|  |  |
| --- | --- |
| Input contribution  Use case | |
| Use Case Title:\* | Use case for semantic mapping between IoT ontology and object identifiers |
| Group Name:\* | RDM#42 |
| Source:\* | CMCC |
| Contact: | Xiaotao Li ( [lixiaotao@chinamobile.com](mailto:lixiaotao@chinamobile.com))  Yawen Niu ([niuyawen@chinamobile.com](mailto:niuyawen@chinamobile.com)) |
| Date:\* | 2019-09-18 |
| Abstract:\* | Propose to add the use case for semantic mapping between IoT ontology and object identifiers. In some smart scenarios (e.g. smart home, self-service store, etc.), IoT objects with heterogeneous identifiers need to work together without the human intervention based on the common knowledge defined in an IoT ontology. It requires oneM2M system to provide machine understandable semantic information for object identifiers to map the identified objects to corresponding classes in the IoT ontology and enable objects to identify what the interactive object is independently. |
| Agenda Item:\* |  |
| Work item(s): | WI 0015 - oneM2M Use Case Continuation |
| Document(s)  Impacted\* | Technical Specification TR 0001 - oneM2M Use Case Technical Report |
| Intended purpose of  document:\* | Decision  Discussion  Information  Other <specify> |
| Decision requested or recommendation:\* | Approval of the Use Case |
| Template Version:23 February 2015 (Dot not modify) | |

**oneM2M Notice**

The document to which this cover statement is attached is submitted to oneM2M. Participation in, or attendance at, any activity of oneM2M, constitutes acceptance of and agreement to be bound by terms of the Working Procedures and the Partnership Agreement, including the Intellectual Property Rights (IPR) Principles Governing oneM2M Work found in Annex 1 of the Partnership Agreement.

## Title

Use case for semantic mapping between IoT ontology and object identifiers.

### Description

With the development of communications and artificial intelligence technologies, the value of Internet of Things (IoT) is no longer limited to ubiquitous communication networks but providing intelligent services to human beings. These intelligent services appear in many representative scenarios without the aid of human intervention, such as smart home, self-service store, etc. To achieve these scenarios, objects with heterogeneous identifiers (e.g. EPC, Handle, OID, ucode, mCode, Ecode, etc.) need to cooperate with each other without the human intervention. However, all the existing identifiers appear as a sequence of characters and generally lack semantic information. IoT objects cannot identify what the interactive or cooperative object is through the identifier alone. Therefore, oneM2M system needs to enable semantic mapping between IoT ontology and identifiers to facilitate the IoT intelligence. Besides, the operations including create, retrieve, update, and delete (CRUD) of instances frequently occur in some smart scenarios. For example, in a self-service store, when a customer enters a store, a new instance belonging to the class “Person” should be added to the IoT ontology [automatically](javascript:;). If a commodity has been bought by a customer, this “commodity” instance should be deleted from the ontology. It is impractical to achieve these instance operations manually. It requires a semantic mapping between the ontology and the identifiers of objects, which will help the IoT system to manage the instances in an unattended manner.

Take a coffee with an EAN-13 identifier (8938515483013) for example. The oneM2M platform can add semantic information (1.4.4.2) to the legacy identifier, which indicates a path from the root node to a class node in the IoT ontology. From the semantic information (1.4.4.2), this object can be mapped to the class “Coffee” other objects can know this identified object is a **Coffee** instance.



Figure 1.1.1-1: A part of the IoT ontology

### Source

CMCC, Huawei

### Actors

* Application: the device or object which wants to be mapped to the IoT ontology through its identifier.
* The ontology is a vocabulary with a structure. It could capture a shared understanding of a domain of interests and provide a formal and machine interpretable model of the domain.
* The M2M service platform provided by the M2M service provider
  + The M2M service platform has a semantic mapping function to discover the associated class that an object belongs to. It’s a service layer functionality provided by the oneM2M System.

### Pre-conditions

The IoT ontology is required to be deployed on the M2M service platform.

The semantic information of identifiers is required to be related with the IoT ontology.

### Triggers

An object is required to be mapped to the IoT ontology as an instance or identify what the interactive object is through its identifier.

### Normal Flow

The normal message flow is described as follows:

Application

M2M service platform

2. extract semantic information from the identifier

3. discover the associated class in the ontology according to the semantic information

find the class related to the object through semantic mapping function

4. return the class that the object belongs to

1. request to be mapped to the ontology with the identifier

Figure 1.1.6-1: Message flow for sematic mapping between the ontology and an identifier

1. An application sends a request for mapping an object to the ontology or identifying an object to the M2M service platform, which contains the identifier of the object with semantic information added.

2. After receiving the request, the oneM2M platform extracts semantic information from the identifier.

3. The oneM2M platform then finds the class associated with the object based on the extracted semantic information of the identifier.

4. The oneM2M platform returns the name of the class to the application.

### Alternative flow

None.

### Post-conditions

None

### High Level Illustration



### Potential requirements

The oneM2M System shall be able to provide machine understandable semantic information for object identifiers to map the identified objects to corresponding classes in the IoT ontology and enable objects to identify what the interactive object is independently.