# ETSI TS 118 112 V2.0.0 (2016-09)

TECHNICAL SPECIFICATION

**oneM2M;**
**Base Ontology**
**(oneM2M TS-0012 version 2.0.0 Release 2)**

Reference
DTS/oneM2M-000012

Keywords
M2M, ontology

*ETSI*

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - NAF 742 C
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° 7803/88



### *Important notice*

The present document can be downloaded from:
http://www.etsi.org/standards-search

The present document may be made available in electronic versions and/or in print. The content of any electronic and/or print versions of the present document shall not be modified without the prior written authorization of ETSI. In case of any existing or perceived difference in contents between such versions and/or in print, the only prevailing document is the print of the Portable Document Format (PDF) version kept on a specific network drive within ETSI Secretariat.

Users of the present document should be aware that the document may be subject to revision or change of status. Information on the current status of this and other ETSI documents is available at
https://portal.etsi.org/TB/ETSIDeliverableStatus.aspx

If you find errors in the present document, please send your comment to one of the following services:
https://portal.etsi.org/People/CommiteeSupportStaff.aspx

# Contents

# Intellectual Property Rights

IPRs essential or potentially essential to the present document may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (https://ipr.etsi.org/).

Pursuant to the ETSI IPR Policy, no investigation, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

# Foreword

This Technical Specification (TS) has been produced by ETSI Partnership Project oneM2M (oneM2M).

# 1 Scope

The present document contains the specification of the oneM2M base ontology. A formal OWL representation of the base ontology can be found at http://www.onem2m.org/ontology/Base_Ontology.

The present document also specifies an instantiation of the base ontology in oneM2M resources which is required for generic interworking.

In addition the present document contains the functional specification for an Interworking Proxy Application Entity (IPE), the oneM2M resources and their usage for generic interworking.

Finally an example is given how external ontologies can be mapped to the base ontology. The example uses the Smart Appliances REFerence (SAREF) ontology (http://ontology.tno.nl/saref).

# 2 References

## 2.1 Normative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

Referenced documents which are not found to be publicly available in the expected location might be found at https://docbox.etsi.org/Reference/.

NOTE: While any hyperlinks included in this clause were valid at the time of publication, ETSI cannot guarantee their long term validity.

The following referenced documents are necessary for the application of the present document.

[1] ETSI TS 118 111: "oneM2M; Common Terminology (oneM2M TS-0011)".

[2] ETSI TS 118 001: "oneM2M; Functional Architecture (oneM2M TS-0001)".

[3] W3C Recommendation: "RDF 1.1 Concepts and Abstract Syntax".



## 2.2 Informative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

NOTE: While any hyperlinks included in this clause were valid at the time of publication, ETSI cannot guarantee their long term validity.

The following referenced documents are not necessary for the application of the present document but they assist the user with regard to a particular subject area.

[i.1] oneM2M Drafting Rules.

NOTE: Available at http://www.onem2m.org/images/files/oneM2M-Drafting-Rules.pdf.

[i.2] The Smart Appliances REFerence (SAREF) ontology.

NOTE: Available at http://ontology.tno.nl/saref/.

[i.3] Open-source ontology editor PROTÉGÉ.

NOTE: Available at http://protege.stanford.edu/.

[i.4] W3C OWL Working Group: "OWL 2 Web Ontology Language Document Overview".

NOTE: Available at http://www.w3.org/TR/owl2-overview/.

# 3 Definitions and abbreviations

## 3.1 Definitions

For the purposes of the present document, the terms and definitions given in ETSI TS 118 111 [1] and the following apply:

**annotation property:** property that can be used to add information (metadata/data about data) to classes, individuals and Object/Data Properties

**class:** OWL standard ontology language from the World Wide Web Consortium (W3C) (see [i.4]), Concepts are called "Classes"

**concept:** entity of an Ontology that has an agreed, well defined, meaning within the domain of interest of that ontology

NOTE: A Concept is conceptually grouping a set of Individuals.

**data property:** property that relates an individual of a Class to data of a stecified type and range

**interworked device:** non-oneM2M device (NoDN) for which communication with oneM2M entities can be achieved via an Interworking Proxy Application Entity (IPE)

**ontology:** formal specification of a conceptualization, that is defining Concepts as objects with their properties and relationships versus other Concepts

**generic interworking:** generic interworking allows interworking with many types of non- oneM2M Area Networks and Devices that are described in the form of a oneM2M compliant ontology which is derived from the oneM2M Base Ontology

NOTE: Generic interworking supports the interworking variant "full mapping of the semantic of the non-oneM2M data model to Mca" as indicated in clause F.2 of oneM2M TS-0001 [2].

**object property:** property that relates an individual of a domain Class to an individual of a range Class

**property:** in OWL standard ontology language Properties represent relations among individuals

NOTE: Properties can be sub-categorized as Object Properties, Data Properties and Annotation Properties.

**proxied device:** virtual Device (i.e. a set of oneM2M resources together with an IPE) that represents the Interworked Device in the oneM2M System

**relation:** (also called "interrelation" or "property") stating a relationship among individuals

**restriction:** describes a class of individuals based on the relationships that members of the class participate in

NOTE: Restrictions can be sub-categorized as: existential Restrictions, universal Restrictions, Cardinality restrictions and hasValue Restrictions.



## 3.2 Abbreviations

For the purposes of the present document, the abbreviations given in ETSI TS 118 111 [1] and the following apply:

| | |
|---|---|
| AE | Application Entity |
| OWL | Web Ontology Language |
| SAREF | Smart Appliances REFerence ontology |
| SPARQL | SPARQL Protocol and RDF Query Language |

# 4 Conventions

The key words "Shall", "Shall not", "May", "Need not", "Should", "Should not" in the present document are to be interpreted as described in the oneM2M Drafting Rules [i.1].

# 5 General information on the oneM2M Base Ontology (informative)

## 5.1 Motivation and intended use of the ontology

### 5.1.1 Why using ontologies in oneM2M?

#### 5.1.1.1 Introduction to ontologies

In a nutshell an ontology is a vocabulary with a structure. The vocabulary applies to a certain domain of interest (e.g. metering, appliances, medicine, etc.) and it contains concepts that are used within that domain of interest, similar to the "defined terms" in clause 3, "Definitions".

An ontology should:

- Capture a shared understanding of a domain of interest.
- Provide a formal and machine manipulable model of the domain.

The ontology lists and denominates these concepts which have agreed, well defined, meanings within the domain of interest (e.g. the concept of "Device" has an agreed, well defined, meaning within the scope of the Smart Appliances REFerence (SAREF) ontology see [i.2]).

Concepts do not identify individuals but they identify classes of individuals. Therefore, in the OWL standard ontology language from the World Wide Web Consortium (W3C) (see [3]), concepts are called "Classes".

The structure part of the ontology is introduced through agreed, well defined, relationships between its concepts. Such a relationship - in OWL called "Object Property" - links a *subject* concept to an *object* concept.

*subject* concept ➔ relationship ➔ *object* concept

in OWL:

*domain* Class ➔ Object Property ➔ *range* Class

EXAMPLE 1: In SAREF an Object Property "accomplishes" relates the "Device" class to the "Task" class:

Device ➔ accomplishes ➔ Task

Also the relationships/Object Properties of an ontology have agreed, well defined, meanings within the domain of interest. In the example above the "accomplishes" part of the relationship is well documented as part of SAREF (see [i.2]).

A second type of Properties in OWL is called "Data Properties". A Data Property is linking a subject Class to a data. These data may be typed or untyped.

EXAMPLE 2: in SAREF the Data Property "hasManufacturer" links the class "Device with data of datatype "Literal":

Device ➔ hasManufacturer ➔ Literal

Again, the Data Properties of an ontology have agreed, well defined, meanings within the domain of interest.
In the example 2, the Data Property "hasManufacturer" indicates that the Literal, that is linked via this Data Property will indicate the manufacturer of the Device.

Data Properties can be considered similar to attributes in oneM2M.

A third type of Properties in OWL is called "AnnotationProperties". An Annotation Property is used to provide additional information about ontology elements like classes and instances, which typically are external to the ontology and would not be used for reasoning. Example usages for such additional information are for providing a creator, a version or a comment. The object of an annotation property is either a data literal, a URI reference, or an individual.

In general, an individual of a certain Class may or may not have a particular relation (Object Property, Data Property or Annotation Property) that is defined by the ontology. However, if such a relation exists for the individual then that relation should be used with the meaning specified by the ontology.

One additional, crucial aspect differentiates an ontology from a vocabulary with a structure. An ontology enables specified, allowed constructs (based on predicate logic) and can be represented in a formal, machine interpretable form e.g. by the OWL standard ontology language. This allows the creation of queries (e.g. through the SPARQL query language) that search for individuals of specified classes, having specified relationships, etc.

The OWL flavour OWL-DL (where DL stands for "Description Logic"), that is used in the present document and that is supported by the ontology-editing tool "Protégé" (see [i.3]), has the additional advantage that it is underpinned by a description logic. For ontologies that fall into the scope of OWL-DL a reasoner can be used to automatically check the consistency of classes, take what has explicitly stated in the ontology and use it to infer new information. OWL-DL ensures that queries are decidable.

Additionally, OWL-DL allows the creation of Intersection, Union and Complement classes, restrictions (e.g. on the required/allowed number of relationships for any individual of the Class along this property) an other useful constructs.

### 5.1.1.2 The purpose of the oneM2M Base Ontology

#### 5.1.1.2.0 Introduction

Ontologies and their OWL representations are used in oneM2M to provide syntactic and semantic interoperability of the oneM2M System with external systems. These external systems are expected to be described by ontologies.

The only ontology that is specified by oneM2M is the oneM2M Base Ontology, as described in the present document. However, external organizations and companies are expected to contribute their own ontologies that can be mapped (e.g. by sub-classing, equivalence..) to the oneM2M Base Ontology.

Such external ontologies might describe specific types of devices (as e.g. in the SAREF ontology) or, more generally, they might describe real-world "Things" (like buildings, rooms, cars, cities.) that should be represented in a oneM2M implementation. The value for external organizations and companies to provide their ontologies to oneM2M consists in supplementing oneM2M data with information on the meaning/purpose of these data. The OWL representation of that ontology provides a common format across oneM2M.

The oneM2M Base Ontology is the minimal ontology (i.e. mandating the least number of conventions) that is required such that other ontologies can be mapped into oneM2M.

#### 5.1.1.2.1 Syntactic interoperability

Syntactic interoperability is mainly used for interworking with non-oneM2M devices in Area Networks. In this case an ontology - represented as an OWL file - that contains the Area Network specific types of communication parameters (names of operations, input/output parameter names, their types and structures, etc.) is used to configure an Interworking Proxy Entity (IPE).
With the help of this OWL file the IPE is able to allocate oneM2M resources (AEs, containers) that are structured along the Area Network specific parameters and procedures. This enables oneM2M entities to read/write from/into these resources such that the IPE can serialize the data and send/receive them from/to the devices in the Area Network.

The semantic meaning of these resources is implicitly given by the interworked Area Network technology.

Each ontology that describes a specific type of interworked Area Network needs to be derived from the oneM2M Base Ontology. In particular the device types of an ontology of an interworked Area Network need to be mapped (e.g. by sub-typing) into the concept "Interworked Device" of the oneM2M Base Ontology.

#### 5.1.1.2.2 Semantic interoperability

Semantic interoperability is mainly used to describe functionality for services provided by oneM2M compliant devices (M2M Devices).
For example. different, oneM2M compliant types of washing machines may all perform a functionality like "washing-function" ,"drying-function", "select wash temperature"…, however the oneM2M resources (containers), through which these functions can be accessed, can have different resourceNames, child-structures and type of content.

In this case an ontology - represented as an OWL file -contains the specific types of the M2M Application Service and/or Common Service of the M2M Device (e.g. CRUD operation, resourceNames, child-structures and type of content, etc.) together with the functionality of that service (e.g. "washing-function").

Each ontology that describe a specific type of M2M Device needs to be derived from the oneM2M Base Ontology. In particular the device type needs to be mapped (e.g. by sub-typing) into the concept "Device" of the oneM2M Base Ontology.

## 5.1.2 How are the Base Ontology and external ontologies used?

### 5.1.2.1 Overview

This clause describes how an external ontology that is compatible with the Base Ontology can be used in a joint fashion.

NOTE: Further use of external ontologies is left to subsequent releases.

### 5.1.2.2 Introduction to usage of classes, properties and restrictions



An ontology consists of Properties and Classes.

Properties represent relationships, and link individuals from the specified domain (a class) to individuals from the specified range (another class). There are two main types of properties in the Base Ontology, object properties and data properties. An object property describes a relationship between two object individuals. A data properties describes a relationship between an object individuals and a concrete data value that may be typed or untyped.

Classes are interpreted as sets of individuals, and sometimes classes are also seen as a concrete representation of concepts. In the Base Ontology, a Class can be directly defined by the class name and class hierarchy or defined by the properties characteristics of the individuals in the class. The latter method is known as restriction. The classes defined by restriction can be anonymous, which contains all of the individuals that satisfy the restriction.

In the Base Ontology, the restrictions can be divided as existential restrictions, universal restrictions and cardinality restrictions:

- Existential restrictions describe classes of individuals that participate in at least one (some) relationship along a given property to individuals that are members of the class, e.g. since a Device (Class: Device) has at least one function (Object Property: hasFunction) (Class: Function) that this device accomplishes, then (Class: Device) is a subclass of the anonymous class of (Object Property: hasFunction) *some* (Class: Function).

- Universal restrictions describe classes of individuals that for a given property only have relationships along this property to individuals that are members of the class. For example, since a subclass "Watervalve" of (Class: Device) only has a function (Object Property: hasFunction) subclass "Open_or_Close_Valve" of (Class: Function), then (Class:Watervale) is a superclass of the anonymous class of (Object Property: hasFunction) *only* (Class: Open_or_Close_Valve).

- Cardinality restrictions describe classes of individuals that, for a given property, only have a specified number of relationships along this property to individuals that are members of the class.

### 5.1.2.3 Methods for jointly using the Base Ontology and external ontologies

If the Base Ontology is available and the external ontologies are compatible with the Base Ontology, the Base Ontology and the external ontologies can be jointly used in the following ways.

1) Classes and properties mapping:

- The names of the class and properties in different ontologies may be totally different, but the meanings of these class and properties can be relevant. Classes and proporties mapping is used to link the relevant classes and properties in different ontologies.

- The descriptions for the classes and properties mapping relationship of the Base Ontology and external ontologies can be given in an ontology or a semantic rule depending on the frequency of the usage. For the frequent cases, it is better to give the mapping description in an ontology, even in the Base Ontology.

- The classes and properties mapping can be based on the properties defined in OWL and RDFs, e.g. rdfs:subClassOf, owl:equivalentClass, for classifying the hierarchy of the classes and properties in Base Ontology and external ontologies. The inheritance from upper properties and classes will be implied according to the mapped hierarchy. For example, when a class A in an external ontology is mapped as a subclass of the class B in the Base Ontology, it implies that the properties of class B in the Base Ontology will be inherited by the class A in the external ontology.

Table 1 gives a simple example for classes and properties mapping between two ontologies.

**Table 1: An example for classes and properties mapping between two ontologies**

| Properties mapping | | | Classes mapping | | |
|---|---|---|---|---|---|
| property I | mapping relationship | property II | class I | mapping relationship | class II |
| OntologyB: hasSwitch | rdfs:subPropertyOf | OntologyA: hasOperation | OntologyB: appliance | rdfs:subClassOf | OntologyA:device |
| OntologyA: hasPower | owl:equivalentProperty | OntologyB: hasPower | OntologyB:lamp | owl:equivalentClass | OntologyA:light |
| OntologyA: hasVendor | owl:equivalentProperty | OntologyB: hasManufacturer | OntologyB:Switch | rdfs:subClassOf | OntologyA:Operation |



2) Individual annotation across multiple ontologies:

- Though the names of the class and properties in different ontologies may be totally different, the semantic annotation for individuals can be done based on these different ontologies respectively and independently. In this way, the knowledge from different ontologies are used together to describe the individuals.

Table 2 gives a simple example for individual annotation across two ontologies.

**Table 2: An example for individual annotation across two ontologies**

| Individuals | Semantic annotation based on Ontology A | | Semantic annotation based on Ontology B | |
|---|---|---|---|---|
| | Properties | classes | properties | Classes |
| Light A | rdf:type | Ontology A: Light | rdf:type | Ontology B:ledLight |
| | OntologyA: hasOperation | Ontology A:Open | OntologyB: hasColor | rdf:datatype="&xsd;string">'red'< |
| | OntologyA: hasStatus | rdf:datatype="&xsd;boolean">true< | OntologyB: hasSwitch | OntologyB:Switch |
| NOTE: The two methods can be used jointly or independentlly. | | | | |

The compatibility of two ontologies depends on their class hierarchies. When the class hierarchy of one ontology can be mapped as a part or an external part of the class hierarchy of the other ontology, they are compatible. When multiple ontologies are pairwise compatible, they are compatible.

# 5.2 Insights into the Base Ontology

## 5.2.1 General design principles of the Base Ontology

### 5.2.1.1 General Principle

The Base Ontology has been designed with the intent to provide a minimal number of concepts, relations and restrictions that are necessary for semantic discovery of entities in the oneM2M System. To make such entities discoverable in the oneM2M System they need to be be semantically described as classes (concepts) in a - technology/vendor/other-standard specific - ontology and these classes (concepts) need to be related to some classes of the Base Ontology as sub-classes.
Additionally, the Base Ontology enables non-oneM2M technologies to build derived ontologies that describe the data model of the non-oneM2M technology for the purpose of interworking with the oneM2M System.

The Base Ontology only contains Classes and Properties but not instances because the Base Ontology and derived ontologies are used in oneM2M to only provide a semantic description of the entities they contain.
Instantiation (i.e. data of individual entities represented in the oneM2M System - e.g. devices, things, etc.) is done via oneM2M resources

The Base Ontology is available at the web page:

- http://www.onem2m.org/ontology/Base_Ontology;

which contains the latest version of the ontology and individual versions of the ontology (see Annex A).