European Network of
Transmission System Operators
for Electricity

METADATA AND DOCUMENT HEADER DATA EXCHANGE SPECIFICATION

2021-04-21

SOC APPROVED
VERSION 1.0

Copyright notice:

This document is maintained by the ENTSO-E CIM EG. Comments or remarks are to be provided at cim@entsoe.eu

NOTE CONCERNING WORDING USED IN THIS DOCUMENT

The force of the following words is modified by the requirement level of the document in which they are used.

- SHALL: This word, or the terms “REQUIRED” or “MUST”, means that the definition is an absolute requirement of the specification.
- SHALL NOT: This phrase, or the phrase “MUST NOT”, means that the definition is an absolute prohibition of the specification.
- SHOULD: This word, or the adjective “RECOMMENDED”, means that there may exist valid reasons in particular circumstances to ignore a particular item, but the full implications must be understood and carefully weighed before choosing a different course.
- SHOULD NOT: This phrase, or the phrase “NOT RECOMMENDED”, means that there may exist valid reasons in particular circumstances when the particular behaviour is acceptable or even useful, but the full implications should be understood and the case carefully weighed before implementing any behaviour described with this label.
- MAY: This word, or the adjective “OPTIONAL”, means that an item is truly optional.

33

Revision History

Version	Release	Date	Paragraph	Comments
1	0	2021-04-21		Approved by SOC.

CONTENTS

List of figures

List of tables

1 Scope

This document is a deliverable of the ENTSO-E project “Header and metadata for CGM BP and CSA data exchanges”. The objective of the project is to support the Common Grid Model Building Process (CGM BP) and the Coordinated Security Analysis (CSA) data exchange project by building a UML metadata model and header schema to be used by ENTSO-E CGM BP and ENTSO-E CSA projects for data exchange.

Therefore, the items which are in scope and out of scope are illustrated in Figure 1 can be summarized as follows:

- In scope
  - Meet requirements for CGM BP and CSA projects
  - Focus on instance data header which will be needed anyway for a manifest approach that will be developed in the scope of IEC 61970-303
  - Develop canonical model and a “header profile” for the purpose of generating machine readable artifacts
- Out of scope
  - The overall metadata framework, i.e. so called manifest/framework that will be tackled in IEC 61970-303
  - The standardization work related to 61970-303 and 61970-459 standards
  - The implementation of the header in different projects based on IEC 61970 - CGMES exchanges

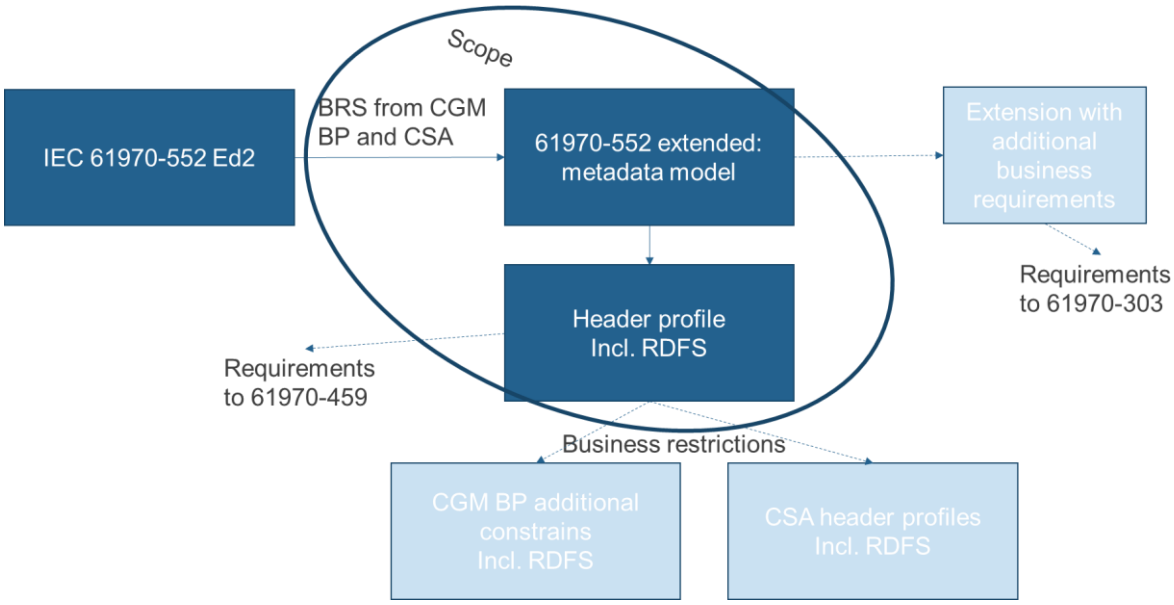


Figure 1. Scope of the project

In it is present version the document introduces the need of master reference data which is referenced from the header of the documents. However, the profile to govern the master reference data is not included in this version of the document. It will be covered in a next edition. This is why some sections of the document are intentionally left blank.

Annex A gives background information on the document header in the part related to modelling authority sets and versioning. Annex B contains one example of a document header which is used to illustrate some of the properties included in the document header.

2 Normative references

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

- IEC TS 61970-600-1:2017 Energy management system application program interface (EMS-API) - Part 600-1: Common Grid Model Exchange Specification (CGMES) - Structure and rules;
- IEC TS 61970-600-2:2017 Energy management system application program interface (EMS-API) - Part 600-2: Common Grid Model Exchange Specification (CGMES) - Exchange profiles specification.
- IEC 61970-600-1:FDIS Energy management system application program interface (EMS-API) - Part 600-1: Common Grid Model Exchange Standard (CGMES) - Structure and rules;
- IEC 61970-600-2:FDIS Energy management system application program interface (EMS-API) - Part 600-2: Common Grid Model Exchange Standard (CGMES) - Exchange profiles specification.
- W3C PROV-O: The PROV Ontology
- W3C Data Catalog Vocabulary (DCAT)
- W3C Time Ontology in OWL
- European Commission: Data Catalog Vocabulary Application Profile (DCAT-AP) for data portals in Europe
- IEC 61970-552: Energy management system application program interface (EMS-API) Part 552: CIMXML Model exchange format

3 Terms and definitions

3.1

serialisation
encoding of an ontology or dataset into a format that can be stored, typically in a file.

Note 1 to entry: The definition is adapted from W3C-RDF11-XML.

[SOURCE: ISO 21597-1:2020, 3.1.13]

3.2

ontology
specification of concrete or abstract things, and the relationships among them, in a prescribed domain of knowledge

Note 1 to entry: The specification should be computer processable.

Note 2 to entry: The definition is adapted from W3C-OWL2-SPEC.

[SOURCE: ISO 21597-1:2020, 3.1.7]

3.3

payload
primary information in the form of documents that is included within the container

Note 1 to entry: This does not include the header file or the ontology resource files.

[SOURCE: ISO 21597-1:2020, 3.1.2]

3.4

document
fixed and structured amount of information that can be managed and interchanged as a unit between users and systems

Note 1 to entry: This unit may not necessarily be human perceptible. Information is usually stored on a data medium.

Note 2 to entry: Used in the ISO 21597 series to refer to any document that forms part of the payload in the container, including any 2D or 3D models that represent built or natural assets in the physical world; these may be held in any standard or proprietary format.

[SOURCE: ISO 21597-1:2020, 3.1.3]

3.5

namespace
group of identifiers for elements and attributes that are collectively bound to a URI such that their use will not cause naming conflicts

Note 1 to entry: The definition is adapted from W3C-RDF11-CONCEPTS, 1.

[SOURCE: ISO 21597-1:2020, 3.1.19]

3.6

resource
something in the world (the "universe of discourse") denoted by an IRI or literal

Note 1 to entry: Anything can be a resource, including physical things, documents, abstract concepts, numbers and strings; the term is synonymous with "entity" as it is used in the RDF Semantics specification.

Note 2 to entry: The definition is adapted from W3C-RDF11-CONCEPTS.

[SOURCE: ISO 21597-1:2020, 3.1.14]

3.7

dataset
RDF(S)/OWL file that contains individuals that comply with the classes as specified by ontologies

[SOURCE: ISO 21597-1:2020, 3.1.10]

3.8
supersede
an entity (document, model, standard, profile, etc.) that has been replaced with a newer version of the same entity, or by a suitable other entity that contains the most current, reliable and/or available information

Note 1 to entry: The definition is adapted from ISO/IEC Guide 59:2019, 3.11.

3.9

model
collection of data describing instances, objects or entities, real or computed. In the context of CIM the semantics of the data is defined by profiles. Hence a model can contain equipment data, power flow initial values, power flow results etc.

Note 1 to entry: In power system analysis, a model is a set of static data describing the power system. Examples of Models include the Static Network Model, the Topology Solution, and the Network Solution produced by a power flow or state estimator application.

[SOURCE: IEC 61970-552:2016, 3.8]

3.10
modelling authority set
an abstract entity which is attributed to an agent (modelling authority). The modelling authority set is versioned by the agent.

3.11
modelling authority set version
a specialization of the modelling authority set which is attributed to an agent. A version of the modelling authority set can be seen as an envelop for models which conform to different profiles.

3.12

model exchange
the storing, accessing, transferring, and archiving of models

3.13

profile
schema that defines the structure and semantics of a model that may be exchanged

Note 1 to entry: A Profile is a restricted subset of the more general CIM.

[SOURCE: IEC 61970-552:2016, 3.9]

3.14

profile document
collection of profiles intended to be used together for a particular business purpose

[SOURCE: IEC 61970-552:2016, 3.10]

3.15

object property; property
name that may be used to qualify an object reference to get a value from or pass a value to an object

[SOURCE: ISO/IEC 1989:2014, 4.140]

4 Abbreviated terms

CIM	Common Information Model (electricity)
CGMES	Common Grid Model Exchange Standard
DSO	Distribution System Operator
ENTSO-E	European Network of Transmission System Operators for Electricity
IEC	The International Electrotechnical Commission
IOP	Interoperability Test
SO	System Operator
MAS	Model Authority Set
mRID	CIM Master Resource Identifier

OCL	Object Constraint Language
OWL	Web Ontology Language
RDF	Resource Description Framework
RDFS	RDF Schema
SHACL	Shapes Constraint Language
TSO	Transmission System Operator
URI	Uniform Resource Identifier
UUID	Universally Unique Identifier
XML	Extensible Markup Language
XSD	XML Schema Definition

5 Overview and methodology

5.1 Overview of current status of metadata discussion

A few years ago, IEC WG13 opened the discussion on 61970-303 (canonical metadata) and 61970-459 (profile) realizing this will need to grow to cover many use cases and wanted to remove the header from IEC 61970-552.

All IEC 61970-45x series and -600 series profiles (CGMES), as well as CGMES 2.4 do not include document header as part of the profile, i.e. when documents refer to EQ profile this does not include header definition.

In previous work there was a strong requirement that the current header (IEC 61970-552) shall not be changed, and all should be built on top. Main driver was the scheduled implementation of CGMES 2.4. It should be noted that when a document header is to be implemented, it impacts every import and export tool as it affects every single instance file.

Figure 2 Is a high-level illustration on different export types that are maintained in tools.

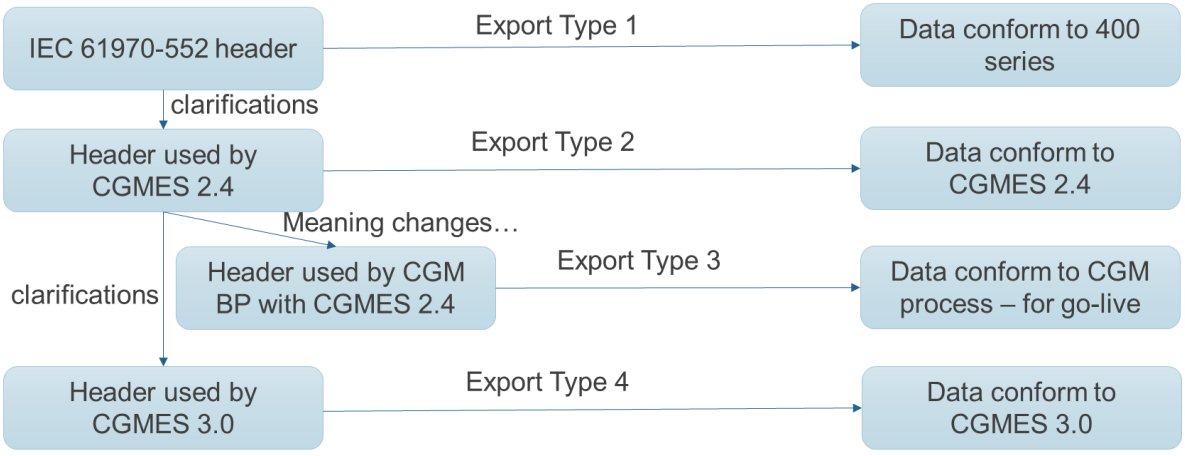


Figure 2. Different types of headers existing for IEC 61070 implementations

Although the implementation of the document header is not in scope the proposed solution, the project conducted a high-level discussion on possible variants for transitioning, thus the proposed solution is designed to enable options and facilitate the migration process.

5.2 Methodology and approach taken

As the metadata is in general data that describes other data, the project had to keep close collaboration with CGM BP and CSA projects in order to collect requirements and consult to collect feedback on the proposed solution for header and metadata. During the stage of collecting requirement more than 50 potential data fields (attributes or properties) were collected. These data field covered the following main groups:

- Data identification (e.g. identifiers, date of creation, version)
- Data linking (e.g. dependency or revision of data)
- Instance file type (e.g. type of profile, conformance to document/standard)
- Exact time period which the data represents/is valid for (e.g. scenario time, period start and end)
- Data description (e.g. free text description)
- Involved entity and its role (e.g. source data provider, service provider, intended data receiver)
- The area which the data represents (e.g. region, domain level)
- Process type (e.g. usage, service, CGM creation process, CSA)
- Process target period (e.g. time frame, target period)
- Document or process status (e.g. coordination run, iteration, document status)
- Data on the tool that created the data (e.g. Name of the tool and version or release)
- Data on Process Settings (e.g. power flow settings)

The project reviewed all requirements and identified which of the requested data fields have are overlapping in terms of meaning. A harmonization effort was performed and as a result a smaller set of data fields remained to be described and included in the proposed solution.

The project took into account the fact that both CGMES v2.4 (IEC TS 61970-600-1 and -2) and CGMES v3.0 (IEC 61970-600-1 and -2) utilize the header and metadata definitions by IEC 61970-552 with minimal adaptations. Within ENTSO-E and IEC there are discussions and standardisation work in progress which is focused on defining dedicated data model and profiles related to the exchange of metadata. These efforts aim at separation of the metadata from the instance data related to the so called “content” profiles.

Considering this the project investigated a number of W3C recommendations which are used worldwide and are positively recognise by the European Commission (EC). The analysis of available material from W3C and EC concluded that it is recommended to design a solution which mainly uses Provenance ontology (PROV-O), Time Ontology and Data Catalog Vocabulary (DCAT). However, in cases where it is not possible to find necessary information in the ontologies the project agreed to extend with properties under European metadata namespace (eumd).

The header/metadata requires availability of a set of reference metadata. For instance, the attribute prov:wasGeneratedBy requires a reference to an activity which produced the model or the related process. The activities are defined as reference metadata and their identifiers are referenced from the header to enable the receiving entity to retrieve the “static” (reference) information that it is not modified frequently. This approach imposes a requirement that both the sending entity and the receiving entity have access to a unique version of the reference

metadata. Therefore, each business process shall define which reference metadata is used and where it is located.

The proposed solution based on W3C approach will support any direction chosen in future, including the manifest approach currently under discussion in the scope of IEC 61970-303. In addition, the use of W3C ontologies will enable implementors to use a wide range of tooling not necessary designed for power system modelling, but which can interpret and visualise metadata natively.

The following figure illustrates the linkage between W3C Time Ontology, W3C Provenance ontology, W3C DCAT, the existing header and the extensions that were added.

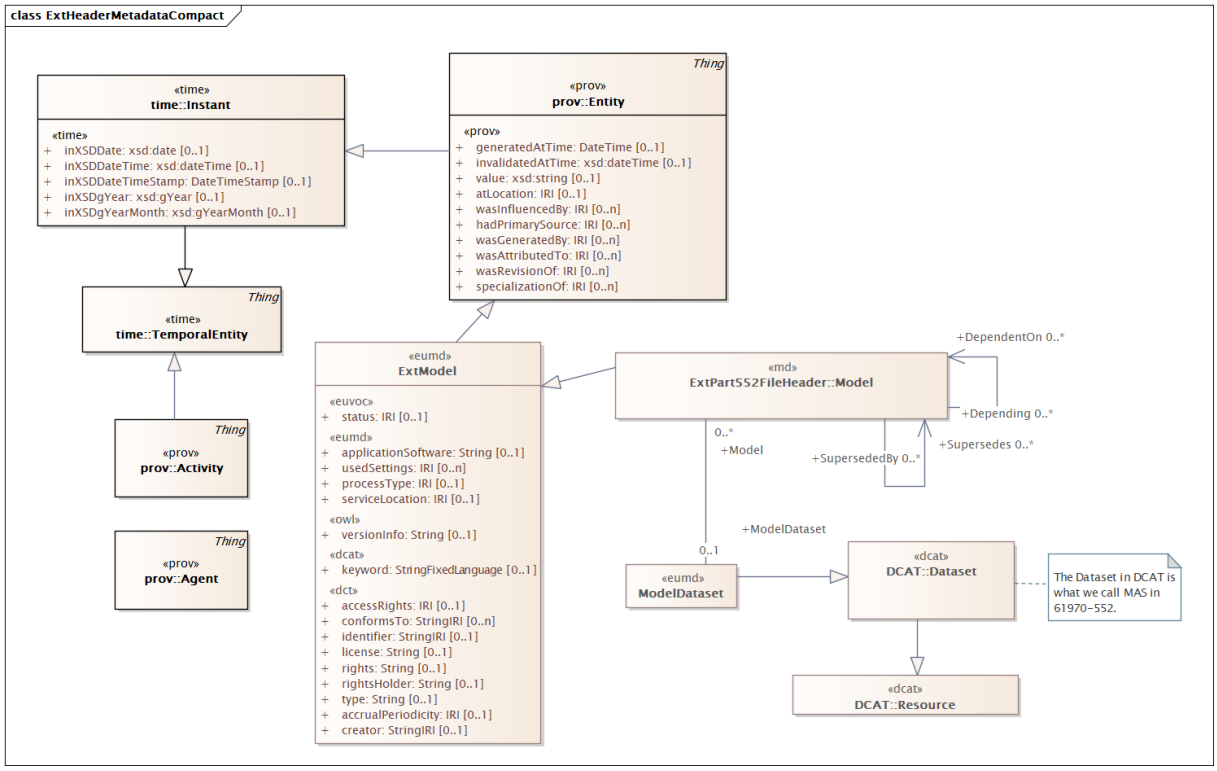


Figure 3. Canonical model and linkage between different ontologies

It should be noted that the objective of the project is to extend the existing header and the task to finalise the canonical model clarifying all details, which are to a large extend related to overall metadata exchange, is still to happen. In addition, the project faces multiple constraints such as profiling methods are currently not designed for metadata related profiles, the backwards compatibility, etc. Here it should be pointed out that the project intentionally selected the option to not introduce nested structure in the document header in order to keep the same kind of exchange as currently done. However, this is seen as a transition and in order to fully utilize W3C DCAT and Provenance next versions will need to be allowed to go beyond current practices. Especially when manifest is defined and in the description of master reference data.

5.3 Model, document and its header

At present stage the document header contains information about the metadata related to the model as well as its serialisation. This is considered intermediate solution until the overall framework and manifest exchange is standardised. However, this approach may cause confusion and misinterpretation.

352 As the description of the properties/attributes in the document header profile may not fully
353 clarify, the following table contains information which properties part of the document header
354 relate to the model that is serialised in the document and which relate to the document itself.

355 **Table 1 – Document header properties. Relationship to model or document**

name	Classification: model or document
created	Relates to the document.
description	Relates to the model.
modelingAuthoritySet	Relates to the model. The version of the MAS.
scenarioTime	Relates to the model.
profile	Relates to the model.
version	Relates to the document.
keyword	Relates to the model.
accessRights	Relates to the model.
conformsTo	Relates to the model and the document. For instance, a model conforms to the profile and the URI of the profile is given; a model also conforms to specifications and quality rules or constraints; the document conforms to the serialisation specifications, etc.
generatedAtTime	Relates to the document.
inXSDDateTimeStamp	Relates to the model.
applicationSoftware	Relates to the document.
hasXSDDuration	Relates to the model.
identifier	Relates to the model.
license	Relates to the model.
rights	Relates to the model.
rightsHolder	Relates to the model.
type	Relates to the model.
atLocation	Relates to the model.
status	Relates to the model.
wasInfluencedBy	Relates to the model.
hadPrimarySource	Relates to the model. The version of the MAS from where a version of a model is originating.
wasGeneratedBy	Relates to the model.
wasAttributedTo	Relates to the model.
usedSettings	Relates to the model.
wasRevisionOf	Relates to the model.
versionInfo	Relates to the model.
specializationOf	Relates to the model. The version of the MAS that is managing the version of the model.
DependentOn	Relates to the model.
Supersedes	Relates to the model.
accrualPeriodicity	Relates to the model.
processType	Relates to the model.
creator	Relates to the model.
serviceLocation	Relates to the model.

5.4 Business Process, Time Horizon, Run and Iteration

A concept was introduced to reason about granularity of the Business. The idea is to enable data provider to implicitly indicate for which Business Process sub process the data is intended for and the Service Provider to explicitly indicate in which Business Process sub process the data was generated in.

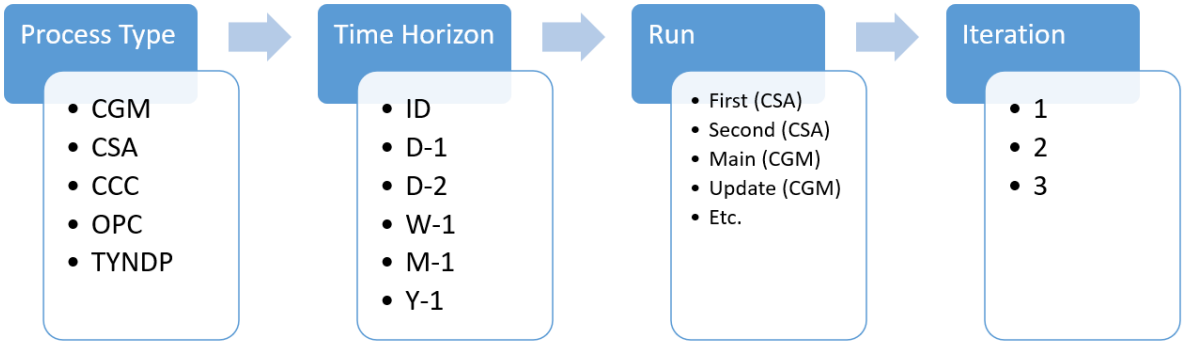
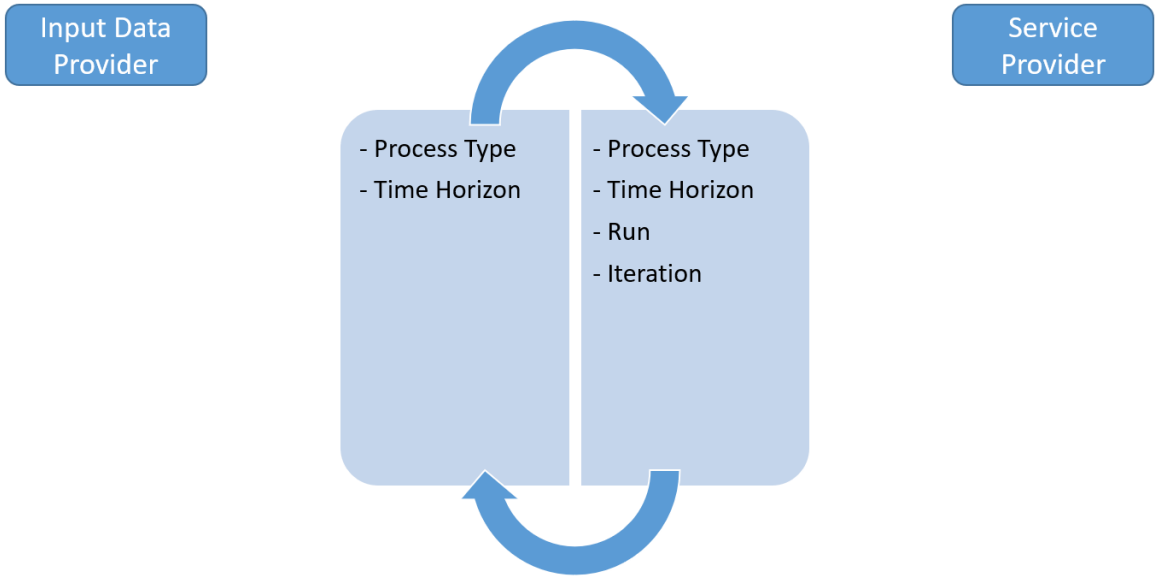


Figure 4. Business Granularity

This for example allows different input data used for different business sub processes. For example, if data provider does not plan to update their data for specific Run and Iteration, then they only need to define the Process Type and Time Horizon and Service Provider can pick up the latest version of data with that metadata for each Run and Iteration. Data Provider could also want to provide data without Time Horizon or Process Type, if they do plan to use exactly the same data in different Time Horizons and Processes. Below is a example for most common use case, where input data provider intends to have only same data to be used within given Process and Time Horizon.

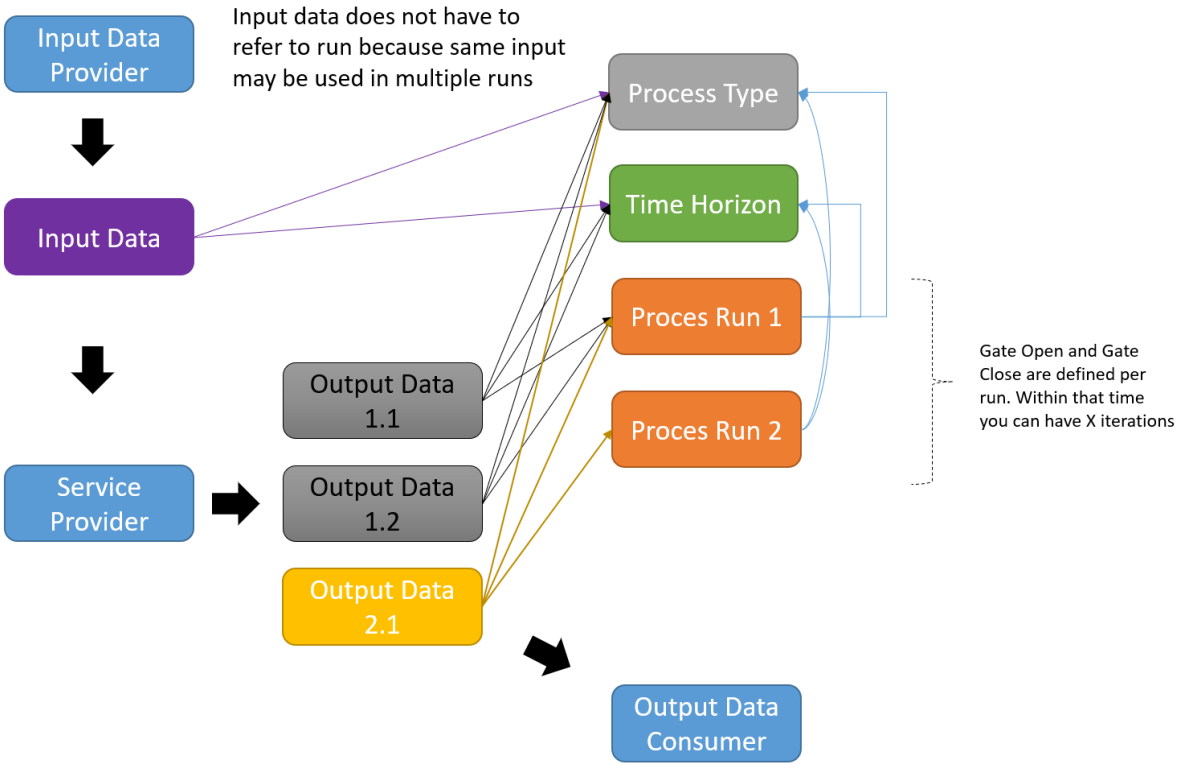
Meaning:

- Data Provider indicates:
  - Process Type
  - Time Horizon
- Service Provider indicates:
  - Process Type
  - Time Horizon
  - Run
  - Iteration



381
382 **Figure 5. Expected Business Metadata**

383
384



385
386 **Figure 6. Example**

387

5.5 Process settings

It was identified that there is a need to know under which conditions the input or output data was generate. In case of CSA and CGM BP the setting are power flow settings used by Data Providers and Service Providers, but it could be any arbitrary set of settings that a project defines.

Meaning:

1. A Project will define a number of settings, that can be used
2. Data Provider or Service Provider refers in the Document Header to the concrete setting that was used to generate the data.

In general, this approach could be also used out of context of Projects where in data exchange Parties refer to their own settings (preferably publicly available)

Example of settings that CSA and CGM BP have collected to define Power Flow (this is not final nor complete example)

Name	Type	Description
mRID	urn	The unique ID of the powerflow setting
name	string	Name of the setting
description	string	Description of the settings
algorithmKind	enum	It defines the power flow algorithm. (fullNewtonRaphson, fixedSlopeNewtonRaphson, fastDecoupled, gaussSeidel, modifiedGaussSeidel, dcPowerFlow)
flatStart	boolean	True means that power flow used a flat start.
pTolerance	ActivePower	The active power tolerance for The given power flow solution. (SV injections can't be greater than tolerance)
qTolerance	ReactivePower	The reactive power tolerance for a given power flow solution.
vTolerance	float	The largest difference between actual and scheduled voltage magnitude for controlled node, in per unit, at each node where voltage is subject to control to a set point, and for which at least one of the devices participating in the control of bus voltage to its set

		point is not at a reactive power limit, must be less than the controlled bus voltage error convergence tolerance. Number in pu of baseVoltage
voltageLimitAngle	AngleDegrees	The maximum allowed voltage angle between two buses for The given power flow solution.
impedanceThreshold	float	0 means not used, number in pu (basePower=100MW) if used for modeling zero or low impendence branches
ircEnabeled	boolean	True means load response characteristics are enabled
respectQlimits	boolean	True means that VAr limits are respected during power flow calculation.
transformerRatioTapControlPriority	integer	0 means not used/disabled, 1 means highest priority.
transformerPhaseTapControlPriority	integer	0 means not used/disabled, 1 means highest priority.
switchShuntControlPriority	integer	0 means not used/disabled, 1 means highest priority.
staticVarCompensatorControlPriority	integer	0 means not used/disabled, 1 means highest priority.
slackDistributionKind	enum	Defines slack distribution: loadDistribution - Slack distribution on ConformLoads generationDistributionOnSlack - Generation distribution kind proportional to GeneratingUnit.normalPF; generationDistributionOnPV – Generation distribution kind of slack per all PV nodes, proportional to reserve. generatonOnReferenceMachine – No distribution is done, slack remains on Reference Machine
enableInterchangeControl	boolean	True means area interchange control is enabled.

shiftKind	enum	Defines type of scaling used to reach defined net-position: conformLoadShift – only conform load is used allLoadShift – all load type is used generationShift – synchronous machine of generation type is used glskShift – both load and generation used

6 Application profile specification

6.1 Version information

The content is generated from UML model file CGMES30v25_501-20v01_HeaderMetaData-10v08_CSA01v35.eap.

The document header profile uses extensions. The prefix and the uri of the namespace are as follows:

- Prefix: eumd
- URI: http://entsoe.eu/ns/Metadata-European#

This edition is based on the IEC 61970 UML version 'IEC61970CIM17v40', dated '2020-08-24'.

6.1.1 Document header profile

- Title: Document header vocabulary
- Keyword: DH
- Description: This vocabulary is describing the document header profile..
- Version IRI: http://entsoe.eu/ns/CIM/DocumentHeader-EU/1.0
- Version info: 2.0.0
- Prior version: 1.0.0
- Conforms to: urn:iso:std:iec:61970-401:draft:ed-1|urn:iso:std:iec:61970-501:draft:ed-2
- Identifier: urn:uuid:1c73cc65-8bcc-445a-8d18-0dbd7c94b118

6.1.2 Reference metadata profile

[left blank intentionally – placeholder for next version]

6.2 Profile constraints

This clause defines requirements and constraints that shall be fulfilled by applications that conform to this document. The naming of the rules shall not be used for machine processing. The rule names are just a string. The naming convention of the constraints is as follows.

"{rule.Type}:{rule.Standard}:{rule.Profile}:{rule.Property}:{rule.Name}"

where

rule.Type: C – for constraint; R – for requirement

rule.Standard: the number of the standard e.g. 301 for 61970-301, 456 for 61970-456, 13 for 61968-13. 61970-600 specific constraints refer to 600 although they are related to one or combination of the 61970-450 series profiles. For document header, DH is used. For reference data, RD is used.

rule.Profile: the abbreviation of the profile, e.g. TP for Topology profile. If set to “ALL” the constraint is applicable to all IEC 61970-600 profiles.

rule.Property: for UML classes, the name of the class, for attributes and associations, the name of the class and attribute or association end, e.g. EnergyConsumer, IdentifiedObject.name, etc. If set to “NA” the property is not applicable to a specific UML element.

rule.Name: the name of the rule. It is unique for the same property.

Example: C:600:ALL:IdentifiedObject.name:stringLength

This document is the master for rules and constraints tagged "MD". For the sake of self-containment, the list below also includes a copy of the relevant rules from IEC 61970-452, tagged "452".

- C:452:ALL:NA:datatypes

  According to 61970-501, datatypes are not exchanged in the instance data. The UnitMultiplier is 1 in cases none value is specified in the profile.

- R:452:ALL:NA:exchange

  Optional and required attributes and associations must be imported and exported if they are in the model file prior to import.

- R:452:ALL:NA:exchange1

  If an optional attribute does not exist in the imported file, it does not have to be exported in case exactly the same data set is exported, i.e. the tool is not obliged to automatically provide this attribute. If the export is resulting from an action by the user performed after the import, e.g. data processing or model update the export can contain optional attributes.

- R:MD:ALL:NA:exchange

  The selection of optional and required attributes as well as their cardinality is made so as to ensure a minimum set of required attributes without which the exchange does not fulfil its basic purpose. Business processes governing different exchanges can require mandatory exchange of certain optional attributes or associations or restrict the usage of some attributes, without modifying their meaning. Optional and required attributes and associations shall therefore be supported by applications which claim conformance with this document. This provides flexibility for the business processes to adapt to different business requirements and base the exchanges on profile compliant applications.

- R:MD:ALL:NA:exchange1

  An exporter may, at his or her discretion, produce a serialization containing additional data described by the metadata profiles or in a custom namespace. This data is not

subject to extensive data validation and shall not invalidate the document which is exchanged.

6.3 Available code components

The following code components are available:

- Enterprise architect project file
- RDFS: The RDFS for the header is generated by CimSyntaxGen. The version (type of export) of RDFS v2020 which represents an augmented version if IEC 51970-501. Version information related to the RDFS is included in an ontology-based file header of the RDFS.
- SHACL constraints for the header: In this release only basic SHACL shapes are derived. In case of additional requirements and dependencies are found the set of constraints can be further developed.

7 Detailed profile specification

7.1 Document header profile

7.1.1 General

The package describes the profile for the extended header.

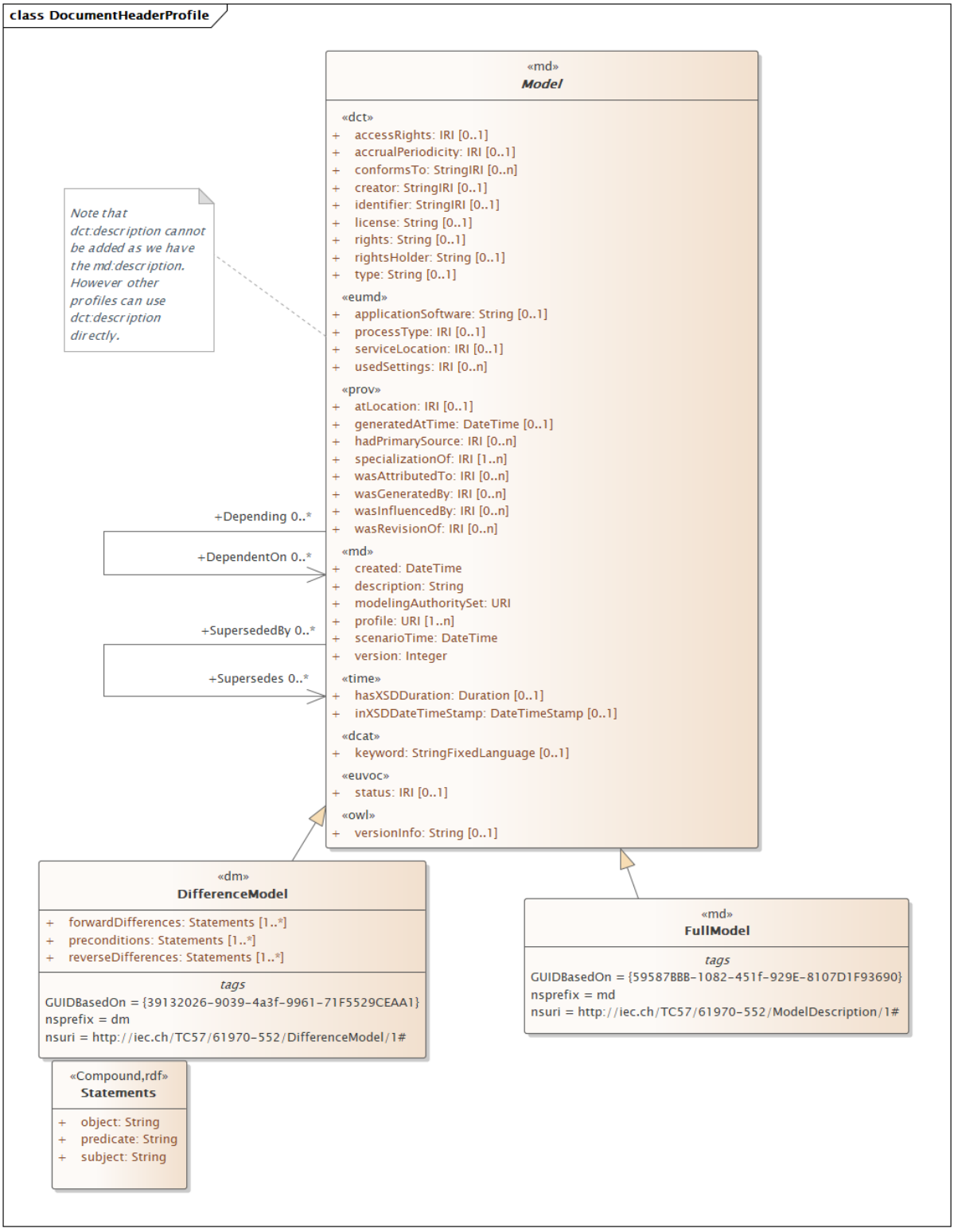


Figure 7 – Class diagram DocumentHeaderProfile::DocumentHeaderProfile

Figure 7: The diagram defines the extended document header model.

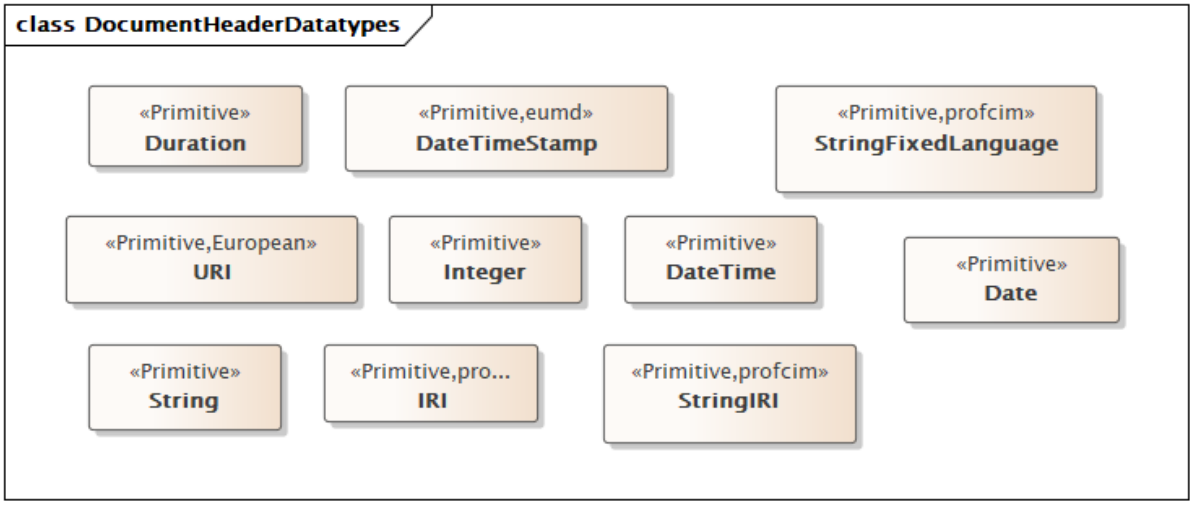


Figure 8 – Class diagram DocumentHeaderProfile::DocumentHeaderDatatypes

Figure 8: The diagram shows datatypes that are used by classes in the profile. Stereotypes are used to describe the datatypes. The following stereotypes are defined:
<<enumeration>> A list of permissible constant values.
<<Primitive>> The most basic data types used to compose all other data types.
<<CIMDatatype>> A datatype that contains a value attribute, an optional unit of measure and a unit multiplier. The unit and multiplier may be specified as a static variable initialized to the allowed value.
<<Compound>> A composite of Primitive, enumeration, CIMDatatype or other Compound classes, as long as the Compound classes do not recurse.
For all datatypes both positive and negative values are allowed unless stated otherwise for a particular datatype.

7.1.2 (abstract,md) Model root class

A Model is a collection of data describing instances, objects or entities, real or computed. In the context of CIM the semantics of the data is defined by profiles. Hence a model can contain equipment data, power flow initial values, power flow results etc.
The Model class describes the header content that is the same for the FullModel and the DifferenceModel. A Model is identified by an rdf:about attribute. The rdf:about attribute uniquely describes the model data and not the CIMXML document. A new rdf:about identification is generated for created documents only when the model data has changed. A repeated creation of documents from unchanged model data shall have the same rdf:about identification as previous document generated from the same model data.
Table 2 shows all attributes of Model.

Table 2 – Attributes of DocumentHeaderProfile::Model

name	mult	type	description
created	1..1	DateTime	(md) The date and time when the model was created. It is the time of the serialization. The format is an extended format according to the ISO 8601-2005. European exchanges shall refer to UTC, e.g. <md:Model.created>2014-05-15T17:48:31.474Z</md:Model.created>.
generatedAtTime	0..1	DateTime	(prov) Generation is the completion of production of a new entity by an activity. This entity did not exist before generation and becomes available for usage after this generation. [CIM context:

name	mult	type	description
			The date and time when the model was serialized in the document where the header is located. The format is an extended format according to the ISO 8601-2005. European exchanges shall refer to UTC.].
hasXSDDuration	0..1	Duration	(time) Extent of a temporal entity, expressed using xsd:duration. [CIM context: The duration of the validity period of the model that it is serialized in the document where the header is located. It is only used in relation to the inXSDDateTimeStamp property which indicates the beginning of the validity period of the model. The end of the validity period is derived from both inXSDDateTimeStamp and hasXSDDuration.].
inXSDDateTimeStamp	0..1	DateTimeStamp	(time) Position of an instant, expressed using xsd:dateTimeStamp, in which the time-zone field is mandatory. [CIM context: The date and time that this model represents, i.e. for which the model is (or was) valid. If used in relation with hasXSDDuration it indicates the beginning of the validity period. It is indicating either an instant (in cases where the model is only valid for a point in time) or the start time of a period. If not provided the model is considered valid for any time stamp. The format is an extended format according to the ISO 8601-2005. European exchanges shall refer to UTC.].
status	0..1	IRI	(euvoc) Indicates the status of a skos:Concept or a skosxl:Label, or any resource related to controlled vocabulary management. [CIM context: The condition or position of an object with regard to its standing. (Validated, Primary, Backup etc.)].
applicationSoftware	0..1	String	(eumd) Identifies the application software which generated this instance file. The application software term is defined in ISO/IEC/IEEE 24765:2017. The application software can be identified either: - as a string which contains information on the software name and version, e.g. <tool_name>-<major_version>.<minor_version>.<patch>, or - as a reference to a software identification tag as defined by ISO/IEC 19770-2:2015 and ISO/IEC/IEEE 24765:2017.
description	1..1	String	(md) A description of the model, e.g. the name of person that created the model and for what purpose. The number of UTF-8 characters is limited to 2000.
modelingAuthoritySet	1..1	URI	(md) A URN/URI referring to the organisation role / model authority set reference. The organization role is the source of the model. It is the same for all profiles part of a model exchange.
atLocation	0..1	IRI	(prov) A location can be an identifiable geographic place (ISO 19112), but it can also be a non-geographic place such as a directory, row, or column. As such, there are numerous ways in

name	mult	type	description
			which location can be expressed, such as by a coordinate, address, landmark, and so forth. [CIM context: Reference to a region or a domain for which this model is provided.].
keyword	0..1	StringFixedLanguage	(dcat) A keyword or tag describing a resource. [CIM context: The intended content type of the model, usually the profile keyword. Used to identify what profiles and content is expected in the document, e.g., Equipment, Boundary, SSH, AE, etc. The same keyword is used for different versions of same profile. It can be also used to identify different content based on the same profile. For instance, as the equipment profile can be used for both boundary data and equipment not related to boundary, the keyword is different to indicate that boundary data is exchanged. In order to avoid ambiguity the property is not exchanged in cases where the document contains multiple profiles referenced by dct:conformsTo.].
scenarioTime	1..1	DateTime	(md) The date and time that this model represents, i.e. for which the model is valid. The format is an extended format according to the ISO 8601-2005. European exchanges shall refer to UTC, e.g. <md:Model.scenarioTime>2030-01-15T17:00:00.000Z</md:Model.scenarioTime>.
accessRights	0..1	IRI	(dct) Information about who access the resource or an indication of its security status.Access Rights may include information regarding access or restrictions based on privacy, security, or other policies. [CIM context: Reference to the confidentiality level that shall be applied when handling this model.].
profile	1..n	URI	(md) URN/URI describing the profiles that governs this model. It uniquely identifies the profiles and its version, e.g. http://iec.ch/TC57/61970-456/SteadyStateHypothesis/2/0.
wasInfluencedBy	0..n	IRI	(prov) Influence is the capacity of an entity, activity, or agent to have an effect on the character, development, or behavior of another by means of usage, start, end, generation, invalidation, communication, derivation, attribution, association, or delegation. [CIM context: A reference to the model on which the model serialised in this document depends on. The references are maintained by the producer of the model. Minimum requirements for the dependency are specified and can be restricted within a business process as long as they do not contradict requirements by standards. For instance, IEC 61970-600-1 defines minimum requirements for the profiles defined in that standard.].
conformsTo	0..n	StringIRI	(dct) An established standard to which the described resource conforms. [CIM context:

name	mult	type	description
			An IRI describing the profile that governs this model. It uniquely identifies the profile and its version. Multiple instances of the property describe all standards or specifications to which the model and the document representing this model conform to. A document would normally conform to profile definitions, the constraints that relate to the profile and/or the set of business specific constrains. A reference to a machine- readable constraints or specification indicates that the document was tested against these constraints and it conforms to them.].
hadPrimarySource	0..n	IRI	(prov) A primary source for a topic refers to something produced by some agent with direct experience and knowledge about the topic, at the time of the topic's study, without benefit from hindsight. Because of the directness of primary sources, they 'speak for themselves' in ways that cannot be captured through the filter of secondary sources. As such, it is important for secondary sources to reference those primary sources from which they were derived, so that their reliability can be investigated. A primary source relation is a particular case of derivation of secondary materials from their primary sources. It is recognized that the determination of primary sources can be up to interpretation, and should be done according to conventions accepted within the application's domain. [CIM context: Reference to a modelling authority set version sourcing the model. It is only used in cases where a model is modified by an agent which has different version of modelling authority set. The agent that makes a revision of a model indicates the primary source using this property and also refers to its own version of modelling authority set using prov:specializationOf.].
version	1..1	Integer	(md) The version of the model. If the instance file is imported and exported with no change, the version number is kept the same. The version changes only if the content of the file changes. It is the same logic as for the header id. The version is the human readable id. [CIM context: It relates to the version of the document and not the version of the model which is serialized.].
identifier	0..1	StringIRI	(dct) An unambiguous reference to the resource within a given context. Recommended practice is to identify the resource by means of a string conforming to an identification system. Examples include International Standard Book Number (ISBN), Digital Object Identifier (DOI), and Uniform Resource Name (URN). Persistent identifiers should be provided as HTTP URIs. [CIM context: A unique identifier of the model which is serialised in the document where the header is located. The identifier is persistent for a given version of the model and shall change when the model changes. If a model is serialized as complete (full) model or as difference model exchange the identifier shall be the same. The identifier shall not be

name	mult	type	description
			used as an identifier of the document which can be different for a given version of a model.].
wasGeneratedBy	0..n	IRI	(prov) Generation is the completion of production of a new entity by an activity. This entity did not exist before generation and becomes available for usage after this generation. [CIM context: Reference to an activity or the exact business nature (process, configuration) which produced or uses the model.].
license	0..1	String	(dct) A legal document giving official permission to do something with the resource. Recommended practice is to identify the license document with a URI. If this is not possible or feasible, a literal value that identifies the license may be provided. [CIM context: Reference to the license under which the data is made available. If no license holder is defined, then the original data provider holds the license.].
wasAttributedTo	0..n	IRI	(prov) Attribution is the ascribing of an entity to an agent. [CIM context: Reference to the agent (or service provider) from which the model originates.].
usedSettings	0..n	IRI	(eumd) Reference to a set of parameters describing used settings (e.g. power flow settings, process settings, etc.) applied to the model prior its serialisation.
wasRevisionOf	0..n	IRI	(prov) A revision is a derivation for which the resulting entity is a revised version of some original. The implication here is that the resulting entity contains substantial content from the original. Revision is a particular case of derivation. [CIM context: When a model is updated the resulting model supersedes the models that were used as basis for the update. Hence this is a reference to the model which are superseded by this model. A model can supersede 1 or more models, e.g. a difference model or a full model supersede multiple models (difference or full). In this case, multiple properties are included in the header. The referenced document(s) is (are) identified by the URN/MRID/UUID in the FullModel rdf:about attribute when full model(s) is (are) referenced and by the URN/MRID/UUID in the DifferenceModel rdf:about attribute when difference model(s) is (are) referenced.].
rights	0..1	String	(dct) A statement that concerns all rights not addressed with dct:license or dct:accessRights, such as copyright statements.
rightsHolder	0..1	String	(dct) Information about rights held in and over the resource. Typically, rights information includes a statement about various property rights associated with the resource, including intellectual property rights. Recommended practice is to refer to a rights statement with a URI. If this is not possible or feasible, a literal

name	mult	type	description
			value (name, label, or short text) may be provided.
type	0..1	String	(dct) The nature or genre of the resource. Recommended practice is to use a controlled vocabulary such as the DCMI Type Vocabulary [DCMI-TYPE]. To describe the file format, physical medium, or dimensions of the resource, use the property Format.
processType	0..1	IRI	(eumd) The exact business nature. Reference to Business Process configurations.
accrualPeriodicity	0..1	IRI	(dct) The frequency with which items are added to a collection. [CIM context: Reference to the time frame.].
specializationOf	1..n	IRI	(prov) An entity that is a specialization of another shares all aspects of the latter, and additionally presents more specific aspects of the same thing as the latter. In particular, the lifetime of the entity being specialized contains that of any specialization. Examples of aspects include a time period, an abstraction, and a context associated with the entity. [CIM context: Reference to modelling authority set version sourcing the model. The agent that makes a revision of a model indicates the primary source using prov:hadPrimarySource and refers to its own version of modelling authority set using this property.].
creator	0..1	StringIRI	(dct) An entity responsible for making the resource. Recommended practice is to identify the creator with a URI. If this is not possible or feasible, a literal value that identifies the creator may be provided. [CIM context: The name of the agent (Modeling Authority) from which the model originates].
serviceLocation	0..1	IRI	(eumd) Reference to a service location (region or a domain).
versionInfo	0..1	String	(owl) The annotation property that provides version information for an ontology or another OWL construct. DCAT-AP definition: This property contains a version number or other version designation of the Dataset. OWL definition: An owl:versionInfo statement generally has as its object a string giving information about this version, for example RCS/CVS keywords. This statement does not contribute to the logical meaning of the ontology other than that given by the RDF(S) model theory. Although this property is typically used to make statements about ontologies, it may be applied to any OWL construct. For example, one could attach a owl:versionInfo statement to an OWL class. [CIM context: The version of the model. If the document is imported and exported with no change the

name	mult	type	description
			version number is the kept same. The version changes only if the content of the model changes. It is the same logic as for the header identifier. The version is the human readable identifier.].

Table 3 shows all association ends of Model with other classes.

Table 3 – Association ends of DocumentHeaderProfile::Model with other classes

mult from	name	mult to	type	description
0..*	DependentOn	0..*	Model	A reference to the model documents that the model described by this document depends on. In general there can be 0 or many Model.DependentOn depending on the profile and the content of the instance file. For instance: – A load flow solution depends on the topology model it was computed from – A topology model computed by a topology processor depends on the network model it was computed from. The referenced models are identified by the FullModel rdf:about attribute for full model documents and by DifferenceModel rdf:about attribute for difference model documents. The references are maintained by the producer of the CIMXML document and the references are valid for the model with version and identifier for which the document was created.
0..*	Supersedes	0..*	Model	When a model is updated the resulting model supersedes the models that were used as basis for the update. Hence this is a reference to the CIMXML documents which are superseded by this model. A model (or instance file) can supersede 1 or more models, e.g. a difference model or a full model supersede multiple models (difference or full). In this case more than one Model.Supersedes are included in the header. The referenced document(s) is (are) identified by the URN/MRID/UUID in the FullModel rdf:about attribute when full model(s) is (are) referenced and by the URN/MRID/UUID in the DifferenceModel rdf:about attribute when difference model(s) is (are) referenced.

7.1.3 (dm) DifferenceModel

Inheritance path = Model
It represents the difference model header. The content is described by the Model class, the association role forwardDifferences and association role reverseDifferences. Both association roles may have one set of Statements.
Table 4 shows all attributes of DifferenceModel.

523 **Table 4 – Attributes of DocumentHeaderProfile::DifferenceModel**

name	mult	type	description
preconditions	1..*	Statements	A property of the difference model whose value is the collection of precondition statements.
reverseDifferences	1..*	Statements	A property of the difference model whose value is the collection of reverse difference statements.
forwardDifferences	1..*	Statements	A property of the difference model whose value is a collection of statements (i.e., resources of type rdf:Statement) representing the forward difference statements.
created	1..1	DateTime	(md) inherited from: Model
generatedAtTime	0..1	DateTime	(prov) inherited from: Model
hasXSDDuration	0..1	Duration	(time) inherited from: Model
inXSDDateTimeStamp	0..1	DateTimeStamp	(time) inherited from: Model
status	0..1	IRI	(euvoc) inherited from: Model
applicationSoftware	0..1	String	(eumd) inherited from: Model
description	1..1	String	(md) inherited from: Model
modelingAuthoritySet	1..1	URI	(md) inherited from: Model
atLocation	0..1	IRI	(prov) inherited from: Model
keyword	0..1	StringFixedLanguage	(dcat) inherited from: Model
scenarioTime	1..1	DateTime	(md) inherited from: Model
accessRights	0..1	IRI	(dct) inherited from: Model
profile	1..n	URI	(md) inherited from: Model
wasInfluencedBy	0..n	IRI	(prov) inherited from: Model
conformsTo	0..n	StringIRI	(dct) inherited from: Model
hadPrimarySource	0..n	IRI	(prov) inherited from: Model
version	1..1	Integer	(md) inherited from: Model
identifier	0..1	StringIRI	(dct) inherited from: Model
wasGeneratedBy	0..n	IRI	(prov) inherited from: Model
license	0..1	String	(dct) inherited from: Model
wasAttributedTo	0..n	IRI	(prov) inherited from: Model
usedSettings	0..n	IRI	(eumd) inherited from: Model
wasRevisionOf	0..n	IRI	(prov) inherited from: Model
rights	0..1	String	(dct) inherited from: Model
rightsHolder	0..1	String	(dct) inherited from: Model
type	0..1	String	(dct) inherited from: Model
processType	0..1	IRI	(eumd) inherited from: Model
accrualPeriodicity	0..1	IRI	(dct) inherited from: Model
specializationOf	1..n	IRI	(prov) inherited from: Model
creator	0..1	StringIRI	(dct) inherited from: Model
serviceLocation	0..1	IRI	(eumd) inherited from: Model

524

525 Table 5 shows all association ends of DifferenceModel with other classes.

526 **Table 5 – Association ends of DocumentHeaderProfile::DifferenceModel with other**
527 **classes**

mult from	name	mult to	type	description
0..*	DependentOn	0..*	Model	inherited from: Model
0..*	Supersedes	0..*	Model	inherited from: Model

528

529 ### 7.1.4 (md) FullModel

530 Inheritance path = Model
531 It represents the full model header and its contents is described by the Model class.
532 Table 6 shows all attributes of FullModel.

533 **Table 6 – Attributes of DocumentHeaderProfile::FullModel**

name	mult	type	description
created	1..1	DateTime	(md) inherited from: Model
generatedAtTime	0..1	DateTime	(prov) inherited from: Model
hasXSDDuration	0..1	Duration	(time) inherited from: Model
inXSDDateTimeStamp	0..1	DateTimeStamp	(time) inherited from: Model
status	0..1	IRI	(euvoc) inherited from: Model
applicationSoftware	0..1	String	(eumd) inherited from: Model
description	1..1	String	(md) inherited from: Model
modelingAuthoritySet	1..1	URI	(md) inherited from: Model
atLocation	0..1	IRI	(prov) inherited from: Model
keyword	0..1	StringFixedLanguage	(dcat) inherited from: Model
scenarioTime	1..1	DateTime	(md) inherited from: Model
accessRights	0..1	IRI	(dct) inherited from: Model
profile	1..n	URI	(md) inherited from: Model
wasInfluencedBy	0..n	IRI	(prov) inherited from: Model
conformsTo	0..n	StringIRI	(dct) inherited from: Model
hadPrimarySource	0..n	IRI	(prov) inherited from: Model
version	1..1	Integer	(md) inherited from: Model
identifier	0..1	StringIRI	(dct) inherited from: Model
wasGeneratedBy	0..n	IRI	(prov) inherited from: Model
license	0..1	String	(dct) inherited from: Model
wasAttributedTo	0..n	IRI	(prov) inherited from: Model
usedSettings	0..n	IRI	(eumd) inherited from: Model
wasRevisionOf	0..n	IRI	(prov) inherited from: Model
rights	0..1	String	(dct) inherited from: Model
rightsHolder	0..1	String	(dct) inherited from: Model
type	0..1	String	(dct) inherited from: Model
processType	0..1	IRI	(eumd) inherited from: Model
accrualPeriodicity	0..1	IRI	(dct) inherited from: Model
specializationOf	1..n	IRI	(prov) inherited from: Model

name	mult	type	description
creator	0..1	StringIRI	(dct) inherited from: Model
serviceLocation	0..1	IRI	(eumd) inherited from: Model

Table 7 shows all association ends of FullModel with other classes.

Table 7 – Association ends of DocumentHeaderProfile::FullModel with other classes

mult from	name	mult to	type	description
0..*	DependentOn	0..*	Model	inherited from: Model
0..*	Supersedes	0..*	Model	inherited from: Model

7.1.5 Date primitive

Date as "yyyy-mm-dd", which conforms with ISO 8601. UTC time zone is specified as "yyyy-mm-ddZ". A local timezone relative UTC is specified as "yyyy-mm-dd(+/-)hh:mm".

7.1.6 (profcim) URL primitive

A Uniform Resource Locator (URL), colloquially termed a web address, is a reference to a web resource that specifies its location on a computer network and a mechanism for retrieving it. A URL is a specific type of Uniform Resource Identifier (URI), although many people use the two terms interchangeably.URLs occur most commonly to reference web pages (http), but are also used for file transfer (ftp), email (mailto), database access (JDBC), and many other applications.

7.1.7 (eumd) DateTimeStamp primitive

Position of an instant, expressed using xsd:dateTimeStamp, in which the time-zone field is mandatory.

7.1.8 Duration primitive

Duration as "PnYnMnDTnHnMnS" which conforms to ISO 8601, where nY expresses a number of years, nM a number of months, nD a number of days. The letter T separates the date expression from the time expression and, after it, nH identifies a number of hours, nM a number of minutes and nS a number of seconds. The number of seconds could be expressed as a decimal number, but all other numbers are integers.

7.1.9 (profcim) IRI primitive

An IRI (Internationalized Resource Identifier) within an RDF graph is a Unicode string that conforms to the syntax defined in RFC 3987.
The primitive is serialized as rdf:resource in RDFXML.
IRIs in the RDF abstract syntax must be absolute, and may contain a fragment identifier.
IRI equality: Two IRIs are equal if and only if they are equivalent under Simple String Comparison according to section 5.1 of [RFC3987]. Further normalization must not be performed when comparing IRIs for equality.
IRIs are a generalization of URIs [RFC3986] that permits a wider range of Unicode characters. Every absolute URI and URL is an IRI, but not every IRI is an URI. When IRIs are used in operations that are only defined for URIs, they must first be converted according to the mapping defined in section 3.1 of [RFC3987]. A notable example is retrieval over the HTTP protocol. The mapping involves UTF-8 encoding of non-ASCII characters, %-encoding of octets not allowed in URIs, and Punycode-encoding of domain names.

7.1.10 (profcim) StringFixedLanguage primitive

A string consisting of a sequence of characters. The character encoding is UTF-8. The string length is unspecified and unlimited.
The primitive is serialized as literal without language support.

7.1.11 (profcim) StringIRI primitive

An IRI (Internationalized Resource Identifier) within an RDF graph is a Unicode string that conforms to the syntax defined in RFC 3987.
The primitive is serialized as literal without language support.
IRIs in the RDF abstract syntax must be absolute, and may contain a fragment identifier.
IRI equality: Two IRIs are equal if and only if they are equivalent under Simple String Comparison according to section 5.1 of [RFC3987]. Further normalization must not be performed when comparing IRIs for equality.
IRIs are a generalization of URIs [RFC3986] that permits a wider range of Unicode characters. Every absolute URI and URL is an IRI, but not every IRI is an URI. When IRIs are used in operations that are only defined for URIs, they must first be converted according to the mapping defined in section 3.1 of [RFC3987]. A notable example is retrieval over the HTTP protocol. The mapping involves UTF-8 encoding of non-ASCII characters, %-encoding of octets not allowed in URIs, and Punycode-encoding of domain names.

7.1.12 String primitive

A string consisting of a sequence of characters. The character encoding is UTF-8. The string length is unspecified and unlimited.

7.1.13 Integer primitive

An integer number. The range is unspecified and not limited.

7.1.14 DateTime primitive

Date and time as "yyyy-mm-ddThh:mm:ss.sss", which conforms with ISO 8601. UTC time zone is specified as "yyyy-mm-ddThh:mm:ss.sssZ". A local timezone relative UTC is specified as "yyyy-mm-ddThh:mm:ss.sss-hh:mm". The second component (shown here as "ss.sss") could have any number of digits in its fractional part to allow any kind of precision beyond seconds.

7.1.15 (European) URI primitive

URI is a string following the rules defined by the W3C/IETF URI Planning Interest Group in a set of RFCs of which one is RFC 3305.

7.1.16 (rdf) Statements compound

It represent a set of Definition and/or Description elements.
Table 8 shows all attributes of Statements.

Table 8 – Attributes of DocumentHeaderProfile::Statements

name	mult	type	description
subject	1..1	String	Statement subject.
predicate	1..1	String	Statement predicate.
object	1..1	String	Statement object.

7.2 Reference metadata profile

[left blank intentionally – placeholder for next version]

Annex A: Document header and model exchange

A.1 General

Due to the present stage of development and standardisation of approaches related to metadata and document header information the defined solution in this document is considered as a transitory solution. Taking into account this nature it is necessary to clarify some assumptions that are applied when designing the solution. The aim of this section is to bring clarity of some of the attributes in the document header that were protentional misused in past and current model exchanges.

A.2 Modelling authority set, model and their versions

Modelling authority set (MAS) is seen as an abstract entity. It is more related to the sender of the information as it is linked and maintained by the sender of the information. A utility, a TSO, can have multiple abstract entities (in terms of W3C provenance Entity) to represent different scope. These could be MAS for planning, MAS for operation, MAS for asset, etc. The choice is up to the utility how to internally organise. Each of these abstract MAS entities have their versions which are important for the sending party to understand where models that are associated to a given MAS fit. This needs to be considered together with the knowledge of the definition of a model. The set of data governed by a profile is considered a model. For instance, data that relate to equipment profile is a model, data that relate to state variables profile is a model. However, a collection of equipment and state variables is also a model that can be called individual grid model.

Depending on the nature of models that are part of a version of a MAS, the MAS will be considered as a kind of envelop for models and will contain information where the MAS fits in the overall model exchange framework.

Note that the MAS or its version is not directly identifying the agent that is responsible for it or belonging to region or process, as this is more or less the current practice due to lack of other mechanisms to express that information. However, receiving party can retrieve information to the version of the modelling authority set via the reference data in case the business process agrees that this information is maintained in the master reference data. In this way, additional data such as the name of the agent, its location, role, models part of the version of the mas, contact information, other dependencies can be retrieved from the reference data.

Figure 9 is illustrating the relationship between a utility (agent), the modelling authority set, its versions, models part of a version of MAS and the versions of the models. Please note that some part of the abstraction is not explicitly included in the figure in order not to confuse. Also, only the terms related to W3C provenance are indicated as the link to W3C DCAT will make the view more complex. More detailed information will be part of the standardisation efforts which will be dealing with overall framework.

When using master reference data, the concept is that a document header or a manifest document (in the future) would refer to an identification of a version of a MAS. The URN, IRI or URL of the version of the MAS is part of the reference data and when that data is consulted (queried) additional information about the version of the MAS can be collected. Such information can be the name of the utility (TSO), what models' types are part of this version of MAS, e.g. is it only EQ and SSH or also TP or DL can be part of it, etc.

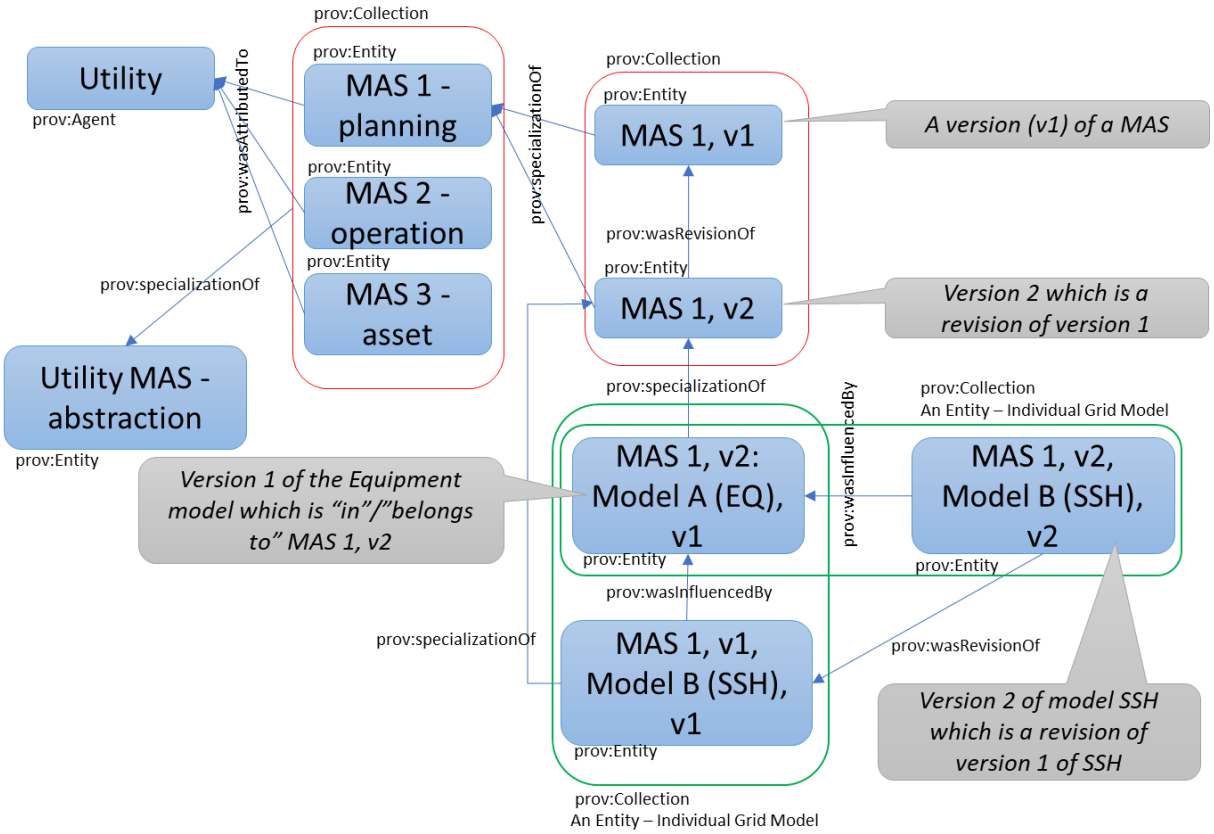


Figure 9. Modelling authority set, its version and model versions

A.3 Identification and versioning

Document header is primarily identifying the model which is serialised in the document. Therefore, the dct:identifier and rdf:about (in case of RDF serialization) of the document header are identical.

There are two attributes, which indicate the version is a human readable way. The md:version which in this version of the metadata indicates the version of the document and the owl:versionInfo which is related to the version of the model. It should be noted that to some extend the md:version was not used correctly and the current practice is to exchange the version of the document. The document header defined in this document tries to legalise this usage. The property owl:versionInfo shall follow the revisions of the model. In case a model changes its version, the identifier of the model changes, owl:versionInfo changes and the prov:wasRevisionOf refers to the identifier of the version which is superseded but the last version of the model. In case the modelling authority set version which produces the new version is not the same as the original MAS version, the prov:specializationOf refers to the MAS version which created the revised version of the model and prov:hadPrimarySource refers to the version of the MAS from which the previous version of the model originated.

Annex B (informative): Sample data

B.1 General

This Annex is designed to illustrate the profile by using fragments of sample data. It is not meant to be a complete set of examples covering all possibilities of using the profile. Defining a complete set of test data is considered a separate activity to be performed for the purpose of setting up interoperability testing and conformity related to this profile.

The sample data is not covering all possibilities of different references or information that can be provided.

B.2 Document header for available remedial action profile

```
<?xml version="1.0" encoding="utf-8"?>
<rdf:RDF
  xmlns:cim="http://iec.ch/TC57/CIM100#"
  xmlns:md="http://iec.ch/TC57/61970-552/ModelDescription/1#"
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:eu="http://iec.ch/TC57/CIM100-European#"
  xmlns:dct="http://purl.org/dc/terms/#"
  xmlns:dcat="http://www.w3.org/ns/dcat#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xmlns:time="http://www.w3.org/2006/time#"
  xmlns:eumd="http://entsoe.eu/ns/Metadata-European#"
  xmlns:owl="http://www.w3.org/2002/07/owl#"
  xmlns:csa="http://entsoe.eu/ns/csa#"
  xmlns:prov="http://www.w3.org/ns/prov#">

  <!--Header -->
  <md:FullModel rdf:about="urn:uuid:d2630bd5-9578-4fab-9647-13991c692d07"><!-- ID of the Full Model in RDF-->

    <!-- ID of the Full Model in Data Model-->
    <dct:identifier>urn:uuid:d2630bd5-9578-4fab-9647-13991c692d07</dct:identifier> <!--This is an example for mRID of the header -->

    <!-- creation time of the Documnent  -->
    <prov:generatedAtTime>2021-01-28T17:01:03Z</prov:generatedAtTime>

    <!-- Version of the Documnent -->
    <md:version>1</md:version>

    <!-- Validity/scenario period / delivery day [Optional]-->
    <time:inXSDDateTimeStamp>2021-11-25T17:00:00Z</time:inXSDDateTimeStamp>
    <time:hasXSDDuration>P1Y</time:hasXSDDuration>
```

680

```
    <!-- Description -->
    <dct:description>This is an example of available remedial action</dct:description>

    <!-- Profile, Schema or Specification -->
    <dct:conformsTo>http://entsoe.eu/ns/CIM/AvailableRemedialAction-EU/1.0</dct:conformsTo>
    <dct:conformsTo>http://entsoe.eu/ns/CIM/AvailableRemedialAction-EU/constraints/1.0</dct:conformsTo> <!--This is an example how to refer to SHACL constraints -->

    <!-- Message Type -->
    <dcat:keyword>PaneModel</dcat:keyword>

    <!-- md:Model.DependentOn -->
    <prov:wasInfluencedBy rdf:resource="urn:uuid:f0063d01-1dac-46f0-91a4-2b7479991173" />

    <!-- md:Model.Supersedes [OPTIONAL] (ID of pervious version of the Model) -->
    <prov:wasRevisionOf rdf:resource="urn:uuid:8341cd19-779b-4a84-bafb-06b8bb56f767" />

    <!-- Modeling Authority -->
    <prov:wasAttributedTo rdf:resource="urn:eic:10X1001A1001A094"/>

    <!-- Modeling Region -->
    <prov:atLocation rdf:resource="urn:eic:10YBE----------2"/>

    <!-- Process Type -->
    <eumd:processType rdf:resource="urn:entsoe.eu:ProcessTypeList#CSA"/>

    <!-- TimeFrame -->
    <dct:accrualPeriodicity rdf:resource="urn:entsoe.eu:wgedi:TimeFrameList#Y-1"/>

    <!-- Modelling Authority of the originator of the model -->
    <dct:creator>urn:eic:10X1001A1001A094</dct:creator>

    <!-- Confidentaliaty for Security Plan -->
    <dct:accessRights rdf:resource="http://entsoe.eu/MVS/2016/Confidentialyt/OPDE_Secret"/>

  </md:FullModel>

<!-- Here below is the content of the ARA (available remedial action) instance data -->
```

681

```
</rdf:RDF>
```

682

B.3 Reference metadata

684 *[left blank intentionally – placeholder for next version]*