ETSI TR 118 517 V2.0.0 (2016-09)

oneM2M;
Home Domain Abstract Information Model
(oneM2M TR-0017 version 2.0.0)

Reference

DTR/oneM2M-000017

Keywords

IoT, M2M

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 Report (TR) has been produced by ETSI Partnership Project oneM2M (oneM2M).

1 Scope

The present document allows application developers to describe the status of devices as resources on oneM2M-based platform in various ways. Thus different application developers can create different resource trees even when they build the same kinds of applications. Moreover when handling the same kinds of devices from different vendors on M2M platforms, application developers may create disunited resource trees without common information model.

In order to solve such issues, the present document intends to provide the common and unified APIs on one M2M platform for the home domain by defining an abstract information model for the home domain devices such as TV, refrigerator, air conditioner, smart meter, and lighting equipment. The definition of the abstract information model will be based on data models that currently exist in the home domain. The home domain abstract information model does not intend to define the interworking function between the oneM2M system and protocols of the data models from which the abstract data model is defined.

Also, the present document intends to define how the developed abstract information model for a device could be represented in the CSEs of the oneM2M system.

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.

Not applicable.

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] ETSI TS 118 111: "Definitions and Acronyms (oneM2M TS-0011)".

[i.3] AllJoyn System Description version 14.06, September 26th, 2014.

[i.4] Home Appliances & Entertainment (HAE) Service Framework Project.

NOTE: Available at https://wiki.allseenalliance.org/hae.

[i.5] HomeKit Developer Site.

NOTE: Available at https://developer.apple.com/homekit/.

[i.6] Designing Accessories for iOS and OS X, WWDC14.

NOTE: Available at https://developer.apple.com/videos/wwdc/2014/.

[i.7] ETSI TS 118 104: "Service Layer Core Protocol Specification (oneM2M TS-0004)".

[i.8] ECHONET Specification Ver.2.11.

NOTE: Available at http://echonet.jp/spec_v211_en/.

[i.9] ECHONET Lite Specification Ver.1.11.

NOTE: Available at http://echonet.jp/spec_v111_lite_en/.

[i.10] ECHONET APPENDIX, Detailed Requirements for ECHONET Device Objects.

NOTE: Available at http://echonet.jp/spec_object_rf_en/.

[i.11] IEC 62394: "Service diagnostic interface for consumer electronics products and networks - Implementation for echonet".

[i.12] ISO/IEC 24767-1: "Information technology -- Home network security -- Part 1: Security requirements".

[i.13] ISO/IEC 24767-2: "Information technology -- Home network security -- Part 2: Internal security services: Secure Communication Protocol for Middleware (SCPM)".

[i.14] IEC 62480: "Multimedia home network - Network interfaces for network adapter".

[i.15] ISO/IEC 14543-4-1: "Information technology -- Home electronic system (HES) architecture -- Part 4-1: Communication layers -- Application layer for network enhanced control devices of HES Class 1".

[i.16] ISO/IEC 14543-4-2: "Information technology -- Home electronic system (HES) architecture -- Part 4-2: Communication layers -- Transport, network and general parts of data link layer for network enhanced control devices of HES Class 1".

[i.17] IEC 62457: "Multimedia home networks - Home network communication protocol over IP for multimedia household appliances".

[i.18] ETSI TS 118 123: "Authorization Architecture for Supporting Heterogeneous Access Control Policies (oneM2M TS-0023)".

[i.19] ETSI TS 118 101: "Functional Architecture (oneM2M TS-0001)".

3 Abbreviations

For the purposes of the present document, the abbreviations given in ETSI TS 118 111 [i.19] and the following apply:

IoE Internet of Everything

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 Home Domain Abstract Information Model

5.1 Anaysis of Abstract Information Models

5.1.1 AllJoyn (AllSeen Alliance)

The AllJoyn system is a proximity-based, peer-to-peer communication platform that provides a framework for enabling communication among internet of everything (IoE) devices across heterogeneous distributed systems [i.3]. Especially, there is a specific project to develop the common way of controlling and monitoring Home Appliances & Entertainment (HAE) devices (such as airconditioner, refrigerator, washer), regardless of device manufacturers [i.4]. In order to achieve this issue, HAE service frame project specifies standard AllJoyn interfaces for controlling and monitoring HAE devices as shown in figure 5.1.1-1.

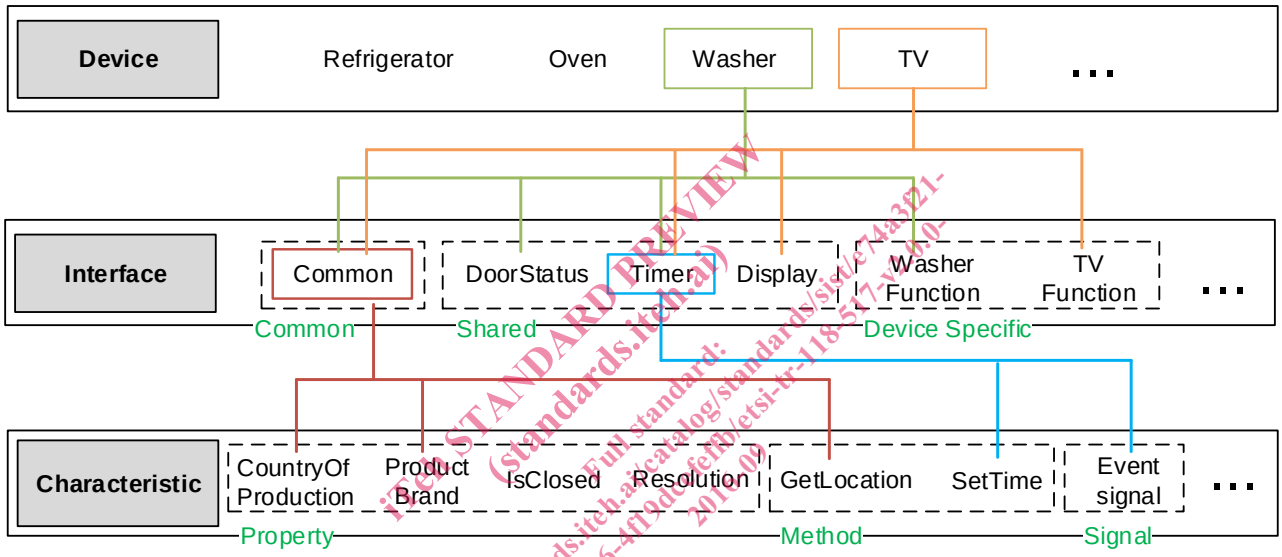

Figure 5.1.1-1: Information Model in AllJoyn

In the AllJoyn information model, 'Device' which represents physical home appliance (such as refrigerator, oven, washer, TV) should have 'Interface(s)' that consist of 'Characteristic(s)'.

'Characteristic' specifies pre-defined device properties(such as country of production, product brand, status of door, display resolution), method(such as getting location, setting the timer) and signal(similar to notification). By combining one or more pre-defined characteristic(s), a specific 'Interface' which provides a particular service(such as subscription or control of the device) can be defined. Furthermore, the AllJoyn system categorizes the 'Interface' into common interface, shared interface and device specific interface for more efficient management. Although the complexity is increased as arraging into the service set, this approach enhances the resource efficiency by resusing the 'Interface'.

5.1.2 HomeKit (Apple Inc.)

HomeKit is a framework in iOS for communicating with and controlling connected accessories in a home domain environment [i.5]. Among developments related document, HomeKit Accessory Profiles defines the information model for some services based on the 'Charateristic' and 'Service' [i.6].

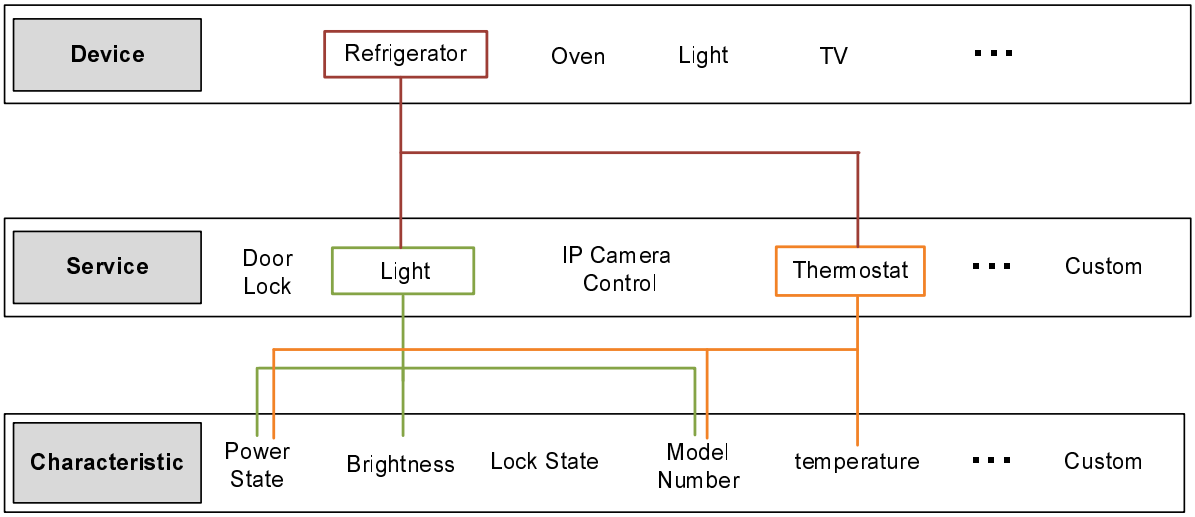

Figure 5.1.2-1: Information Model in HomeKit

As shown in figure 5.1.2-1, the chracteristic specifies pre-defined characteristics and their properties (such as power state, brightness, lock state, model number). Based on the chracteristic, a specific service (such as door lock, light, thermostat) can be defined one or more chracteristic(s). Especially, the 'Custom' value is considered for the un-predefined characteristic and service. As a result, a specific device may have one or more service set(s). Compared to the AllJoyn system approach, the characteristic set in the Homekit are equivalent to the characteristic in the AllJoyn. The HomeKit system has moduled service sets by using characteristic set(s). It is a very similar approach with the 'Interface' in the AllJoyn system.

5.1.3 SmartHome Device Template (HGI)

5.1.3.1 Introduction

The SDT (SmartHome Device Template) is an initiative from HGI to find consensus amongst various SDOs and industry alliances to derive a common approach for device modelling. HGI and partners have the approach to agree on a set of automation commands following a common syntax which are sufficient to model most home appliance functions. The key goals of the SDT are:

1) keep it simple, especially for manufacturers to contribute device information;
2) modularity for functions and device types;
3) make it easy for developers to create unified APIs;
4) be independent of underlying home-area network technologies;
5) enable extendibility of the system in place without service interruption;
6) allow a pass-through mechanism to enable use of proprietary or technology-specific functions.

The SDT approach is to define re-usable basic functions (or services) (labelled "ModuleClass" in figure 5.1.3.1-1) which can represent the typical functions found in many home automation systems, such as "on/off", "dim a lamp", "receive events from binary sensor", "read data from sensor", etc. Each ModuleClass is composed of a (small) number of actions, datapoint read/write operations, or asynchronous events. For example, an "on/off" ModuleClass would consist perhaps of just one Action, but a "ReadKeypad" Action might have a number of possible events, each with some data value and (usually) a sequence-ID or timestamp start/stop to indicate when and how long each key was pressed.

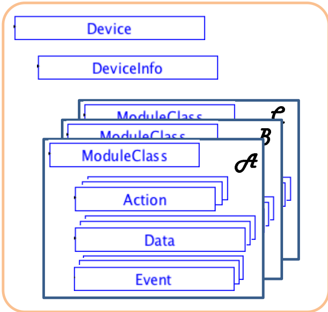

Figure 5.1.3.1-1: SmartHome Device Template (XSD) for a generic device (simplification of draft, under discussion with SDOs)

The SDT represents the device models introduced in figure 5.1.3.1-1 by using an XSD schema to allow formal checking of compliance for XML device descriptions of specific appliances. The modularity goal in the XSD schema is achieved with **re-usable XSD fragments** ("ModuleClass" in figure 5.1.3.1-1).

Complex devices or appliances can then be described by an appropriate set or collection of the agreed XSD fragments (ModuleClasses), as indicated in figure 5.1.3.1-1, which also shows an optional DeviceInfo XSD fragment to allow recording of static information such as device manufacturer name, device firmware version, etc.

Note that the SDT concept of ModuleClasses is similar to the HomeKit concept of "services".

HGI has discussed with many SDOs to validate the concept. Consultation with various industry fora continues, to determine an appropriate set of commonly used ModuleClasses, which however allows extensions. SDT is designed to take into account the list of "services" compiled by the SAREF project.

The SDT supports the use of a set of templates for generic devices or appliances (e.g. for a dimmable lamp, a basic washing machine, etc., which would be specific instances of the "Device" object shown in figure 5.1.3.1-2) which form the basis of APIs used by application developers. These templates can also be referenced by manufacturers creating XML documents to describe their specific products. For example, the SDT enables specification of a generic washing machine template, with on/off, set-wash-temperature, pause and a few other commands, which could be referenced by a manufacturer as the schema for a XML description of a basic model washing machine. The SDT allows for vendor-specific additional commands (ModuleClasses) to suit specific product types.

An example of how three different generic devices/appliances might be modelled using 4 different ModuleClasses is shown in figure 5.1.3.1-2. Data values (DataPoints) which might need to be read/written during operation of the devices are shown as the lowest grouping in the figure (DataPoints/Characteristics).

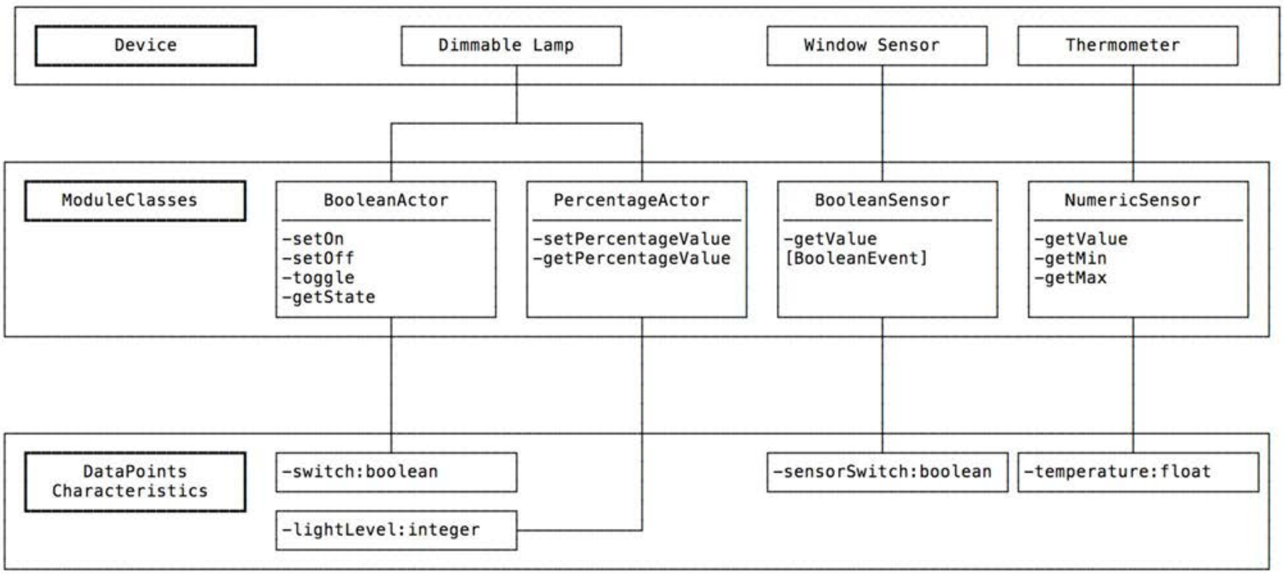

Figure 5.1.3.1-2: SmartHome Template for 3 examples of generic devices

Figure 5.1.3.4-1 showing the SDT structure in more detail is shown in clause 5.1.3.4. It is sufficiently flexible to allow representation of e.g. the SAREF ontology or the future OneM2M ontology.

5.1.3.2 Definitions

This clause provides an overview about the SDT 3.0 definitions and element hierarchy. Terms to be described in detail in this clause are:

Table 5.1.3.2-1: Definitions of SDT Elements

Term	Definition
Domain	Unique name, or "wrapper" which acts like a namespace, set by the organization creating the SDT, allowing reference to a package of definitions for the contained ModuleClasses and device definitions. Can be referenced when extending ModuleClasses. It has two possible uses: to select the scope of a technology domain, or to set the scope of a use case domain (like Home, SmartGrid, etc.).
Device	Physical, addressable, identifiable appliance/sensor/actuator.
Sub-Device	A device (usually one of several) which may be embedded in a Device and/or is addressed via another Device.
ModuleClass	Specification of a **single** service with one or more service methods, the involved abstracted data model and related events. The expectation is that each separate service which may be used in many kinds of Devices (like PowerON/OFF, Open/Close, etc.) will be described by a ModuleClass which can be re-used in many Device definitions.
Module	Instantiation of a ModuleClass for a specific Device or SubDevice.

5.1.3.3 Overview

Various details about recommended structure for SDTs are described in the next clauses. The key point to keep in mind is that HGI sought a compromise between, at the one extreme, complete flexibility (which could describe any device, of any complexity) and, at the other extreme, a rigid structure which could be 100 % validated and lead to validated software APIs.

A major decision, facilitating validation of code and signalling, was to describe services (functionality) of devices in terms of ModuleClasses made up of combinations of three kinds of elements:

a) DataPoints which can be read/written;

b) Actions which consist of more complex sequences of operations;

c) Events which can be signalled ("published") by devices asynchronously.

This structure shown in figure 5.1.3.3-1 and is given in more detail in figure 5.1.3.4-1.

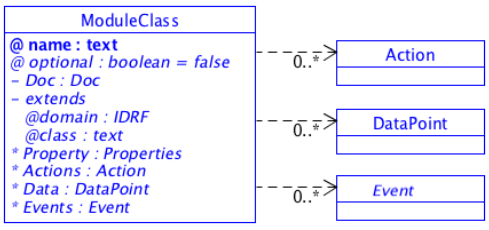

Figure 5.1.3.3-1: Diagram describing device functionality in terms of Actions, DataPoints, Events

5.1.3.4 Structure

The following UML diagram presents an overview of the structure (elements) of every SDT which is conformant with these guidelines. As implied in the above descriptions, there can be many different choices of the details of a SDT, each one optimized for a particular market segment and the types of devices used in that market segment. Obviously an unnecessary proliferation is counter-productive, but as long as each SDT conforms to the structure shown below then it will be possible with little or modest effort for software applications to be adapted accordingly.

The UML diagram below is in a sense **the** meta-format **for** different possible Smart Device Templates (XSDs) **for** device descriptions (XMLs) **of** real devices - sorry about that.

The syntax used in the diagram to model an XML Schema Definition (XSD) as an UML diagram follows the following approaches: Design XML schemas using UML and UML For W3C XML Schema Design.

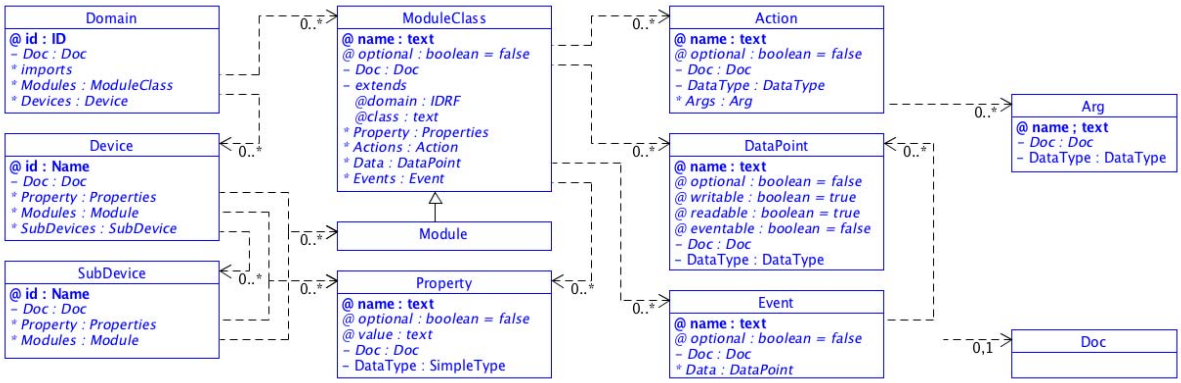

Figure 5.1.3.4-1: UML Diagram providing an overview of the SDT

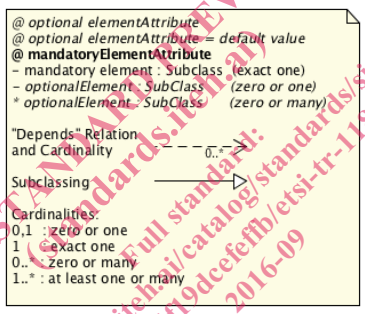

Figure 5.1.3.4-2: Annotations for UML Diagrams in SDT

Several constraints or design decisions were involved in creating the above template, which are describe below at the appropriate Element.

5.1.3.5 "Domain" element

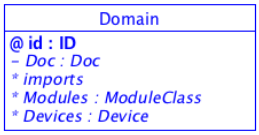

Figure 5.1.3.5-1

The "Domain" element allows labeling of different SDT templates for different technologies and/or industry segments ("verticals"): for example eHealth and Building Management might prefer quite different detailed structures/templates. This also helps keep information in human-friendly and manageable blocks. It is assumed that there will be multiple "SDT Templates" and some of them may be completely proprietary.

It can also be used to collect all specified ModuleClasses and Devices in one referencable logical group.