|  |
| --- |
| Input Contribution |
| Meeting ID\* | SEC 31 |
| Title:\* | Use Case 4 - IoT Data model mapping to oneM2M ontology  |
| Source:\* | Ian Deakin, iconectiv, ideakin@iconectiv.com  |
| Date:\* | 2017-09-21 |
| Input related to\* | WI-0073 - App-ID Registry Function |
| Intended purpose ofdocument:\* | [x]  Decision[ ]  Discussion[ ]  Information[ ]  Other <specify> |
| Impacted other TS/TR(s) | NA |
| Decision requested or recommendation:\* | Approve this section to the TR  |
| Template Version: January 2017 (Do 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.

# 5.x Use Case : IoT Data model mapping to oneM2M ontology

### 5.x.1 Description

IoT applications of the same type and functionality may all have different data models to represent their information with the Service Providers infrastructure. For example a light switch from different manufacturers may present one/off as 1/0 (One/Zero) or true/false, or some other representation. oneM2M has implemented an . Although oneM2M has defined data models/ontologies for specific IoT application class types, this may not be followed be the IoT application developer and or could contain proprietary extensions to differentiate itself in the market.

More so when the IoT is not natively oneM2M and is connected via an interworking function. The data representation may not have any representations which matching the defined oneM2M ontology definitions.

Although it is possible to do this in a manual process, it would require prior knowledge of the connecting IoT application type/model and possibly the software version.

**Using App-ID Registry Function to automate data model mapping to oneM2M ontologies**

By providing App-ID metadata that provides a description of the specific IoT application data model definition which describes the mapping to a compatible oneM2M ontology. Will enable a Service Provider to automate the process for enrolling the IoT application and being able to manage the data it produces or any interactions to control its functionality. The App-ID metadata can provide the following specific information:

* **Data Model:** The data definition for the application, what is projected, and the actions that can be performed. The application data model and mapping to oneM2M ontology.
* **Data privacy:** The data privacy asserted for this App-ID. The generic data privacy model for the application identity. For example, the IoT application is a blood pressure monitor and the data can only be provided to the client’s electronic health record. The data cannot be shared, data mined, or resold by the IoT service provider

By connecting with the App-ID Registry Function, a Service Providers infrastructure can query the metadata for a presented IoT application’s (AE-ID/App-ID) and retrieve a data model definition to both verify the data being used and to manage the permissible use.

**The benefits to the eco-system are:**

– Providing SEMANTIC interoperability

• Annotate M2M data with information, describing e.g.

– Name of the data (this could contain namespace / ontology).

– Relation to other M2M data.

• Abstraction from specific technologies

– Support of Data Brokering / Analytics / Big Data

• Semantic Discovery

– Can be supplemented with additional context information

• Data brokering (advertising available data / finding relevant data)

– Support of Big Data Analytics