilage, par exemple, il est possible de produire un profil syntaxique IEC 61970-501 à partir d'une représentation UML, mais la reproduction du langage UML à partir d'un profil au format IEC 61970-501 n'est en revanche pas possible. Par conséquent, l'IEC 61970-501 ne peut pas être utilisée comme format d'échange entre les outils de profilage fondés sur le langage UML.

De nombreuses sources proposent des vocabulaires normalisés, qui peuvent être utilisés outre le CIM canonique. Il s'agit par exemple des sources suivantes:

- the Dublin Core™ de la Dublin Core Metadata initiative, du 18 décembre 2006;
- the Profiles Vocabulary (DX-PROF) de W3C, note du 18 décembre 2019;
- profile Guidance de W3C, projet du 24 novembre 2021.

Le mode selon lequel ces vocabulaires peuvent être utilisés avec le CIM canonique n'est pas décrit et exige un développement supplémentaire.

4.2 Relations entre CIM canonique, profils et ensembles de données

Les spécifications CIM peuvent être réparties en deux groupes:

- CIM canonique;
- profil dérivé du CIM canonique.

Le CIM canonique est un modèle d'information pour le domaine des réseaux électriques, normalisé par l'IEC. Il est par ailleurs destiné à être normalisé (description unique des informations) sans reproduire les définitions du même concept. Le CIM canonique est destiné à être un modèle d'information pour les concepts qui relèvent d'un domaine et non pas une description des données échangées par messages.

Un profil est un sous-ensemble du CIM canonique destiné à un échange de données spécifique dans un contexte spécifique. Par conséquent, un profil est un modèle d'information pour les données échangées. Les informations du CIM canonique sont réparties en profils qui décrivent l'échange de données dans des contextes différents. Exemples de ce type de profils: données sur les équipements relativement stables avec une faible fréquence de variation et un état de réseau électrique dont les variations dans le temps sont comprises entre des millisecondes et plusieurs heures selon le cas d'utilisation. Le contexte de génération des données sur les équipements prend généralement la forme d'un outil de modélisation des données qui assure le maintien desdites données, tandis que le contexte de génération de l'état des réseaux électriques peut prendre la forme de systèmes en temps réel (WAMS, SCADA) ou de systèmes de planification opérationnelle.

Les normes IEC 61970 (toutes les parties) prennent en charge les fonctions analytiques utilisées dans les études des réseaux électriques. Un cas d'étude est constitué de plusieurs ensembles de données décrits par des profils différents; par exemple, l'exécution d'un flux de puissance nécessite à la fois des données sur les équipements et l'état du réseau électrique. Par conséquent, les profils IEC 61970 dépendent les uns des autres et les cas d'utilisation impliquent généralement plusieurs ensembles de données décrits par des profils différents.

Le présent paragraphe (4.2) explique la relation entre les informations traitées ci-dessus et celles traitées en 4.1. La Figure 2 représente les dépendances.

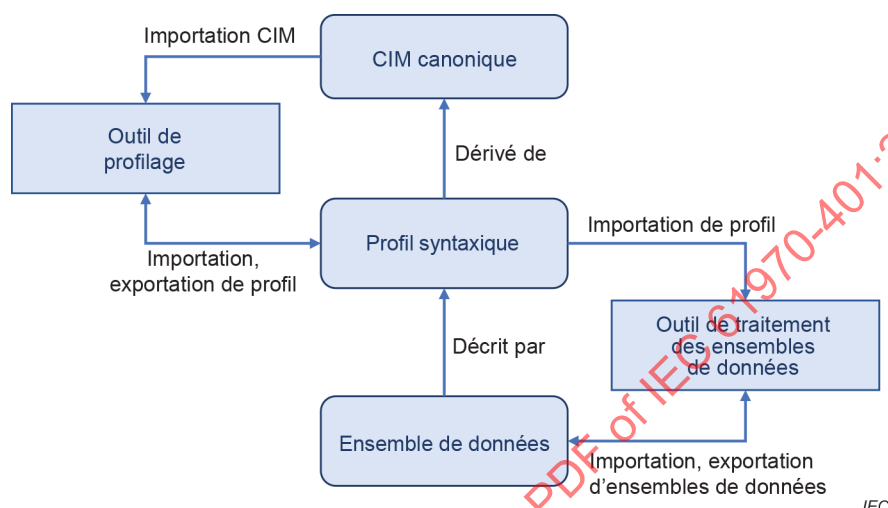


Figure 2 – Relations entre CIM canonique, profils et ensembles de données

La Figure 2 décrit les relations entre le CIM canonique, les profils et les ensembles de données. Le CIM canonique en haut de la Figure 2 est décrit dans l'IEC 61970-300 (toutes les parties) et dans l'IEC 61968-11. Le processus de profilage décrit dans le présent document est représenté par la flèche "Dérivé de" ("Derived from") et l'"Outil de profilage" ("Profiling Tool") qui permet de créer les "Profils syntaxiques" ("Syntactic Profiles"). L'IEC 61970-450 (toutes les parties) et l'IEC 61970-600 (toutes les parties) constituent des exemples de profils créés au moyen du processus de profilage.

Les ensembles de données (bas de la Figure 2) sont généralement échangés sous la forme de fichiers CIMXML, mais d'autres formats de données sont possibles s'ils sont conformes aux profils générés selon le présent document. Un ensemble de données est décrit par un profil "Décrit par" ("Described by" à la Figure 2) et est sérialisé sous la forme de fichiers CIMXML comme cela est décrit dans l'IEC 61970-552.

L'"outil de traitement des ensembles de données" ("Dataset Tool") traite les "Ensembles de données" ("Datasets"), par exemple, opérations d'exportation, d'importation, de validation, etc. Un "Outil de traitement des ensembles de données" peut être codé en dur par rapport à un ensemble de profils, mais peut également importer des profils et les utiliser pour tout traitement tel qu'indiqué à la Figure 2 par la flèche "Importation de profil" ("Profile Import"). Le fonctionnement de ce processus nécessite une version de profil lisible par une machine, comme un profil syntaxique au format RDFS tel que décrit dans l'IEC 61970-501. Des sérialisations d'autres langages de modélisation de l'information sont également possibles, par exemple, XMI, OWL ou Ecore, mais l'IEC 61970 utilise des techniques fondées sur le format RDF telles que définies par l'IEC 61970-501. Le terme "architecture guidée par modèle" désigne parfois la capacité d'un "Outil de traitement des données" à importer et utiliser une version de profil lisible par une machine ("Importation de profil" à la Figure 2), dans la mesure où l'outil s'adapte aux profils. Pour prendre en charge l'utilisation d'un "Outil de traitement des ensembles de données" guidé par modèle, l'"Outil de profilage" doit importer et exporter les profils syntaxiques au format IEC 61970-501. Exemples de cas d'utilisation de profils syntaxiques:

- outils de profilage qui lisent, actualisent et sauvegardent des profils;
- outils de validation qui lisent les fichiers CIMXML et vérifient qu'ils sont conformes aux profils syntaxiques;
- outils de gestion des données qui importent et exportent les fichiers CIMXML;
- fonctions analytiques qui importent les données d'entrée et exportent les données de résultat.

Le langage de profil syntaxique défini dans l'IEC 61970-501:2006 comporte des extensions personnalisées du format RDFS du fait de la finalisation non encore effective de la norme RDFS au moment de la publication de l'IEC 61970-501. L'évolution et l'extension du schéma RDFS permettent de remplacer les extensions personnalisées spécifiées dans l'IEC 61970-501:2006. Noter également que l'IEC 61970-501:2006 ne fournit pas d'informations suffisantes pour permettre l'interopérabilité des outils. Il est nécessaire de prendre en considération d'autres formats à cet égard, par exemple, le format XMI normalisé par le langage OMG et le format OWL normalisé par W3C.

4.3 Profils et processus métier

L'échange des ensembles de données s'effectue dans le contexte d'un processus métier qui signifie généralement plusieurs ensembles de données définis par des profils différents. Le CIM canonique est généralement réparti dans les différents profils. Par conséquent, il est nécessaire d'assembler les ensembles de données dans un modèle complet dans lequel chaque ensemble de données fournit des données selon son profil de définition.

L'IEC 61970-450 (toutes les parties) et l'IEC 61970-600 (toutes les parties) décrivent les données utilisées dans les calculs analytiques (par exemple, calculs du flux de puissance, analyse de sécurité, calculs des courts-circuits). Les entrées de ces calculs sont des modèles qui comprennent un ensemble complet de données nécessaires pour résoudre le calcul analytique. Ce processus est courant pour répartir les données en groupes. Le profilage prend en charge le regroupement par une répartition des informations en plusieurs profils, chaque profil décrivant un groupe. La création de groupes de données peut être motivée par des origines et des taux de variation différents. Les équipements de réseau électrique sont stables dans la durée et ne sont modifiés que lorsque de nouveaux équipements sont ajoutés ou que d'anciens équipements sont retirés. Ainsi, les données qui décrivent les équipements de réseau électrique ont un faible taux de variation et proviennent généralement d'outils de gestion de données dédiés. L'état du réseau électrique peut varier tous les mois pour la planification à long terme, toutes les heures pour la planification opérationnelle, toutes les secondes pour l'exploitation en temps réel et à la milliseconde près pour les études du comportement dynamique. Par conséquent, les données relatives à l'état du réseau électrique bénéficient d'un échange distinct des données relatives aux équipements. Il en résulte que les données d'un modèle utilisé par les fonctions analytiques sont réparties en profils qui prennent en charge l'échange de parties du modèle sous forme de fichiers CIMXML.

Par conséquent, l'exécution d'un calcul analytique avec des fichiers CIMXML en entrée exige un processus d'assemblage dans lequel les fichiers CIMXML sont assemblés en un modèle complet. Le processus de profilage doit ensuite assurer que les profils décrivent les relations non seulement à l'intérieur d'un fichier CIMXML, mais également entre les données de différents fichiers CIMXML.

Le caractère exhaustif et valide d'un modèle assemblé est déterminé dans le cadre d'un processus de validation dans lequel tous les profils définis pour le processus métier sont utilisés. Une validation par profil est dans nombre de cas non significative, car les informations réparties dans des profils différents ne peuvent pas être vérifiées avant l'assemblage du modèle complet.

5 Structure des documents de profils

Les documents de profils IEC 61970-450 et IEC 61970-600 doivent suivre strictement une structure commune qui comporte les articles suivants:

- 1) "domaine d'application" qui décrit l'application du document;
- 2) "références normatives" qui contiennent une liste de publications auxquelles il est fait référence dans le texte de sorte qu'elles constituent, pour tout ou partie de leur contenu, les exigences du présent document;
- 3) "Termes, définitions et termes abrégés" utilisés dans le profil;
- 4) "spécification du profil d'application" qui précise les éléments suivants
 - a) Une référence au modèle d'information canonique sous forme de nom de fichier XMI duquel le profil est dérivé.
 - b) Une liste des profils définis dans le document. Chaque définition de profil comprend les éléments suivants:
 - i) nom, par exemple, StateVariables (SV);
 - ii) version, par exemple, 4;
 - iii) date de la version;
 - iv) URI qui identifie de manière unique le profil, y compris sa version, par exemple <http://iec.ch/TC57/2013/61970-456/StateVariables/4>.
 - c) exigences;
 - d) contraintes.
- 5) "Vue d'ensemble" qui décrit le contexte dans lequel ce profil est destiné à être utilisé, ainsi que la relation avec les autres profils. Les cas d'utilisation pour lesquels le profil est applicable sont également décrits;
- 6) les cas d'utilisation qui décrivent les situations d'application du profil;
- 7) les sections dédiées aux profils qui décrivent les informations internes aux profils.

6 Cas d'utilisation de profilage

6.1 Vue d'ensemble

Le présent paragraphe 6.1 décrit les cas d'utilisation propres au mode de création des profils à partir du CIM canonique. Le présent paragraphe constitue une vue d'ensemble descriptive. L'utilisation normative est décrite à l'Article 7.

Le profilage est le processus de sélection des classes, des attributs, des rôles et des types de données UML afin de les intégrer dans un profil. Ce processus suit les règles décrites à l'Article 7.

Dans le processus de profilage, un sous-ensemble des attributs et des rôles d'une classe au format CIM canonique est intégré dans un profil. La modification des cardinalités des attributs et des rôles est alors possible. Lorsque de nouvelles informations qui ne sont pas au format CIM canonique sont nécessaires, les classes existantes peuvent être étendues par l'inclusion de plus d'attributs ou plus de rôles, ainsi que par l'ajout de nouvelles classes au CIM canonique. Lors de l'ajout de nouveaux attributs, de nouveaux types de données peuvent également être nécessaires. Ils sont alors ajoutés en premier dans le CIM canonique.

Les cas d'utilisation de profilage qui ont été identifiés peuvent se présenter comme suit.

- a) Ils intègrent différents ensembles d'attributs et de rôles de la même classe dans des profils différents, voir 6.2.
- b) Ils intègrent des ensembles de recouvrement d'attributs et de rôles de la même classe dans des profils différents, voir 6.3.
- c) Ils intègrent une classe de base avec différents ensembles d'attributs ou de rôles, voir 6.4.
- d) Ils étendent une classe existante, voir 6.5.
- e) Ils étendent une classe existante utilisée de manière différente dans des profils différents, voir 6.6.
- f) Ils comportent des cardinalités différentes, voir 6.7.
- g) Ils ajoutent un nouveau type de données, voir 6.8.

Les cas de profilage susmentionnés se limitent aux cas d'utilisation liés aux profils IEC 61970-450 et à la capacité de modélisation du langage UML. Le profilage nécessite souvent l'application de contraintes supplémentaires qui s'étendent au-delà de la capacité du langage UML. Ces contraintes peuvent être les suivantes

- Contrainte de l'espace de valeurs d'un attribut. Il peut s'agir de contraintes de valeurs numériques, de listes d'énumération, de contraintes de longueur et de contenu de chaînes, de comparaisons complexes qui incluent des attributs de classes connexes, etc.
- Contrainte des cardinalités des rôles pour les sous-classes qui héritent des associations d'une classe plus générique.
- Contrainte de l'occurrence d'instances de classes selon le contexte.

Des contraintes de ce type peuvent être décrites par les langages formels OCL ou SHACL. Étant donné que les contraintes OCL ou SHACL ne sont pas incluses dans le CIM canonique, elles ne font pas partie intégrante du processus de profilage, mais sont ajoutées aux profils après réalisation du profilage.

Les paragraphes 6.2 à 6.8 contiennent des exemples de profils existants, par exemple:

- hypothèse en régime établi (SSH - *Steady State Hypothesis*) (IEC 61970-456 et IEC 61970-600);
- équipements (EQ) (IEC 61970-452 et IEC 61970-600).

6.2 Classe dans des profils différents sans recouvrement

Il s'agit du cas d'utilisation le plus courant. Un sous-ensemble de non-recouvrement des attributs et des rôles d'une classe donnée est intégré dans un ou plusieurs profils, voir la Figure 3.

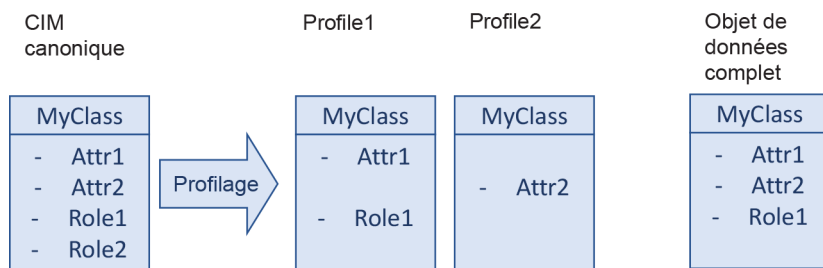


Figure 3 – Classe dans des profils différents sans recouvrement

Un exemple de MyClass à la Figure 3 peut consister en toute classe CIM avec des données d'injection du flux de puissance dans le profil SSH et des données statiques dans le profil EQ, par exemple, cim:EnergyConsumer.

Une application de flux de puissance doit alors collecter les données des deux ensembles de données des profils EQ et SSH afin d'établir le cas d'utilisation du flux de puissance. Un objet complet comportant à la fois les données EQ et les données SSH peut alors être construit et décrit par l'"Objet de données complet" ("Complete data object") comme cela est indiqué à la Figure 3. Noter qu'il convient également de prendre en considération l'héritage, même s'il n'est pas présenté à la Figure 3 pour des raisons de simplicité.

6.3 Inclusion d'ensembles de recouvrement d'attributs et de rôles de la même classe dans des profils différents

Il s'agit d'un cas d'utilisation peu courant rencontré dans quelques cas. La présence du même attribut ou du même rôle dans deux profils différents peut être source d'ambiguïté concernant les données à utiliser dans les ensembles de données pour les différents profils. Les attributs des profils peuvent avoir une signification légèrement différente (par exemple, une valeur par défaut dans un profil et une valeur actuelle dans un autre profil). Ce mode de profilage doit être évité, car il est susceptible d'engendrer une certaine confusion. La signification définie dans le CIM canonique ne doit pas être modifiée.

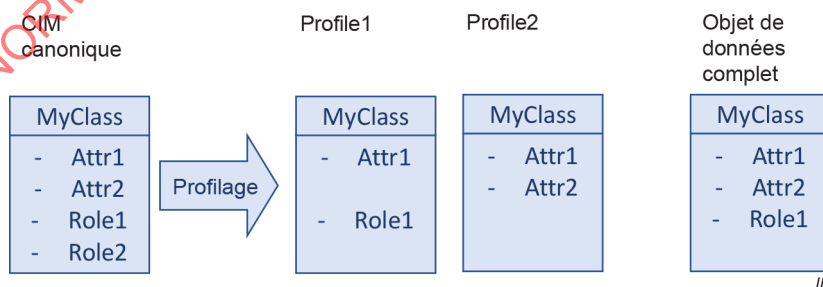
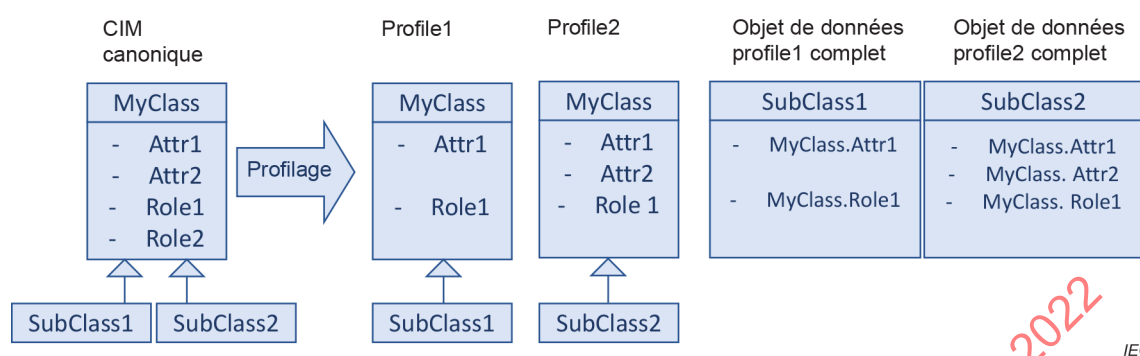


Figure 4 – Classe dans des profils différents avec recouvrement

La Figure 4 décrit le même cas que la Figure 3 à la seule différence que le même attribut "Attr1" est intégré dans deux profils ou plus. cim:OperationalLimits constitue un exemple dans lequel les valeurs limites apparaissent à la fois dans le profil Equipment (Équipements) (IEC 61970-452:2021) et dans les profils d'état de réseaux électriques résolus (IEC 61970-456:2018).

6.4 Inclusion d'une classe de base avec différents ensembles d'attributs ou de rôles

Il s'agit d'un cas d'inclusion différent de la même classe dans des profils différents.



IEC

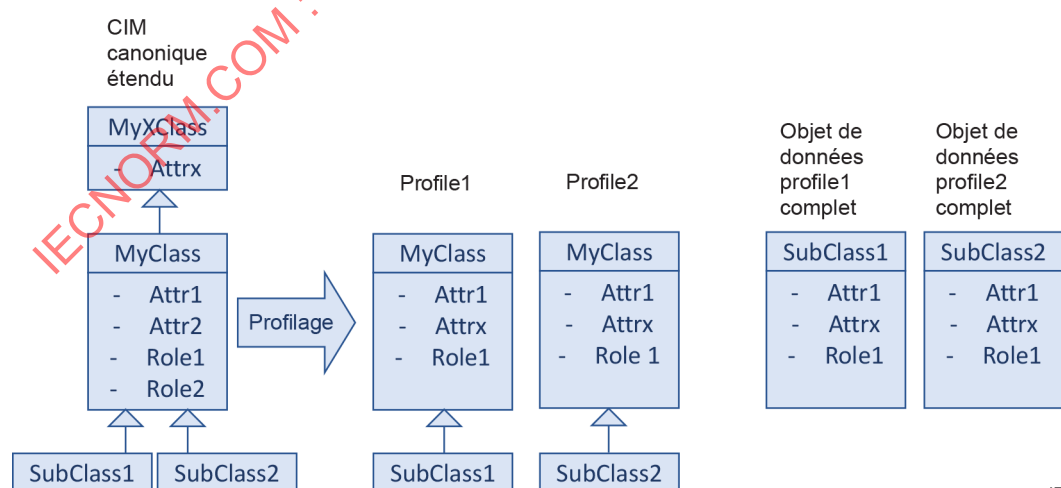
Figure 5 – Classe de base identique avec différents ensembles d'attributs ou de rôles

La Figure 5 présente un cas dans lequel "MyClass" est la classe de base par rapport à deux autres classes "SubClass1" et "SubClass2". Le profilage place "SubClass1" dans "Profile1" et "SubClass2" dans "Profile2". Les attributs et les rôles de "MyClass" intégrés dans les profils ne sont pas exactement les mêmes. Exemple: "MyClass" correspond à `cim:IdentifiedObject` avec intégration de `cim:IdentifiedObject.name` et de `cim:IdentifiedObject.description` dans "Profile2", mais intégration uniquement de `cim:IdentifiedObject.name` dans "Profile1". "SubClass2" peut correspondre à toute classe dans le profil EQ et "SubClass1" peut correspondre à `cim:TopologicalNode` dans le profil TP.

6.5 Extension d'une classe existante

L'extension d'une classe s'effectue avec l'exigence que ladite extension soit identique pour tous les profils. Les règles d'extension du CIM canonique ne relèvent pas du domaine d'application du présent document.

Dans l'étape de profilage, les attributs et les rôles d'extension sont transférés dans une version de profil de la classe comme cela est présenté à la Figure 6.



IEC

Figure 6 – Extension identique d'une classe dans tous les profils applicables

La Figure 6 présente le mode d'extension de "MyClass" avec un attribut "Attrx" hérité de "MyXClass". "Attrx" est intégré dans la classe "MyClass" pour les profils qui intègrent ladite classe. Exemple de ce type de cas: extension de cim:IdentifiedObject avec un nouvel attribut (par exemple, shortName) utilisé dans tous les profils qui intègrent cim:IdentifiedObject, par exemple, la plupart des classes dans le profil EQ et cim:TopologicalNode dans le profil TP.

6.6 Extension d'une classe existante utilisée de manière différente dans des profils différents

L'extension d'une classe s'effectue avec l'exigence que ladite extension soit identique pour tous les profils avec lesquels elle est utilisée. Il s'agit du même cas que celui présenté en 6.5, avec toutefois une intégration différente de l'extension dans des profils différents.

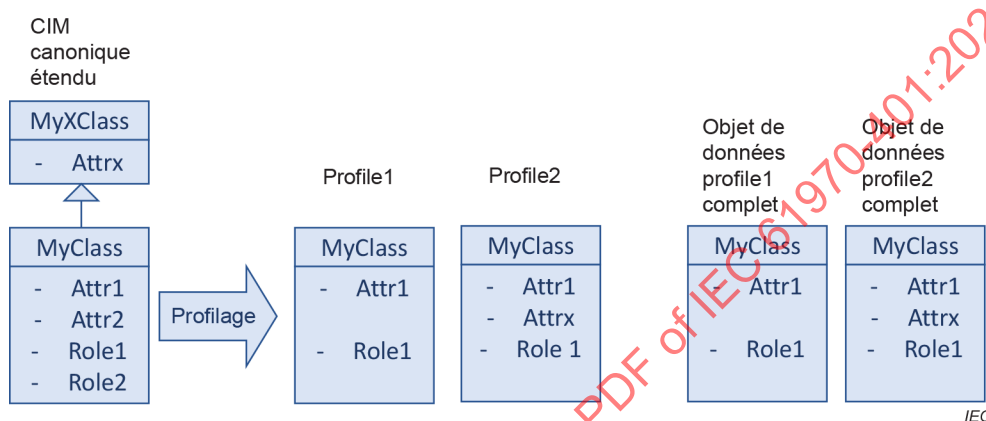


Figure 7 – Extension d'une classe existante utilisée de manière différente dans des profils différents

Dans la Figure 7, "MyClass" comporte l'extension "Attrx" dans "Profile2", mais non dans "Profile1". Il s'agit du même cas d'utilisation que celui présenté en 6.4, mais dédié aux extensions.

Exemple: extension de cim:ConnectivityNode et de cim:TopologicalNode avec utilisation des données supplémentaires dans les profils frontières qui ne modifient pas les versions non frontières de cim:ConnectivityNode et de cim:TopologicalNode.

6.7 Cardinalités différentes

En cas d'intégration d'une classe dans des profils différents comme en 6.4 et 6.6, il est possible d'utiliser des cardinalités différentes pour le même attribut dans les différents profils, ce qui signifie que les classes profilées ont le même nom de classe, mais des cardinalités d'attributs ou de rôles différentes.

Il s'agit d'une violation de la conception orientée objet étant donné qu'une classe qui porte le même nom est définie de manière différente dans des profils différents. Exemple: cim:IdentifiedObject.name est exigé dans le profil Equipment, mais non dans le profil Topology (Topologie) comme tel est le cas dans l'IEC TS 61970-600. Les conceptions orientées objet exigent que des classes différentes aient des noms différents.

Comme traité en 6.1, les profils de la série IEC 61970-450 décrivent des documents CIMXML assemblés dans des modèles utilisés par des calculs analytiques. Si des instances qui portent le même nom de classe, mais avec des cardinalités d'attribut ou de rôle différentes, apparaissent dans le modèle, des noms de classe différents sont alors exigés pour distinguer les différences de cardinalité. Comme tel n'est pas le cas, il est nécessaire que les ingénieurs d'application fassent des efforts supplémentaires pour contourner ce problème, par exemple par l'utilisation de noms de classe différents dans la mise en œuvre ou par l'application d'une technique de décoration.

6.8 Ajout d'un nouveau type de données

Lors de l'ajout d'attributs, il peut apparaître que les types de données existants ne couvrent pas les besoins. Il est par conséquent nécessaire d'ajouter un nouveau type de données.

7 Règles de profilage

7.1 Vue d'ensemble

Le présent Article 7 intègre les règles qui doivent être suivies pour la création des profils IEC 61970. Les articles précédents indiquent le contexte d'établissement des règles.

7.2 R0101 Noms de classe, d'attribut, de rôle et de type de données d'un modèle d'information

La modification des noms de classe, d'attribut, de rôle ou de type de données n'est pas admise lors du profilage. La raison de ce défaut d'autorisation est de permettre un traçage direct des entités profilées jusqu'au CIM canonique, ce qui permet l'interopérabilité entre les profils créés. Il s'agit également d'une condition préalable à l'assemblage efficace de modèles (voir 6.1), car elle n'exige aucune mise en correspondance des informations contenues dans les différents profils.

7.3 Règles applicables aux classes sans stéréotype

7.3.1 R0201 Intégration d'une classe

Seules les classes du CIM canonique ou des extensions fondées sur le CIM canonique peuvent être intégrées dans un profil.

7.3.2 R0202 Ajout d'une nouvelle classe

Lorsqu'une nouvelle classe se révèle nécessaire, le CIM canonique doit être étendu avec la classe comme cela est décrit à l'Article 8. Après cette extension, la nouvelle classe est intégrée comme cela est décrit en 6.5.

7.3.3 R0203 Description d'une classe

La description de la classe dans le CIM canonique ou dans les extensions fondées sur le CIM canonique doit être incluse non modifiée et aucun ajout n'est permis.

7.3.4 R0204 Nom d'une classe

La modification du nom d'une classe incluse dans un profil n'est pas permise.

7.3.5 R0205 Cardinalité d'une classe

Il doit être possible de spécifier la cardinalité d'une classe, c'est-à-dire le nombre d'instances qui peuvent apparaître pour une classe.

Les exemples de spécification utile de la cardinalité d'une classe sont les suivants.

- Au moins un `cim:ConnectivityNode` pour les modèles construits avec `cim:ConnectivityNodes`.
- Un `cim:ControlArea` exactement est exigé pour un modèle de réseau TSO.
- Au moins un `cim:TopologicallyIsland` est exigé pour un cas de flux de puissance résolu.

7.3.6 R0206 Classe concrète

Il est possible de spécifier si une classe est autorisée à comporter des instances, c'est-à-dire que la classe est concrète. Une classe qui n'est pas autorisée à comporter des instances doit être déclarée abstraite (attribut de métadonnées UML).

7.4 Règles applicables aux attributs

7.4.1 R0301 Intégration d'un attribut

Pour le langage UML, un attribut d'une classe du CIM canonique ou des extensions fondées sur le CIM canonique peut être intégré lorsque la classe qui comporte l'attribut est déjà intégrée dans le profil.

7.4.2 R0302 Ajout d'un nouvel attribut

Lorsqu'un nouvel attribut se révèle nécessaire pour une classe déjà existante dans le CIM canonique, la classe doit être étendue avec le nouvel attribut avant de l'intégrer dans le profil comme cela est décrit à l'Article 8.

7.4.3 R0303 Description d'un attribut

La description d'un attribut dans le CIM canonique ou dans les extensions avec des attributs supplémentaires fondés sur le CIM canonique doit être incluse non modifiée.

7.4.4 R0304 Nom d'un attribut

La modification du nom d'un attribut intégré dans un profil n'est pas permise.

7.4.5 R0305 Cardinalité d'un attribut

La cardinalité d'un attribut peut être rendue plus restrictive, c'est-à-dire qu'elle peut être modifiée de 0..1 à 1..1. Le profil ne doit pas rendre la cardinalité moins restrictive comme le définit le CIM canonique ou le CIM canonique étendu.

7.4.6 R0306 Type de données d'un attribut

Les types de données sont définis dans le CIM canonique ou le CIM canonique étendu. Les types de données sont classés en différents types au moyen d'une classe de stéréotype telle que Primitive, CIMDatatype, Compound ou Enumeration (Énumération).

Le type de données d'un attribut dans le CIM canonique ou des extensions fondées sur le CIM canonique ne doit pas être modifié dans un profil.

7.4.7 R0307 Valeur initiale d'un attribut

Lorsqu'un attribut a une valeur initiale dans le CIM canonique ou les extensions fondées sur le CIM canonique, il ne doit pas être annulé dans le profil.

Lorsqu'un attribut n'a pas de valeur initiale dans le CIM canonique ou les extensions fondées sur le CIM canonique, il peut être spécifié dans le profil.

7.5 Types de données

7.5.1 R0401 Intégration d'un type de données

Un type de données utilisé par un attribut dans un profil doit être intégré dans le profil.

7.5.2 R0402 Ajout d'un nouveau type de données

Lorsqu'un nouveau type de données se révèle nécessaire, le CIM canonique doit être étendu avec ce dernier avant de l'intégrer dans le profil.

7.5.3 R0403 Type de données Primitive

Un type de données Primitive utilisé dans un profil doit être intégré non modifié.

L'espace de valeurs d'un type de données Primitive existant peut être davantage restreint par l'utilisation de contraintes supplémentaires décrites dans OCL ou SHACL, par exemple, les valeurs de chaîne peuvent être restreintes par des expressions régulières ou des listes de valeurs.

7.5.4 R0405 CIMDatatype

Une classe de stéréotype CIMDatatype comporte trois attributs, c'est-à-dire valeur, unité et multiplicateur. Un profil peut établir que l'unité et le multiplicateur sont déterminés pour lui-même ou pour le contexte de l'échange. Ces attributs sont donc identiques pour toutes les instances d'une classe de stéréotype CIMDatatype particulière. Lorsque tel est le cas, les attributs (valeur, unité et multiplicateur) sont supprimés et la classe de stéréotype CIMDatatype se voit attribuer le type de données de l'attribut valeur de cette même classe.

Par conséquent, l'intégration d'une classe de stéréotype CIMDatatype dans un profil peut s'effectuer de deux manières différentes.

- a) L'aplatissement de la classe de stéréotype CIMDatatype s'effectue par l'exclusion des attributs (valeur, unité et multiplicateur) et compte tenu du type de données issu de la valeur de l'attribut valeur dans la classe de stéréotype CIMDatatype.
- b) Les attributs (valeur, unité et multiplicateur) de la classe de stéréotype CIMDatatype sont intégrés en l'état.

La méthode de profilage utilisée parmi les méthodes décrites en a) et b) est déterminée par profil ou par contexte pour l'échange de données.

L'espace de valeurs d'un CIMDatatype existant peut être davantage restreint par l'utilisation de contraintes supplémentaires décrites dans OCL ou SHACL.

7.5.5 R0405 Type de données Compound

Un type de données Compound suit les mêmes règles qu'une classe.

7.5.6 R0406 Enumeration

Un type de données Enumeration utilisé dans un profil doit être intégré non modifié afin d'éviter les problèmes d'interopérabilité entre les profils au moyen de sous-ensembles différents de l'énumération dans le CIM canonique.

7.5.7 R0407 Description d'un type de données

La description d'un type de données dans le CIM canonique ou dans les extensions fondées sur le CIM canonique doit être incluse non modifiée.

7.5.8 R0408 Nom d'un type de données

La modification du nom d'un type de données intégré dans un profil n'est pas permise.