Identity discussion

Date: 2023-12-11

Version: 0.8

Contents

[1. Related Redmine issues 1](#_Toc153213755)

[2. UML Information model (61970-301) 1](#_Toc153213756)

[2.1. 301/UML Information model 1](#_Toc153213757)

[Name root class 2](#_Toc153213758)

[2.2. **Discussion point 1**: Changes to .mRID attribute 3](#_Toc153213759)

[2.3. **Discussion point 2**: alternative (compromise) 4](#_Toc153213760)

[2.4. **Discussion point 3**: Definition of deprecated 5](#_Toc153213761)

[2.5. Resulting "agreed" model including "European" extensions part of -301 5](#_Toc153213762)

[2.6. **Discussion point 4**: Additional proposal 7](#_Toc153213763)

[2.7. Inf package clean up 8](#_Toc153213764)

[3. Profiles 9](#_Toc153213765)

[3.1. Profiles (61968-3) 9](#_Toc153213766)

[3.2. Profiles (61970-45x) 9](#_Toc153213767)

[3.3. ObjectRegistry Profile 9](#_Toc153213768)

# Related Redmine issues

* mRID topics: [CIM Issues #5963: mRID topics [incl GMDM #19] - CIM Joint Issues - UCAIug Issue Tracking System](https://redmine.ucaiug.org/issues/5963)
* [CIM Issues #5978: Remove serialization instructions from the UML description of mRID - WG13 Issues - UCAIug Issue Tracking System](https://redmine.ucaiug.org/issues/5978)
* [CIM Issues #6630: Update Identity and naming description - WG13 Issues - UCAIug Issue Tracking System](https://redmine.ucaiug.org/issues/6630)
* [CIM Issues #6632: update definition of deprecated - WG13 Issues - UCAIug Issue Tracking System](https://redmine.ucaiug.org/issues/6632)
* [CIM Issues #6634: Simplify Name model using enum - WG13 Issues - UCAIug Issue Tracking System](https://redmine.ucaiug.org/issues/6634)
* … [maybe we have some more]

# UML Information model (61970-301)

## 301/UML Information model

Modelling (CIM18v08) – prior Join meeting in Minneapolis :



Below is the description of some of the changes that was done as part of the group work:

### Name root class

CIM17:

"

The Name class provides the means to define any number of human readable names for an object. A name is not to be used for defining inter-object relationships. For inter-object relationships instead use the object identification 'mRID'.

"

CIM18v08:

"

The Name class, in possible combination with a name type and a naming authority provides the means to define any number of names or alternative identifiers for an object.

"

CIM17 attributes of Name.

Table 114 – Attributes of Core::Name

| name | mult | type | description |
| --- | --- | --- | --- |
| name | 0..1 | [String](#UML62) | Any free text that name the object. |

CIM18v08 attributes of Name.

Table 114 – Attributes of Core::Name

| name | mult | type | description |
| --- | --- | --- | --- |
| name | 0..1 | [String](#UML62) | Any free text that used as a name or alternative identifier of the object. |
| language | 0..1 | [String](#UML62) | Shall be specified as an IETF BCP 47 language tag (e.g. en-US).  Applies to the Name.name attribute. IETF language tags combine subtags from other standards such as ISO 639, ISO 15924, ISO 3166-1, and UN M.49. The tag structure has been standardized by the IETF in Best Current Practice (BCP) 47; the subtags are maintained by the IANA Language Subtag Registry. |

CIM17 association ends of Name with other classes.

Table 115 – Association ends of Core::Name with other classes

| mult from | name | mult to | type | description |
| --- | --- | --- | --- | --- |
| 0..\* | IdentifiedObject | 1..1 | [IdentifiedObject](#UML80) | Identified object that this name designates. |
| 0..\* | NameType | 1..1 | [NameType](#UML83) | Type of this name. |

CIM18v08 shows all association ends of Name with other classes.

Table 115 – Association ends of Core::Name with other classes

| mult from | name | mult to | type | description |
| --- | --- | --- | --- | --- |
| 0..\* | IdentifiedObject | 0..1 | [IdentifiedObject](#UML80) | Identified object that this name designates. |
| 0..\* | UniqueIdentifiedObject | 0..1 | [IdentifiedObject](#UML80) | Identified object that this alternative identifier designates. |
| 0..\* | NameType | 0..1 | [NameType](#UML83) | Type of this name. |

CIM17 all association ends of IdentifiedObject with other classes.

Table 109 – Association ends of Core::IdentifiedObject with other classes

| mult from | name | mult to | type | description |
| --- | --- | --- | --- | --- |
| 0..1 | DiagramObjects | 0..\* | [DiagramObject](#UML1055) | The diagram objects that are associated with the domain object. |
| 1..1 | Names | 0..\* | [Name](#UML82) | All names of this identified object. |

CIM18v08 all association ends of IdentifiedObject with other classes.

Table 109 – Association ends of Core::IdentifiedObject with other classes

| mult from | name | mult to | type | description |
| --- | --- | --- | --- | --- |
| 0..1 | DiagramObjects | 0..\* | [DiagramObject](#UML1055) | The diagram objects that are associated with the domain object. |
| 0..1 | Name | 0..\* | [Name](#UML82) | All names of this identified object. |
| 0..1 | AlternativeIdentfier | 0..\* | [Name](#UML82) | All alternative identifiers of this identified object. No two identified objects can have the same alternative identifier. |

## **Discussion point 1**: Changes to .mRID attribute

**Current mRID description (specified for the above mRID attributes that appears in the above CIM17 UML diagram):**

“Master resource identifier issued by a model authority. The mRID is unique within an exchange context. Global uniqueness is easily achieved by using a UUID, as specified in IETF RFC 4122, for the mRID. The use of UUID is strongly recommended.

~~For CIMXML data files in RDF syntax conforming to IEC 61970-552, the mRID is mapped to rdf:ID or rdf:about attributes that identify CIM object elements.~~”

The deleted part above is due to ticket: [CIM Issues #5978: Remove serialization instructions from the UML description of mRID – WG13 Issues – UCAIug Issue Tracking System](https://redmine.ucaiug.org/issues/5978)

**In addition, it is proposed that we update to the following, where we have some text added in red:**

"Master resource identifier issued by a model authority. The mRID is unique within an exchange context. Global uniqueness is easily achieved by using a UUID, as specified in IETF RFC 4122, for the mRID. The use of UUID is strongly recommended.

In the case of non-UUID (represented as a string), avoid the use of non-ASCII characters, space (" "), colons (":"), question marks ("?"), control characters (such as newline, tab or other non-printable characters) or symbols."

This proposal from Minneapolis has been added to ticket: [CIM Issues #5978: Remove serialization instructions from the UML description of mRID – WG13 Issues – UCAIug Issue Tracking System](https://redmine.ucaiug.org/issues/5978)

For classes that are not inheriting from **IdentifiedObject** it is then necessary to add the attribute **mRID** as shown in **Name**, **NameType** and **NamingAuthority** classes. This is a preferred proposal primarily to avoid minimum changes to implemented solution.

Regardless of decision we would need to make update to -301 to describe the difference between identifier and name. [CIM Issues #6630: Update Identity and naming description - WG13 Issues - UCAIug Issue Tracking System](https://redmine.ucaiug.org/issues/6630)

## **Discussion point 2**: alternative (compromise)

* Add a class **Identity** with **mRID** as the only attribute.
* Tag **IdentifiedObject.mRID** and **Name.mRID**, **NameType.mRID** and **NamingAuthority.mRID** as Deprecated.
* Make the classes **Name**, **NameType**, **NamingAuthority** and all classes that should be identifiable (can be changed) inheriting from **Identity.**



This proposal is backed by the discussion that happened in WG14 on 31 Oct 2023, namely

31-Oct-2023: We had a WG14 call the Tuesday after our Hybrid meetings (Chris Kardos, Svein, Jim Horstman, Eric Stephan, Jugoslav Dujic, Michael Johnson, Henry Dotson & Todd Viegut were in attendance). Revisited this in detail with a broader audience then last week (plus for those not able to attend).

* + - Action Item [Part 100 team]: Will change the following:
			* mRIDs will not be required to be UUIDs in the information model, but should be enforced in the profiles that are able to do it.
			* the new Name, NameType, NameAuthority classes will inherit from Identity and incorporated into both the Message.xsd and the ObjectIdentificationRevisionSet.xsd (clean up both profiles and publish both.)
			* (Note that IdentifiedObject will be inheriting from Identity with the proposal from last week.)
			* Since only Part-100 are using **Name.mRID**, **NameType.mRID** and **NamingAuthority.mRID** they can be removed from the information model and the Part-100 will use Identiy.mRID.
			* We will then issue an Amendment with this as a “fix”

Note that: IdentifiedObject.aliasName should not have been tagged as <<deprecated>> is should be <<deleted>> and be part of the Name/NameType.

## **Discussion point 3**: Definition of deprecated

IEC 61970-301 defines <<deprecated>> as:

“4.7.5 Deprecations

The CIM model may at times contain packages, classes, attributes, or associations that have been identified as deprecated. These items will be noted in the documentation or with a UML stereotype of “deprecated”. A deprecated item is retained in the present version of the model, but is expected to be removed from future versions. An item that has not been deprecated is not guaranteed to be retained in future versions, but using an item marked as deprecated should be avoided if possible."

**There is a proposal to add the following sentence to -301 section 4.7.5. This proposal needs to be discussed:**

"One reason for deprecating a class, attribute or association rather than removing it, could be to allow for the use of the item in an updated profile that is updated with non-breaking changes. New profile should not use any deprecated items."

This is addressed in ticket: [CIM Issues #6632: update definition of deprecated - WG13 Issues - UCAIug Issue Tracking System](https://redmine.ucaiug.org/issues/6632)

## Resulting "agreed" model including "European" extensions part of -301



**<<New>>**

**Identity**

**Note:** Root class that provides common identification for all specialised classes needing identification.

**<<update>>**

**IdentifiedObject**

**Note (old):** This is a root class to provide common identification for all classes needing identification and naming attributes.

**Note (updated):** Provide common identification, name and description for all specialised classes.

**The plan is to add description in -301 on instance data to describe the meaning of the Name and NameType:**

*The following tables will need to be added to the respective 301/4xx/600-x standards where applicable.*

*NOTE: Each respective IEC 61968 Part standard would provide their own table of reserved types if and as needed.*

*Reserved* ***NameType*** *values for already-deprecated and potentially future-deprecated* ***IdentifiedObject*** *attributes:*

| **IdentifiedObject Attribute** | **Value for: NameType.name** | **Value for: NameType.description** | **Description** |
| --- | --- | --- | --- |
| aliasName | ‘IdentifiedObject.aliasName’ | ‘Reserved by IEC TC57 WG13’ | The aliasName is free text human readable name of the object alternative to IdentifiedObject.name. It may be non unique and may not correlate to a naming hierarchy.The attribute aliasName is retained because of backwards compatibility between CIM releases. It is however recommended to replace aliasName with the Name class as aliasName is planned for retirement at a future time. |
| localName | ‘IdentifiedObject.localName’ | ‘Reserved by IEC TC57 WG13’ | A local short name of the instance. Objects that are structured in a functional naming hierarchy have this name local to each particular level in the hierarchy. The name shall be unique among objects contained by the same parent.localName has already been deprecated; however, an instance of Name with this NameType.name can be used to convey what used to be a localName. |
| name | ‘IdentifiedObject.name’ | ‘Reserved by IEC TC57 WG13’ | The name is any free human readable and possibly non unique text naming the object.There is no current intent to deprecate IdentifiedObject.name at any point in the future. |
| pathName | ‘IdentifiedObject.pathName’ | ‘Reserved by IEC TC57 WG13’ | Objects that are structured in a functional naming hierarchy have a pathName which contains all the IdentifiedObject.localNames from the object to the root. The pathName, then, is a concatenation of all these names from the leaf object up to the root of the containment hierarchy, similar to a file path name. For example, if node "A" contains node "B" that contains node "C", then the pathName for node "C" may look like "A.B.C". The type of delimiters used between localNames is not specified but is a local implementation issue. pathName has already been deprecated; however, an instance of Name with this NameType.name can be used to convey what used to be a pathName. |

*Reserved* ***NameType*** *values for transitioning of deprecated European* ***IdentifiedObject*** *extension attributes:*

|  |  |  |  |
| --- | --- | --- | --- |
| **IdentifiedObject Attribute** | **Value for:** **NameType.name** | **Value for: NameType.description** | **Description** |
| energyIdentCodeEic | ‘IdentifiedObject.energyIdentCodeEic’ | ‘Reserved by IEC TC57 WG13’ | This attribute is used for an exchange of the EIC code (Energy identification Code). The length of the string is 16 characters as defined by the EIC code. For details on EIC scheme please refer to ENTSO-E web site. |
| shortName | ‘IdentifiedObject.shortName’ | ‘Reserved by IEC TC57 WG13’ | The attribute is used for an exchange of a human readable short name with length of the string 12 characters maximum. |

## **Discussion point 4**: Additional proposal – Simplify Name model using enum

[CIM Issues #6634: Simplify Name model using enum - WG13 Issues - UCAIug Issue Tracking System](https://redmine.ucaiug.org/issues/6634)

The current agreement is not particularly good. We needed to keep the current structure to make minimum changes to the existing profiles in Part-100, Part-3 and Part-6 (?).

However, as we have now decided to create an ObjectRegistry profile as part IEC 61970-452 (EQ) it would make sense that we make this forward compatible by adding relevant attributes and classes and make existing item deprecated until existing profile are updated.

Therefore, Svein Olsen proposes the following information model:

 

By using enum rather than "name" we can do schema based validation rather than instanse based validation. This is very important when it comes to performance.

Proposed attributes of Name.

Table 114 – Attributes of Core::Name

| name | mult | type | description |
| --- | --- | --- | --- |
| name | 0..1 | [String](#UML62) | Any free text that used as a name or alternative identifier of the object when it deviate from the default name and identification.. |
| language | 0..1 | [String](#UML62) | Shall be specified as an IETF BCP 47 language tag (e.g. en-US).  Applies to the Name.name attribute. IETF language tags combine subtags from other standards such as ISO 639, ISO 15924, ISO 3166-1, and UN M.49. The tag structure has been standardized by the IETF in Best Current Practice (BCP) 47; the subtags are maintained by the IANA Language Subtag Registry. |
| Kind | 0..1 | NameKind | Kind of name. |
| nameType | 0..1 | [String](#UML62) | Describe a non-standard name type. |
| namingAuthority | 0..1 | [String](#UML62) | Authority responsible for creation and management of the given name. This is typically an organization or an enterprise system. |
| namingReference | 0..1 | URI | Reference to the name and/or naming authority. |

### NameKind enumeration

Kind of Name.

all literals of NameKind.

 Literals of Core::NameKind

| literal | value | description |
| --- | --- | --- |
| 61850name |  | Name is an alternative name following the 61850 naming convention. |
| alternativeName |  | Name is an alternative name that is governed by an application named in namingAuthority. It is a free text human readable of the object alternative to IdentifiedObject.name. It may be non unique and may not correlate to a naming hierarchy |
| alternativeIdentifier |  | Name is an alternative identifier that is governed by an application named in namingAuthority. It expected to be unique when combined with namingAuthority. |
| alternativeUniqueIdentifier |  | Name is a unique alternative identifier that is governed by an organisation or algorithm, given by namingAurhority, that ensure uniqueness. |
| aliasName |  | Name is free text human readable name of the object alternative to IdentifiedObject.name. It may be non unique and may not correlate to a naming hierarchy. |
| devicelName |  | Name is an alternative name that is given by a device and governed by the manufactory. The reference to the manufacture is given in namingAuthority. |
| displayName |  | Name is preferred display name shown to the uses in online screens and diagrams. |
| energyIdentCodeEic |  | Name is an alternative unique identifier that conforms to IANA urn-format EIC (urn:eic).  |
| iCCPname |  | Name is an alternative name used for ICCP exchange. |
| localName |  | Name is an alternative name that is unique among object contained by the same containment (e.g. cim:Bay).  |
| other |  | Name is an alternative name that cannot be described with the current name type. |
| pathName |  | Name is an alternative name that is structured in a functional naming hierarchy that forms the path through the containment.  |
| rDS |  | Name is an alternative name that conforms to Reference Designation System (RDS) Power Systems, RDS-PS defined in ISO/IEC 81346-10. |
| shortName |  | Name is an alternative name when the length for exchanging and use the name is constrained to be with length of the string 12 characters maximum. |
|  |  |  |

## Inf package clean up

The following diagram should be deleted as we now have covered it in CIM18.

Model\CIM\Grid\InfGrid\InfENTSOEextensionsNetworkCodes\ExtNetworkCodes\ExtObjectRegistry



# Profiles

## Profiles (61968-3)

The existing profiles in 61968-series can use the deprecated classes and attributes in any updated based on CIM18 so that there a full backwards compatibility. However, the profile should include note that in upcoming release there might be a move to the new model class.

## Profiles (61970-45x)

The existing profiles in 61970-series uses the deprecated IdentifiedObject.mRID in any updated based on CIM18 so that there a full backwards compatibility. The new ObjectRegistry profile (in -452) uses the new class, attributes and association so that we do not need to make any update to this profile for upcoming CIM versions particularly related to replacing IdentifiedObject.mRID with Identity.mRID.

In the information RDFS we declare that IdentifiedObject.mRID is that same as Identity.mRID:

<rdf:Description rdf:about="#IdentifiedObject.mRID">

 <rdf:type rdf:resource = "http://www.w3.org/2002/07/owl#DatatypeProperty" />

 <rdfs:label xml:lang="en">mRID</rdfs:label>

 <rdfs:domain rdf:resource="#IdentifiedObject"/>

 <rdfs:range rdf:resource = "http://www.w3.org/2001/XMLSchema#string" />

 <skos:definition xml:lang="en">

Master resource identifier issued by a model authority. The mRID is unique within an exchange context. Global uniqueness is easily achieved by using a UUID, as specified in RFC 4122, for the mRID. The use of UUID is strongly recommended.

In the case of non-UUID (represented as a string), avoid the use of non-ASCII characters, space (" "), colons (":"), question marks ("?"), control characters (such as newline, tab or other non-printable characters) or symbols.

 </skos:definition>

 <owl:equivalentProperty rdf:resource = "#Identity.mRID" />

</rdf:Description>

## ObjectRegistry Profile

For information the Network Code ObjectRegistry profile is defined by ENTSO-E as:



If the additional proposal is agreed the CIM18 ObjectRegistry profile to be included in 61970-452 is:

 

Example of adding EIC scheme code to an existing object would look like:

 <cim:Name rdf:ID="\_c528537a-541c-45bf-9e4e-1854287d5cdc">

 <cim:Name.name>10YBE----------2</cim:Name.name>

 <cim:Name.kind rdf:resource=" http://ucaiug.org/ns/CIM#NameKind.energyIdentCodeEic " />

 <cim:Name IdentifiedObject rdf:resource="#\_b3d1315a-a124-4b6e-9df5-85eddce9793b"/>

 <cim:Name.namingAuthority>ENTSO-E</ cim:Name.namingAuthority>

 <cim:Name.namingReference>https://www.entsoe.eu/data/energy-identification-codes-eic/eic-approved-codes/</ cim:Name.namingAuthorityReference>

 <cim:Identity.mRID>c528537a-541c-45bf-9e4e-1854287d5cdc</cim:Identity.mRID>

 </cim:Name>

# Comments

## Martin Miller [2023-12-06]

Regarding this proposal,

The three things I see that must be changed if we are going to move forward on this proposal:

1. The NameKind enumeration needs an entry for "name" (actual text in the enumeration TBD) to store the information that was previously in IdentifiedObject.name. This is absolutely required for interoperability with 61968 if we are not demanding changes from TF14.

2. The IdentifiedObject and Name classes still require two relationships between them, one for unique names and one for non-unique names. I don't understand why the second relationship was lost in the move from diagram 1 to diagram 2 in section 3.3

3. If we are not demanding changes from 61968, then we should do what we have the capacity to do in our own 61970 profiles to be consistent and stop using IdentifiedObject.name in our 452 and 456 profiles and upgrade to the newer Name-class way of doing things.

An on the other topic of contention:

Regarding the NameType to NameTypeAuthority relationship, this proposal does lose the ability to specify an authority for the type itself instead of just for the individual names. I could see this as potentially problematic.  However, I personally have never used or encountered a NameTypeAuthority in any real-world implementation that I've had to work with so far, so I personally don't care which way it goes. I'd be just as happy to see the authority go away completely, but I will defer to the participants who have use-cases for the authority.

-Martin