ETSI TS 118 123 V2.0.2 (2020-03)

oneM2M; Home Appliances Information Model and Mapping (oneM2M TS-0023 version 2.0.2 Release 2A)

Reference
RTS/oneM2M-000023v2A

Keywords
information model, 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 prevailing version of an ETSI deliverable is the one made publicly available in PDF format at www.etsi.org/deliver.

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

Essential patents

IPRs essential or potentially essential to normative deliverables 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.

Trademarks

The present document may include trademarks and/or tradenames which are asserted and/or registered by their owners. ETSI claims no ownership of these except for any which are indicated as being the property of ETSI, and conveys no right to use or reproduce any trademark and/or tradename. Mention of those trademarks in the present document does not constitute an endorsement by ETSI of products, services or organizations associated with those trademarks.

Foreword

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

1 Scope

The present document describes the oneM2M defined information model for home appliances, including the description of how it is mapped with other information models from external organizations. It also explains the ontology for the home domain information model.

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] Home Gateway Initiative Smart Device Template.

NOTE: Available at https://github.com/Homegateway/SmartDeviceTemplate/tree/7c890b69d9764e341ef1768c5a0e7d53a47cff5c.

[2] Java™ coding rule.

NOTE: Available at http://www.oracle.com/technetwork/java/codeconventions-135099.html.

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 TR 118 517: "oneM2M; Home Domain Abstract Information Model (oneM2M TR-0017)".

[i.3] ETSI TS 118 101: "oneM2M; Functional Architecture (oneM2M TS-0001)".

[i.4] IEEE 802.15.4™: "IEEE Standard for Local and metropolitan area networks -- Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs)".

[i.5] ETSI TS 118 112: "oneM2M; Base Ontology (oneM2M TS-0012)".

3 Definition of terms, symbols and abbreviations

3.1 Terms

For the purposes of the present document, the following terms apply:

Device Class ID: URN to identify the Device model definition

ModuleClass ID: URN to identify the ModuleClass model definition

3.2 Symbols

Void.

3.3 Abbreviations

Void.

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 Appliance Information Model

5.1 Introduction

The present document intends to provide the unified means in the oneM2M system by defining a home appliance information model for the home domain devices such as TV, refrigerator, air conditioner, clothes washer, oven, and robot cleaner. For the reasons of interworking with external technologies and efficiency, the principle of the home appliance information model is designed based on HGI SDT 3.0 [1].

The principle of defining the home appliance information model is introduced in clause 5.2. ModuleClasses which oneM2M systems support are explained in clause 5.3. In the subsequent clause 5.4, Device models are defined.

5.2 Design Principle of the Home Appliance Information Model

5.2.1 Basic design principle of information modelling

The design principle of the oneM2M abstract information model of home appliance, is to use SDT 3.0 as introduced in ETSI TR 118 517 [i.2]. Note that those terms starting with a capital letter in this clause are SDT terms and are explained in [1].

Domain is a unique name which acts like a namespace (e.g. "org.oneM2M.home.modules"). It is set by the organization creating the SDT, allowing reference to a package of definitions for the contained ModuleClasses and Device models.

ModuleClasses specifies a single service (e.g. audioVolume, powerOn/Off) with one or more Actions, Properties, DataPoints and Events. Each service which is described as a ModuleClass can be re-used in many Devices.

Device model is a physical, addressable, identifiable appliance, sensor and actuator with one or more Modules, Properties and SubDevices.

SubDevice is a device which may be embedded in a Device and/or is addressed via another Device.

Module is an instantiation of a ModuleClass for a specific Device or SubDevice.

Figure 5.2.1-1 depicts the basic structure of SDT 3.0 [i.2].

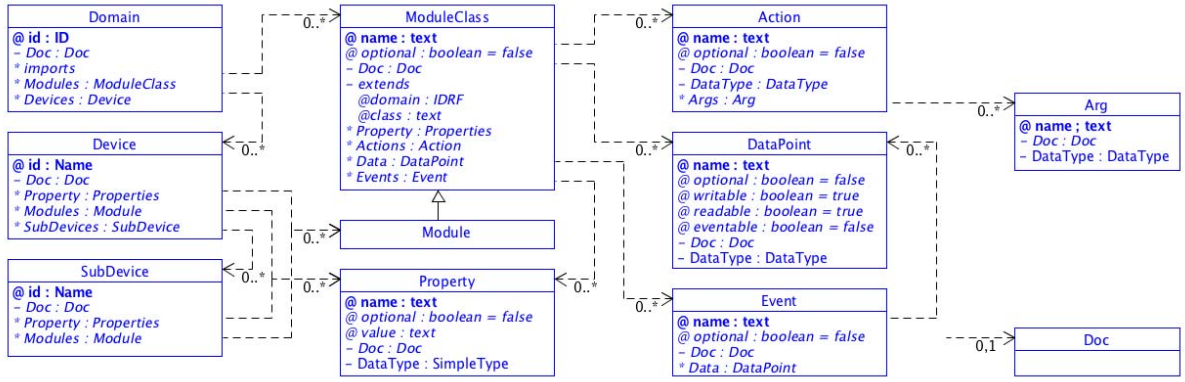

Figure 5.2.1-1: Design Structure of the Home Appliance Information Model using SDT 3.0

5.2.2 Description rules for Module Classes and Device models

When the Home Appliances Information Model is described based on SDT, the following rules shall be applied:

- Rule 1: CamelCase rule:
  - When naming each element, lowerCamelCase shall be used as the Java coding rules [2].
- Rule 2: Rule for description of Action, DataPoint:
  - DataPoint shall be used to represent stateless operations. (e.g. powerState of binarySwitch for on/off operations).
  - Action shall be used when describing stateful condition, handling unknown internal state conditions (e.g. upVolume/downVolume by increasing/decreasing the audioVolume in steps, handling transactional procedures, or checking integrity using username plus password at the same time).
- Rule 3: Rule for description of DataPoint and Property:
  - Non-functional information shall be described as a Property. Functional information shall be described as a DataPoint. (E.g. non-functional information: version, id; functional information: targetTemperature, targetVolume).
- Rule 4: Definition of the Domain:
  - The Domain, in the case of the Home Appliance Information Model, is specified as "org.onem2m.home".
  - The sub-domain for Device and ModuleClass shall be specified as "org.onem2m.home.devices" and "org.onem2m.home.moduleclasses" respectively.
- Rule 5: Naming rule for the element:
  - the name of each element should be concise and avoid repeating its parent element name; but
  - it may include the name of its parent element for readability (e.g. lightDimmerUp, lightDimmerDown under lightDimmer).
- Rule 6: Criteria for marking elements as optional or mandatory:
  - An element shall only be defined as mandatory if it is foreseen to be universally mandatory to all implementing technologies.

- Rule 7: Enumeration type:
  - When describing the meaning of values for enumeration type elements, they may be described in another clause.

The enumeration types for Home Appliance Information Model are based on <xs:integer>, and the numeric values are interpreted as specified in clause 5.5.

5.3 ModuleClasses

5.3.1 alarmSpeaker

This ModuleClass provides the capability to initiate an alarm.

Table 5.3.1-1: DataPoints of alarmSpeaker ModuleClass

Name	Type	Readable	Writable	Optional	Documentation
tone	hd:tone	true	true	true	Representing the tones of the alarm.
light	hd: alertColour Code	true	true	true	Representing the lighting mode of the alarm.
alarmStatus	xs:boolean	true	true	false	"True" indicates the alarm start while "False" indicates the alarm stop.

5.3.2 audioVideoInput

This ModuleClass provides capabilities to control and monitor audio video input source of device such as TV or SetTopBox.

Table 5.3.2-1: DataPoints of audioVideoInput ModuleClass

Name	Type	Readable	Writable	Optional	Documentation
inputSourceID	xs:integer	true	true	false	Activated input source ID in the supported input source list, supportedInputSources.
supportedInput Sources	list of hd:supportedInput Source	true	false	false	List of supported input sources for the given device (see clause 5.5.2).

5.3.3 audioVolume

This ModuleClass provides capabilities to control and monitor volume

Table 5.3.3-1: Actions of audioVolume

Return Type	Name	Argument	Optional	Documentation
none	upVolume	none	true	Increase volume by the amount of the stepValue up to the maxValue.
none	downVolume	none	true	Decrease volume by the amount of the stepValue down to 0.

Table 5.3.3-2: DataPoints of audioVolume

Name	Type	Readable	Writable	Optional	Documentation
volumePercentage	xs:integer	true	true	false	The rounded percentage of the current volume in the range of [0, maxValue]. 0 percentage shall mean no sound produced.
stepValue	xs:integer	true	false	true	Step value used by UpVolume and DownVolume Actions.
maxValue	xs:integer	true	false	true	Maximum value allowed for Volume.
muteEnabled	xs:boolean	true	true	false	The current status of the mute enablement. "True" indicates enabled, and "False" indicates not enabled.

5.3.4 battery

Battery indicates the detection of low battery and gives an alarm if triggering criterion is met. The charge value in the module shows the current battery charge level.

Table 5.3.4-1: DataPoints of battery ModuleClass

Name	Type	Readable	Writable	Optional	Documentation
level	xs:integer	true	false	false	The rounded percentage of the current level of battery in the range of [0, 100]. 0 percentage shall mean no battery remained.
capacity	xs:integer	true	false	true	The total capacity of battery in mAh.
charging	xs:boolean	true	false	true	The status of charging. "True" indicates enabled, and "False" indicates not enabled.
discharging	xs:boolean	true	false	true	The status of discharging. "True" indicates enabled, and "False" indicates not enabled.
lowBattery	xs:boolean	true	false	true	To indicate that the battery is in low charge level.
batteryThreshold	xs:integer	true	true	true	When the battery level is less than batteryThreshold then the lowBattery is true (and optionally to generate an alarm, see clause 5.3.1).

Table 5.3.4-2: Properties of battery ModuleClass

Name	Type	Value	Optional	Documentation
propElectricEnergy	xs:integer		true	Rated electric energy.
propVoltage	xs:integer		true	Rated voltage.
propMaterial	xs:string		true	The material (e.g. lithium ion, nickel and lead) of the cell.

5.3.5 binarySwitch

This ModuleClass provides capabilities to control and monitor the state of power.

Table 5.3.5-1: Actions of binarySwitch ModuleClass

Return Type	Name	Argument	Optional	Documentation
none	toggle	none	true	Toggle the switch.

Table 5.3.5-2: DataPoints of binarySwitch ModuleClass

Name	Type	Readable	Writable	Optional	Documentation
powerState	xs:boolean	true	true	false	The current status of the binarySwitch. "True" indicates turned-on, and "False" indicates turned-off.

5.3.6 bioElectricalImpedanceAnalysis

This ModuleClass provides the analysis of human body tissue based on impedance measurement.

Table 5.3.6-1: DataPoints of bioElectricalImpedanceAnalysis ModuleClass

Name	Type	Readable	Writable	Optional	Documentation
water	xs:float	true	false	false	The water content measurement from the BIA; the common unit is percentage.
fat	xs:float	true	false	false	The fat content measurement from the BIA; the common unit is percentage.
muscle	xs:float	true	false	false	The muscle content measurement from the BIA; the common unit is percentage.
bone	xs:float	true	false	false	The bone content measurement from the BIA; the common unit is percentage.
visceraFat	xs:float	true	false	false	The viscera fat content measurement from the BIA; the common unit is percentage.
kcal	xs:float	true	false	false	The kcal (kilocalories) measurement from the BIA.
resistance	xs:float	true	false	false	The resistance of human body; the common unit is ohm.

5.3.7 boiler

This ModuleClass provides the status of boiling function for water heaters.

Table 5.3.7-1: DataPoints of boiler ModuleClass

Name	Type	Readable	Writable	Optional	Documentation
status	xs:boolean	true	true	false	The status of boiling.

5.3.8 brightness

This ModuleClass describes the brightness of a light e.g. from a lamp. Brightness is scaled as a percentage. A lamp or a monitor can be adjusted to a level of light between very dim (0 % is the minimum brightness) and very bright (100 % is the maximum brightness).

Table 5.3.8-1: DataPoints of brightness ModuleClass

Name	Type	Readable	Writable	Optional	Documentation
brightness	xs:integer	true	true	false	The status of brightness level in percentage.

5.3.9 clock

This ModuleClass provides the information about current date and time.

Table 5.3.9-1: DataPoints of clock ModuleClass

Name	Type	Readable	Writable	Optional	Documentation
currentTime	xs:time	true	true	false	Information of the current time
currentDate	xs:date	true	true	false	Information of the current date

5.3.10 colour

This ModuleClass provides the capabilities to set the value of Red, Green, Blue for the colour device.

Table 5.3.10-1: DataPoints of colour ModuleClass

Name	Type	Readable	Writable	Optional	Documentation
red	xs:integer	true	true	false	The R value of RGB; the range is [0, 255]
green	xs:integer	true	true	false	The G value of RGB; the range is [0, 255]
blue	xs:integer	true	true	false	The B value of RGB; the range is [0, 255]

5.3.11 colourSaturation

This ModuleClass describes a colour saturation value. The value is an integer. A colourSaturation has a range of [0, 100]. A colourSaturation value of 0 means producing black and white images. A colourSaturation value of 50 means producing device specific normal colour images. A colourSaturation value of 100 means producing device very colourful images.

Table 5.3.11-1: DataPoints of colourSaturation ModuleClass

Name	Type	Readable	Writable	Optional	Documentation
colourSaturation	xs:integer	true	true	false	The status of colour saturation level.

5.3.12 doorStatus

This ModuleClass provides the status of a door. It is intended to be part of a larger object such as a refrigerator and an oven that might have multiple doors.

Table 5.3.12-1: DataPoints of doorStatus ModuleClass

Name	Type	Readable	Writable	Optional	Documentation
doorState	hd:doorState	true	false	false	"Closed" indicates that door is closed, "Open" indicates that the door is open, "Opening" indicates that the door is opening, "Closing" indicates that the door is closing, "Stopped" indicates that the door is in stationary state.
openDuration	m2m:timestamp	true	false	true	The time duration the door has been open.
openAlarm	xs:boolean	true	true	true	The state of the door open alarm. "True" indicates that the open alarm is active. "False" indicates that the open alarm is not active.

5.3.13 electricVehicleConnector

This ModuleClass provides the information about charging/discharging devices for electric vehicles.

Table 5.3.13-1: DataPoints of electricVehicleConnector ModuleClass

Name	Type	Readable	Writable	Optional	Documentation
status	xs:boolean	true	false	false	The status of connection.

Table 5.3.13-2: Properties of electricVehicleConnector ModuleClass

Name	Type	Value	Optional	Documentation
propChargingCapacity	xs:integer		true	Rated charging capacity.
propDischargingCapacity	xs:integer		true	Rated discharging capacity.

5.3.14 energyConsumption

This ModuleClass describes the energy consumed by the device since power up. One particular use case for energyConsumption ModuleClass is smart meter.

Table 5.3.14-1: DataPoints of energyConsumption ModuleClass

Name	Type	Readable	Writable	Optional	Documentation
power	xs:float	true	false	false	The power of the device. The common unit is Watt (W).
absoluteEnergyConsumption	xs:float	true	false	true	The absolute energy consumption, reflecting the real measurement of accumulative energy. The common unit is Watt-hour (Wh).
roundingEnergyConsumption	xs: integer	true	false	true	This energy consumption data can be calculated by using significantDigits and multiplyingFactors.
significantDigits	xs:integer	true	false	true	The number of effective digits for data.
multiplyingFactors	xs:floatr	true	false	true	The unit for data (multiplying factors), e.g. 1 kWh, 0,1 kWh, 0,01 kWh, etc.
voltage	xs:float	true	false	true	The voltage of the device. The common unit is volts (V).
current	xs:float	true	false	true	The current of the device. The common unit is ampere (A).
frequency	xs:float	true	false	true	The frequency of the device. The common unit is hertz (Hz).

5.3.15 energyGeneration

This ModuleClass provides information about generation data on electric generator devices such as a photo voltaic power system, fuel cells, or microgeneration.

Table 5.3.15-1: DataPoints of energyGenerationModuleClass

Name	Type	Readable	Writable	Optional	Documentation
powerGenerationData	xs:float	true	false	true	Amount of instantaneousgeneration data.
roundingEnergyGeneration	xs:integer	true	false	true	This energy generation data can be calculated by using significantFigures and multiplyingFactors.
significantDigits	xs:integer	true	false	true	The number of effective digits for data.
multiplyingFactors	xs:floatr	true	false	true	The unit for data (multiplying factors), e.g. 1 kWh, 0,1 kWh, 0,01 kWh, etc.

5.3.16 faultDetection

This ModuleClass provides information about whether a fault has occurred in the actual device.

Table 5.3.16-1: DataPoints of faultDetection ModuleClass

Name	Type	Readable	Writable	Optional	Documentation
status	xs:boolean	true	false	false	Status of fault detection.
code	xs:integer	true	false	true	Code of the fault.
description	xs:string	true	false	true	Message of the fault.

5.3.17 height

This ModuleClass provides the capability to report the measurement of height.

Table 5.3.17-1: DataPoints of height ModuleClass

Name	Type	Readable	Writable	Optional	Documentation
height	xs:float	true	false	false	The height measurement. The common unit is centimetre (cm).

5.3.18 hotWaterSupply

This ModuleClass provides information about the status of supplying hot water into tanks or bath tubs.

Table 5.3.18-1: DataPoints of hotWaterSupply ModuleClass

Name	Type	Readable	Writable	Optional	Documentation
status	xs:boolean	true	false	false	The status of watering operation.
bath	xs:boolean	true	true	true	The status of filling bath tub.

5.3.19 keypad

This ModuleClass provides the capability to require a user defined service through the key-in number. For example, a user can define key 1 as "require a takeout from restaurant XXX with combo meal 1". The IoT service provider or user can define the services.

Table 5.3.19-1: DataPoints of keypad ModuleClass

Name	Type	Readable	Writable	Optional	Documentation
keyNumber	xs:integer	true	false	false	The number of key.

5.3.20 motionSensor

This ModuleClass provides the capabilities to indicate the occurrence of motion and raises an alarm if the triggering criterion is met.

Table 5.3.20-1: DataPoints of motionSensor ModuleClass

Name	Type	Readable	Writable	Optional	Documentation
alarm	xs:boolean	true	false	false	The detection of the motion occurrence.
silentTime	xs:integer	true	true	true	The time that the motionSensor restrains from sending an alarm in case continuous motions are detected after one alarm is produced. This DataPoint can be used to avoid repeated alarm reports.
sensitivity	xs:integer	true	true	true	The level of the detection accuracy of the motion sensor. This DataPoint can be used to control the number of the report.

5.3.21 oximeter

This ModuleClass provides the capability to report the measurement of blood characteristics.

Table 5.3.21-1: DataPoints of oximeter ModuleClass

Name	Type	Readable	Writable	Optional	Documentation
diastolicPressure	xs:integer	true	false	false	The measurement of diastolic pressure by Oximeter. The common unit is millimetre of mercury (mmHg).
systolicPressure	xs:integer	true	false	false	The measurement of systolic pressure by Oximeter. The common unit is millimetre of mercury (mmHg).
pulseRate	xs:integer	true	false	false	The measurement of pulse rate by Oximeter. The common unit is in beats per minute.
oxygenSaturation	xs:integer	true	false	false	The measurement of oxygen saturation by Oximeter. The common unit is in percentage.

5.3.22 powerSave

This ModuleClass provides capabilities to enable power saving mode and monitor the current status.

Table 5.3.22-1: DataPoints of powerSave ModuleClass

Name	Type	Readable	Writable	Optional	Documentation
powerSaveEnabled	xs:boolean	true	true	false	The current status of the Power Saving Mode. "True" indicates enabled, and "false" indicates not enabled.

5.3.23 pushButton

This ModuleClass provides the capability to indicate the operation of a button style switch. A typical application can be an SOS button.

Table 5.3.23-1: DataPoints of pushButton ModuleClass

Name	Type	Readable	Writable	Optional	Documentation
pushed	xs:boolean	true	false	false	To indicate the press of the button.

5.3.24 recorder

This ModuleClass provides the capability to record the video/audio for a defined duration.

Table 5.3.24-1: DataPoints of recorder ModuleClass

Name	Type	Readable	Writable	Optional	Documentation
duration	xs:integer	true	true	false	The duration for video/audio recording. Set to trigger the recorder. The common unit is seconds.

5.3.25 refrigeration

This ModuleClass describes a refrigeration function. This is not a Refrigerator device. The filter state is a read-only value providing the percentage life time remaining for the water filter. RapidFreeze is a boolean that controls the rapid freeze capability if present. RapidCool is a boolean that controls the rapid cool capability if present. Defrost is a boolean that controls the defrost cycle if present.

Table 5.3.25-1: DataPoints of refrigeration ModuleClass

Name	Type	Readable	Writable	Optional	Documentation
filterLifetime	xs:integer	true	false	true	Percentage life time remaining for the water filter.
rapidFreeze	xs:boolean	true	true	true	Indicates whether the unit has a rapid freeze capability active.
rapidCool	xs:boolean	true	true	true	Indicates whether the unit has a rapid cool capability active.
defrost	xs:boolean	true	true	true	Indicates whether a defrost cycle is currently active.

5.3.26 relativeHumidity

This ModuleClass provides the capability for the device to report the humidity based on a specified rule that is vendor discretionary.

Table 5.3.26-1: DataPoints of relativeHumidity ModuleClass

Name	Type	Readable	Writable	Optional	Documentation
relativeHumidity	xs:float	true	false	false	The measurement of the relative humidity value; the common unit is percentage.
desiredHumidity	xs:float	true	true	true	Desired value for Humidity.

5.3.27 rinseLevel

This ModuleClass provides capabilities to control and monitor the level of rinse. It is intended to be part of object which uses rinse such as a washing machine.

Table 5.3.27-1: DataPoints of rinseLevel ModuleClass

Name	Type	Readable	Writable	Optional	Documentation
rinseLevel	hd:liquid Level	true	true	false	The level of rinse (see clause 5.5.3). A higher value indicates a higher rinse level.

5.3.28 runMode

This ModuleClasses provides capabilities to control and monitor the operational modes of appliances.

Table 5.3.28-1: DataPoints of runMode ModuleClass

Name	Type	Readable	Writable	Optional	Documentation
operationMode	hd:supportedModes	true	true	false	Currently active mode.
supportedModes	list of hd:supportedModes	true	true	false	List of possible modes the device supports (see clause 5.5.7)

5.3.29 signalStrength

This ModuleClass provides the capability to monitor the strength of the signal.

Table 5.3.29-1: DataPoints of signalStrength ModuleClass

Name	Type	Readable	Writable	Optional	Documentation
lqi	xs:integer	true	false	false	The current value of link quality indicator, which reflects the scaling of rssi by dividing the received signal strength over reference signal strength. The common unit for lqi is percentage [0, 100]. For the detailed definition, please see IEEE 802.15.4 [i.4], clause 6.7.8.
rssi	xs:float	true	false	true	The current value of received signal strength indicator, which reflects the raw signal level.

5.3.30 smokeSensor

This ModuleClass provides the capabilities to indicate the detection of smoke and raises an alarm if the triggering criterion is met.

Table 5.3.30-1: DataPoints of smokeSensor ModuleClass

Name	Type	Readable	Writable	Optional	Documentation
alarm	xs:boolean	true	false	false	The detection of smoke.
detectedTime	m2m:timestamp	true	true	true	The time the smoke is detected.

5.3.31 spinLevel

This ModuleClass provides capabilities to control and monitor the level of spin. It is intended to be part of objects which use spinning function such as a washing machine and a dryer.

Table 5.3.31-1: DataPoints of spinLevel ModuleClass

Name	Type	Readable	Writable	Optional	Documentation
spinLevelStrength	hd:spinLevel Strength	true	true	false	The value of spin-dry level (see clause 5.5.4). A higher value indicates a higher spin level.

5.3.32 televisionChannel

This ModuleClass provides capabilities to set and get channels of a device that has a channel list.

Table 5.3.32-1: Actions of televisionChannel ModuleClass

Return Type	Name	Argument	Optional	Documentation
none	upChannel	none	true	Change the current channel to the next channel in the stored list of available channels. If the current channel is the last one in the list, the new set channel may be the first one in the list.
none	downChannel	none	true	Change the current channel to the previous channel in the stored list of available channels. If the current channel is the first one in the list, the new set channel may be the last one in the list.

Table 5.3.32-2: DataPoints of televisionChannel ModuleClass

Name	Type	Readable	Writable	Optional	Documentation
channelId	xs:integer	true	true	false	Current channel ID.
availableChannels	list of xs:integer	true	false	true	The list of available channel numbers which may be built by automatic scan and/or manual selection.
previousChannel	xs:integer	true	false	true	The channel number which was selected previously.

5.3.33 temperature

This ModuleClass provides capabilities to represent the current temperature and target temperature of devices such as an air conditioner, refrigerator, oven, etc.

Table 5.3.33-1: DataPoints of temperature ModuleClass

Name	Type	Readable	Writable	Optional	Documentation
currentTemperature	xs:float	true	false	false	The current temperature.
targetTemperature	xs:float	true	true	true	The desired temperature to reach.
unit	xs:string	true	false	true	The unit for the temperature values. The default is Celsius (C).
minValue	xs:float	true	false	true	Minimum value of targetTemperature.
maxValue	xs:float	true	false	true	Maximum value of targetTemperature.
stepValue	xs:float	true	false	true	Step value allowed for targetTemperature.

5.3.34 temperatureAlarm

This ModuleClass provides the capabilities to indicate the detection of abnormal temperatures and raises an alarm if the triggering criterion is met.

Table 5.3.34-1: DataPoints of temperatureAlarm ModuleClass

Name	Type	Readable	Writable	Optional	Documentation
alarm	xs:boolean	true	false	false	The detection of abnormal temperature.
temperature	xs:float	true	false	true	To report the value of the temperature. The common unit is Celsius (C).
temperatureThreshhold	xs:integer	true	true	true	The threshold to trigger the alarm.

5.3.35 timer

This ModuleClass provides capabilities to monitor and control the times when the appliance executes its operations (i.e. when it starts, when it ends, etc.).

Table 5.3.35-1: Actions of timer ModuleClass

Return Type	Name	Argument	Optional	Documentation
none	activateClockTimer	none	true	Activate current clock timer.
none	deactivateClockTimer	none	true	Deactivate current clock timer.

Table 5.3.35-2: DataPoints of timer ModuleClass

Name	Type	Readable	Writable	Optional	Documentation
referenceTimer	xs:integer	true	false	true	A Timer (e.g. a time-based value, App Defined Epoch, Progressive) expressed in seconds. The value indicates a time counter to be used as reference for the other time-based data points of this ModuleClass. Usually it is the time since the last event of power on of the producer (or more in detail the boot of its connectivity node).
targetTimeToStart	xs:integer	true	true	true	A TimeSpan (e.g. a time-based value, App Defined Epoch, Fixed) expressed in seconds. The value indicates the time when the appliance is expected to start its operation, starting counting from the last ReferenceTimer.
targetTimeToStop	xs:integer	true	true	true	A TimeSpan (e.g. a time-based value, App Defined Epoch, Fixed) expressed in seconds. The value indicates the time when the appliance is expected to stop its operation, starting counting from the last ReferenceTimer.
estimatedTimeToEnd	xs:integer	true	false	true	A Timer (e.g. a time-based value, App Defined Epoch, Progressive) expressed in seconds. The value indicates the time to the ends of appliance operations. It is calculated at runtime by device itself during the execution of its operation.
runningTime	xs:integer	true	false	true	It is a Timer (e.g. a time-based value, App Defined Epoch, Progressive) expressed in seconds. It indicates the time of the current operation. Usually its value is increasing of one value each second. It starts counting from 0 when the operation starts and stops counting when the operation ends.
targetDuration	xs:integer	true	false	true	A TimeSpan (e.g. a time-based value, App Defined Epoch, Fixed) expressed in seconds. The value indicates a time, representing the target duration of the operation as per user selection.
absoluteStartTime	m2m:time stamp	true	true	true	An absolute time to specify the start time.
absoluteStopTime	m2m:time stamp	true	true	true	An absolute time to specify the stop time.

5.3.36 turbo

This ModuleClass provides capabilities to enable turbo mode and monitor the current status of the turbo function. It is intended to be part of objects which use turbo function such as an air conditioner, a washing machine, etc.

Table 5.3.36-1: DataPoints of turbo ModuleClass

Name	Type	Readable	Writable	Optional	Documentation
turboEnabled	xs:boolean	true	true	false	The current status of the Turbo Mode. "True" indicates enabled, and "False" indicates not enabled.

5.3.37 waterFlow

This ModuleClass is for controlling water strength of a device.

Table 5.3.37-1: DataPoints of waterFlow ModuleClass

Name	Type	Readable	Writable	Optional	Documentation
waterLevelStrength	hd:waterFlowStrength	true	true	false	The desired level of water flow (see clause 5.5.8). A higher value indicates higher water flow.

5.3.38 waterLevel

This ModuleClass provides the level and supply source of water for an appliance.

Table 5.3.38-1: DataPoints of waterLevel ModuleClass

Name	Type	Readable	Writable	Optional	Documentation
liquidLevel	hd:liquidLevel	true	true	false	The desired level of water (see clause 5.5.3).

5.3.39 waterSensor

This ModuleClass provides the capabilities to indicate whether or not water has been sensed and raises an alarm if the triggering criterion is met.

Table 5.3.39-1: DataPoints of waterSensor ModuleClass

Name	Type	Readable	Writable	Optional	Documentation
alarm	xs:boolean	true	false	false	The detection of water.

5.3.40 weight

This ModuleClass provides the capability to report the measurement of weight.

Table 5.3.40-1: DataPoints of weight ModuleClass

Name	Type	Readable	Writable	Optional	Documentation
weight	xs:float	true	false	false	The weight measurement. The common unit is kilogram (kg).

5.3.41 wind

This ModuleClass is for controlling wind strength and direction of a device.

Table 5.3.41-1: DataPoints of wind ModuleClass

Name	Type	Readable	Writable	Optional	Documentation
windStrength	hd:windStrength	true	true	false	The current strength of the wind (see clause 5.5.9).
directionUp	xs:boolean	true	true	true	The current status of the upward blowing enablement. "True" indicates enabled, and "False" indicates not enabled.
directionDown	xs:boolean	true	true	true	The current status of the downward blowing enablement. "True" indicates enabled, and "False" indicates not enabled.

Name	Type	Readable	Writable	Optional	Documentation
directionRight	xs:boolean	true	true	true	Right side enablement (0:off, 1:on) The current status of the rightward blowing enablement. "True" indicates enabled, and "False" indicates not enabled.
directionLeft	xs:boolean	true	true	true	The current status of the leftward blowing enablement. "True" indicates enabled, and "False" indicates not enabled.
directionAuto	xs:boolean	true	true	true	The current status of the automatic blowing enablement. "True" indicates enabled, and "False" indicates not enabled.

5.4 Device models

5.4.1 deviceAirConditioner

An air conditioner is a home appliance used to alter the properties of air (primarily temperature and humidity) to more comfortable conditions. This air conditioner information model provides capabilities to control and monitor air conditioner specific functions and resources.

Table 5.4.1-1: Modules of deviceAirConditioner Device model

Module Instance Name	Module Class Name	Optional	Description
binarySwitch	binarySwitch	true	See clause 5.3.5
runMode	runMode	true	See clause 5.3.28
temperature	temperature	true	See clause 5.3.33
timer	timer	true	See clause 5.3.35
turbo	turbo	true	See clause 5.3.36
wind	wind	true	See clause 5.3.41

5.4.2 deviceClothesWasher

A clothes washer is a home appliance that is used to wash laundry, such as clothing and sheets. This information model provides capabilities to interact with specific functions and resources of clothes washers.

Table 5.4.2-1: Modules of deviceClothesWasher Device model

Module Instance Name	Module Class Name	Optional	Description
binarySwitch	binarySwitch	true	See clause 5.3.5
timer	timer	true	See clause 5.3.35
runMode	runMode	true	See clause 5.3.28
temperature	temperature	true	See clause 5.3.33
waterLevel	waterLevel	true	See clause 5.3.38
rinseLevel	rinseLevel	true	See clause 5.3.27
waterFlow	waterFlow	true	See clause 5.3.37
spinLevel	spinLevel	true	See clause 5.3.31

5.4.3 deviceElectricVehicleCharger

An electric vehicle charger is a device that is used for charging or discharging electric vehicles.

Table 5.4.3-1: Modules of deviceElectricVehicleCharger Device model

Module Instance Name	Module Class Name	Optional	Description
faultDetection	faultDetection	false	See clause 5.3.16
binarySwitch	binarySwitch	false	See clause 5.3.5
runMode	runMode	false	See clause 5.3.28
battery	battery	false	See clause 5.3.10
electricVehicleConnector	electricVehicleConnector	false	See clause 5.3.13

5.4.4 deviceLight

A light is a device that is used to control the state of an illumination device.

Table 5.4.4-1: Modules of deviceLight Device model

Module Instance Name	Module Class Name	Optional	Description
faultDetection	faultDetection	true	See clause 5.3.16
binarySwitch	binarySwitch	false	See clause 5.3.5
runMode	runMode	true	See clause 5.3.28
colour	colour	true	See clause 5.3.10
colourSaturation	colourSaturation	true	See clause 5.3.11
brightness	brightness	true	See clause 5.3.8

5.4.5 deviceMicrogeneration

A microgeneration is a Home Energy Management System (HEMS) device that is used to create energy. Examples of microgeneration devices are photovoltaics device or fuel cells.

Table 5.4.5-1: Modules of deviceMicrogeneration Device model

Module Instance Name	Module Class Name	Optional	Description
faultDetection	faultDetection	true	See clause 5.3.16
binarySwitch	binarySwitch	true	See clause 5.3.5
runMode	runMode	true	See clause 5.3.28
energyGeneration	energyGeneration	false	See clause 5.3.15

Table 5.4.5-2: Device specific properties of deviceMicrogeneration Device model

Name	Type	Value	Optional	Documentation
propGenerationSource	xs:string	-	false	The type of generating source.

5.4.6 deviceOven

An oven is a home appliance used to roast and heat food in a complete stove. This information model is applicable to different types of ovens: gas ovens, electrical ovens, steam ovens, microwave ovens, etc. This information model provides capabilities to interact with specific functions and resources of ovens.

Table 5.4.6-1: Modules of deviceOven Device model

Module Instance Name	Module Class Name	Optional	Description
binarySwitch	binarySwitch	true	See clause 5.3.5
runMode	runMode	true	See clause 5.3.28
timer	timer	true	See clause 5.3.35
temperature	temperature	true	See clause 5.3.33

5.4.7 deviceRefrigerator

A refrigerator is a home appliance used to store food at temperatures which are a few degrees above the freezing point of water. This information model provides capabilities to interact with specific functions and resource of refrigerators.

Table 5.4.7-1: Modules of deviceRefrigerator Device model

Module Instance Name	Module Class Name	Optional	Description
binarySwitch	binarySwitch	true	See clause 5.3.5
powerSave	powerSave	true	See clause 5.3.22
doorstatusdoorStatus	doorStatus	true	See clause 5.3.12
frozenTemperature	temperature	true	See clause 5.3.33
freshTemperature	temperature	true	See clause 5.3.33
customTemperature	temperature	true	See clause 5.3.33
refrigeration	refrigeration	true	See clause 5.3.25

5.4.8 deviceRobotCleaner

A robot cleaner is an autonomous robotic vacuum cleaner that has intelligent programming and a limited vacuum cleaning system. This robot cleaner information model provides capabilities to control and monitor robot cleaner specific functions and resources.

Table 5.4.8-1: Modules of deviceRobotCleaner Device model

Module Instance Name	Module Class Name	Optional	Description
binarySwitch	binarySwitch	true	See clause 5.3.5
runMode	runMode	true	See clause 5.3.28
battery	battery	true	See clause 5.3.10
timer	timer	true	See clause 5.3.35

5.4.9 deviceSmartElectricMeter

A smart electric meter is a metering device that is used to measure consumption data for electricity.

Table 5.4.9-1: Modules of deviceSmartElectricMeter Device model

Module Instance Name	Module Class Name	Optional	Description
faultDetection	faultDetection	true	See clause 5.3.16
binarySwitch	binarySwitch	true	See clause 5.3.5
runMode	runMode	true	See clause 5.3.28
clock	clock	true	See clause 5.3.9
energyConsumption	energyConsumption	false	See clause 5.3.14
energyGeneration	energyGeneration	true	See clause 5.3.15

Table 5.4.9-2: Device specific properties of deviceSmartElectricMeter Device model

Name	Type	Value	Optional	Documentation
propMeasuringScope	xs:string	-	true	Measuring scope of the meter (e.g. whole house, room, or device).

5.4.10 deviceStorageBattery

A storage battery is a HEMS device that is used to provide the home with electrical energy.

Table 5.4.10-1: Modules of deviceStorageBattery Device model

Module Instance Name	Module Class Name	Optional	Description
faultDetection	faultDetection	true	See clause 5.3.16
binarySwitch	binarySwitch	true	See clause 5.3.5
runMode	runMode	true	See clause 5.3.28
battery	battery	false	See clause 5.3.10

5.4.11 deviceTelevision

A television (TV) is a home appliance used to show audio and visual content such as broadcasting programs and network streaming. This TV information model provides capabilities to control and monitor TV specific resources.

Table 5.4.11-1: Modules of deviceTelevision Device model

Module Instance Name	Module Class Name	Optional	Description
binarySwitch	binarySwitch	true	See clause 5.3.5
audioVolume	audioVolume	true	See clause 5.3.6
televisionChannel	televisionChannel	true	See clause 5.3.32
audioVideoInput	audioVideoInput	true	See clause 5.3.5

5.4.12 deviceThermostat

A thermostat is used to control the ambient temperature of rooms within for example a house. This information model provides capabilities to interact with specific functions of thermostats.

Table 5.4.12-1: Modules of deviceThermostat Device model

Module Instance Name	Module Class Name	Optional	Description
runMode	runMode	true	See clause 5.3.28. The possible values of the supportedModes datapoint for the Thermostat device are included in clause 5.5.7
timer	timer	true	See clause 5.3.35
temperature	temperature	false	See clause 5.3.33

5.4.13 deviceWaterHeater

A water heater is a device that is used to provide hot water through home facilities.

Table 5.4.13-1: Modules of deviceWaterHeater Device model

Module Instance Name	Module Class Name	Optional	Description
faultDetection	faultDetection	true	See clause 5.3.16
binarySwitch	binarySwitch	false	See clause 5.3.5
runMode	runMode	true	See clause 5.3.28
clock	clock	true	See clause 5.3.9
boiler	boiler	true	See clause 5.3.7
hotWaterSupply	hotWaterSupply	true	See clause 5.3.18

5.5 Enumeration type definitions

5.5.1 hd:deviceType

Used for Device Type property of device models.

Table 5.5.1-1: Interpretation of deviceType

Value	Interpretation	Note
1	deviceAirConditioner	See clause 5.4.1
2	deviceClothesWasher	See clause 5.4.2
3	deviceElectricVehicleCharger	See clause 5.4.3
4	deviceLight	See clause 5.4.4
5	deviceMicrogeneration	See clause 5.4.5
6	deviceOven	See clause 5.4.6
7	deviceRefrigerator	See clause 5.4.7
8	deviceRobotCleaner	See clause 5.4.8
9	deviceSmartElectricMeter	See clause 5.4.9
10	deviceStorageBattery	See clause 5.4.10
11	deviceTelevision	See clause 5.4.11
12	deviceThermostat	See clause 5.4.12
13	deviceWaterHeater	See clause 5.4.13
0	undefinedVendorExt	For vendor specific usage
NOTE: See clause 5.4 "Device models".		

5.5.2 hd:supportedInputSources

Used for supportedInputSources DataPoint of AudioVideoInput ModuleClass.

Table 5.5.2-1: Interpretation of supportedInputSources

Value	Interpretation	Note
1	tuner	
2	component	
3	composite	
4	svideo	
5	rgb	
6	dvi	
7	hdmi	
8	displayPort	
9	scart	
10	externalStorage	
11	network	
NOTE: See clause 5.3.2 "audioVideoInput".		

5.5.3 hd:liquidLevel

Used for DataPoint indicating level of liquid such as rinseLevel.

Table 5.5.3-1: Interpretation of liquidLevel

Value	Interpretation	Note
1	zero	
2	low	
3	medium	
4	high	
5	maximum	
NOTE: See clause 5.3.27 "rinseLevel", clause 5.3.38 "waterLevel".		

5.5.4 hd:spinLevelStrength

Used for DataPoints indicating strength of a spinLevel.

Table 5.5.4-1: Interpretation of strength

Value	Interpretation	Note
1	zero	
2	sensitive	
3	weak	
4	medium	
5	strong	
6	maximum	
NOTE: See clause 5.3.31 "spinLevel".		

5.5.5 hd:doorState

Used for doorState DataPoint of doorStatus ModuleClass.

Table 5.5.5-1: Interpretation of doorState

Value	Interpretation	Note
1	Closed	
2	Open	
3	Opening	
4	Closing	
5	Stopped	
NOTE: See clause 5.3.12 "doorStatus".		

5.5.6 hd:tone

Used for tone DataPoint of alarmSpeaker ModuleClass.

Table 5.5.6-1: Interpretation of tone

Value	Interpretation	Note
1	Fire	
2	Theft	
3	Emergency	
4	Doorbell	
5	DeviceFail	
NOTE: See clause 5.3.1 "alarmSpeaker".		

5.5.7 hd:supportedModes

Used for supportedModes DataPoint of runMode ModuleClass.

Table 5.5.7-1: Interpretation of supportedModes

Value	Interpretation	Note
1	antifreeze	This mode sets the thermostat to a minimum temperature to avoid home system to freeze when the habitants are not there for a long time
2	manual	This mode allows for direct change of the temperature indication for the thermostat by the user
3	eco	This is to set the thermostat to the economic mode
4	program	The program mode is used to set the thermostat to a predefined mode
5	off	
6	ready	
7	running	
8	paused	
9	aborted	
10	cancelled	
11	completed	
12	washing	
13	spinning	
14	drying	
15	rinsing	
16	warming up	
17	cooking	
18	cooling	
19	dehumidifying	
20	energy saving	
21	charging	
22	homing	
23	docking	
NOTE: See clause 5.3.28 "runMode".		

5.5.8 hd:alertColourCode

Used for light DataPoint of alarmSpeaker ModuleClass.

Table 5.5.8-1: Interpretation of alertColourCode

Value	Interpretation	Note
1	Red	This colour indicates the alarm status.
2	Green	This colour indicates the alarm has been cleared.
NOTE: See clause 5.3.1 "alarmSpeaker".		

5.5.9 hd:waterFlowStrength

Used for DataPoints indicating strength of a waterflow.

Table 5.5.9-1: Interpretation of waterFlowStrength

Value	Interpretation	Note
1	zero	
2	sensitive	
3	weak	
4	medium	
5	strong	
6	maximum	
NOTE: See clause 5.3.37 "waterFlow".		

5.5.10 hd:windStrength

Used for DataPoints indicating strength of wind.

Table 5.5.10-1: Interpretation of windStrength

Value	Interpretation	Note
1	zero	
2	sensitive	
3	weak	
4	medium	
5	strong	
6	maximum	
NOTE: See clause 5.3.41 "wind".		

5.6 Universal and Common Properties for Device models

Properties specified in table 5.6-1 are applicable to all device models. Some properties are mandatory for all device models and called "Universal Properties", since they are universally seen in typical device types and carry necessary information to identify each device instance. Others are optional for all device models and called "Common Properties", since they are commonly used in many device types but not always.

Universal and Common Properties are not repeated in the property table of each device model in clause 5.4, where only device specific properties shall be specified. Universal and Common Properties shall be instantiated in each device model following the optionality specified in table 5.6-1.

NOTE: The instantiated values of the universal properties might be empty in case of exceptional scenarios, e.g. interworking with non-oneM2M device models.

Table 5.6-1: Universal and Common Properties

Name	Type	Value	Optional	Documentation
propDeviceSerialNum	xs:string	-	false	Device serial number (assigned by manufacturer)
propDeviceType	hd:deviceType	-	false	Device type (see clause 5.5.1)
propDeviceModelName	xs:string	-	false	Device model name
propDeviceManufacturer	xs:string	-	false	Manufacturer name of the device
propDeviceName	xs:string	-	true	Device name
propDeviceSubModelName	xs:string	-	true	Device sub-model name
propDeviceAliasName	xs:string	-	true	Device alias name
propDeviceFirmwareVersion	xs:string	-	true	Device firmware version
propHardwareVersion	xs:string	-	true	Device hardware version
propOsVersion	xs:string	-	true	Version of the operation system (defined by manufacturer)
propProtocol	xs:string	-	true	A comma separated list of MIME types for all supported communication protocol(s) of the device. Example: "application/x-alljoin;version=1.0,application/x-echonet-lite;version=1.0" indicates the device supports both AllJoyn v1.0 and Echonet Lite v1.0
propCountry	xs:string	-	true	Country code of the device
propLocation	xs:string	-	true	Location where the device is installed. It may be configured via the user interface provided by the 'presentationURL' property or any other means
propSystemTime	xs:datetime	-	true	Reference time for the device
propManufacturerDetailsLink	xs:url	-	true	URL to manufacturer's website
propDateOfManufacture	xs:datetime	-	true	Manufacturing date of device
propSupportURL	xs:url	-	true	URL that points to product support information of the device

Name	Type	Value	Optional	Documentation
propPresentationURL	xs:url	-	true	To quote UPnP: "the control point can retrieve a page from this URL, load the page into a web browser, and depending on the capabilities of the page, allow a user to control the device and/or view device status. The degree to which each of these can be accomplished depends on the specific capabilities of the presentation page and device"

6 The Principle of Resource Mapping for Home Appliance Information Model

6.1 Introduction

Home appliance information models which are defined in clause 5 need to be represented as resources in the oneM2M system. This clause defines the principle of resource mapping based on <flexContainer>. The individual information mapping is provided in annex A, clauses A.1 to A.4.

6.2 The Resource Mapping Rules

6.2.1 Introduction

The present clause specifies the rule to map the "Home Appliance Information Model" in clause 5, to oneM2M resources.

6.2.2 Resource mapping for Device model

When the AE exposes a controlling interface for a home domain device which is specified as an information model in clause 5.4, a specialization of the <flexContainer> resource shall be created as the mapping of the model following conversion rules:

Rule 1-1: Each Device model defined in clause 5.4 shall be mapped to a specialization of <flexContainer> resource with associated 'DeviceClass ID' (e.g. "org.onem2m.home.device.tv") on *containerDefinition* attributes.

Rule 1-2: Each entry of 'Module' table shall be mapped to child resource(s) which is mapped as a specialized <flexContainer> following the rule in clause 6.2.3 'Resource mapping for ModuleClass'.

Rule 1-3: Each entry of 'Property' table shall be mapped to a child resource which is mapped as a specialization of the <flexContainer> resource following the rule in clause 6.2.5.

Rule 1-4: XSD file for each Device model shall be named following naming convention:
'HD-<name of Device model>-v<version of TS>.xsd'
For example, XSD file for 'deviceAirConditioner' is named as 'HD-deviceAirConditioner-v1_0_0.xsd'.

6.2.3 Resource mapping for ModuleClass

The ModuleClass models (in clause 5.3) shall be mapped to the specializations of <flexContainer> resource. The following rules shall be applied.

When the Device model in clause 5.4 is mapped to the <flexContainer> resource, and if the device supports the functionality associated with a ModuleClass in the model, a <flexContainer> resource which is mapped from ModuleClass definitions shall be created as a child resource:

Rule 2-1: The ModuleClass ID shall be specified on the containerDefinition attribute (e.g. "org.onem2m.home.moduleclass. audiovolume").

Rule 2-2: Each entry of 'Action', 'Property', and 'DataPoint' in ModuleClass definitions shall be mapped following the resource mapping rules for them.

Rule 2-3: XSD file for each ModuleClass shall be named following naming convention:
'HD-mod-<name of ModuleClass>-v<version of TS>.xsd'
For example, XSD file for 'binarySwith' is named as 'HD-mod-binarySwitch-v1_0_0.xsd. The Device model which refer any ModuleClass shall include the XSD of the ModuleClasses.

6.2.4 Resource mapping for Action

When the Device model in clause 5.4 or the ModuleClass model in clause 5.3 is mapped to the <flexContainer> resource, and if the device supports the functionality associated with the Action in the model, a <flexContainer> resource which is mapped from the Action definition shall be created as a child resource:

Rule 3-1: The Action ID shall be specified on the containerDefinition attribute (e.g. "org.onem2m.home.moduleclass.audiovolume.upvolume").

Rule 3-2: When the Action supports any 'Arguments' or 'Return Type', they are mapped to [customizedAttribute] with its variable names.

Rule 3-3: XSD file for each Action shall be named following naming convention:
'HD-act-<name of Action>-v<version of TS>.xsd'.
For example, XSD file for 'toggle' is named as 'HD-act-toggle-v1_0_0.xsd'. The device or ModuleClass which refers any Action shall include the XSD of the Action.

Rule 3-4: When the Action does not support any 'Argument' or 'Return Type', the Action shall be triggered by updating with null Content parameter.

6.2.5 Resource mapping for Property

When the Device model (in clause 5.4) or the ModuleClass model (in clause 5.3) is mapped to the <flexContainer> resource, and if the device supports a Property, the following rules shall be applied:

Rule 4-1: Each entry of Property table in Device model or ModuleClass model, shall be mapped to the [customAttribute] of <flexContainer> resource which is mapped from associated Device model or ModuleClass model, with its Property name with prefix 'prop'.

6.2.6 Resource mapping for DataPoint

When the ModuleClass model (in clause 5.3) is mapped to the <flexContainer> resource, and if the ModuleClass supports a DataPoint, the following rules shall be applied:

Rule 5-1: Each entry of DataPoint table in ModuleClass model, shall be mapped to [customAttribute] of <flexContainer> resource which is mapped from associated ModuleClass model, with its DataPoint name.

6.3 Short names

6.3.1 Introduction

XML and JSON representations require the explicit encoding of the names of resource attributes, (in the case of XML) and resource types. Whenever a protocol binding transfers such a name over a oneM2M reference point, it shall use a shortened form of that name. Short names enable payload reduction on involved telecommunication interfaces.

The mapping between the full names and their shortened form is given in the clauses that follow.

6.3.2 Resource types

In protocol bindings resource type names for device models shall be translated into short names of table 6.3.2-1.

Table 6.3.2-1: Specialization type short names (Device models)

Resource Type Name	Short Name
deviceAirConditioner	***deACr***
deviceClothesWasher	***deCWr***
deviceElectricVehicleCharger	***dEVCr***
deviceLight	***devLt***
deviceMicrogeneration	***devMn***
deviceOven	***devOn***
deviceRefrigerator	***devRr***
deviceRobotCleaner	***deRCr***
deviceSmartElectricMeter	***dSEMr***
deviceStorageBattery	***deSBy***
deviceTelevision	***devTn***
deviceThermostat	***devTt***
deviceWaterHeater	***deWHr***

In protocol bindings resource type names for module classes shall be translated into short names of table 6.3.2-2.

Table 6.3.2-2: Specialization type short names (ModuleClasses and Module Instances)

Resource Type Name	Short Name
alarmSpeaker	***alaSr***
audioVideoInput	***auVIt***
audioVolume	***audVe***
battery	***batty***
binarySwitch	***binSh***
bioElectricalImpedanceAnalysis	***bEIAs***
boiler	***boilr***
brightness	***brigs***
clock	***clock***
colour	***color***
colourSaturation	***colSn***
customTemperature	***cusTe***
doorStatus	***dooSe***
electricVehicleConnector	***eIVCr***
energyConsumption	***eneCn***
energyGeneration	***eneGn***
faultDetection	***fauDn***
freshTemperature	***freTe***
frozenTemperature	***froTe***
height	***heigt***
hotWaterSupply	***hoWSy***
keypad	***keypd***
motionSensor	***motSr***
oximeter	***oximr***
powerSave	***powS0***
pushButton	***pusBn***

Resource Type Name	Short Name
recorder	***recor***
refrigeration	***refrn***
relativeHumidity	***relHy***
rinseLevel	***rinLl***
runMode	***runMe***
signalStrength	***sigSh***
smokeSensor	***smoSr***
spinLevel	***spiLl***
televisionChannel	***telCl***
temperature	***tempe***
temperatureAlarm	***temAm***
timer	***timer***
turbo	***turbo***
waterFlow	***watFw***
waterLevel	***watLl***
waterSensor	***watSr***
weight	***weigt***
wind	***wind***

In protocol bindings resource type names for actions shall be translated into short names of table 6.3.2-3.

Table 6.3.2-3: Specialization type short names (Actions)

Resource Type Name	Short Name
activateClockTimer	***acCTr***
deactivateClockTimer	***deCTr***
downChannel	***dowCl***
downVolume	***dowVe***
toggle	***togge***
upChannel	***uphCl***

6.3.3 Resource attributes for properties and data points

In protocol bindings resource attributes names for properties of device models shall be translated into short names of table 6.3.3-1.

Table 6.3.3-1: Resource attribute short names (Device properties)

Attribute Name	Occurs in	Short Name
country	deviceAirConditioner, deviceClothesWasher, deviceOven, deviceRefrigerator, deviceRobotCleaner, deviceTelevision, deviceThermostat	***couny***
dateOfManufacture	deviceRefrigerator	***daOMe***
deviceAliasName	deviceAirConditioner, deviceClothesWasher, deviceOven, deviceRefrigerator, deviceRobotCleaner, deviceTelevision	***deANe***
deviceFirmwareVersion	deviceAirConditioner, deviceClothesWasher, deviceOven, deviceRefrigerator, deviceRobotCleaner, deviceTelevision, deviceThermostat	***deFVn***
deviceManufacturer	All devices	***devMr***
deviceModelName	All devices	***deMNe***
deviceName	deviceAirConditioner, deviceClothesWasher, deviceOven, deviceRefrigerator, deviceRobotCleaner, deviceTelevision, deviceThermostat	***devNe***
deviceSerialNum	All devices	***deSNm***
deviceSubModelName	deviceAirConditioner, deviceClothesWasher, deviceOven, deviceRefrigerator, deviceRobotCleaner, deviceTelevision	***dSMNe***
deviceType	All devices	***devTe***
generationSource	deviceMicrogeneration	***genSe***
hardwareVersion	deviceRefrigerator	***harVn***
location	deviceElectricVehicleCharger, deviceLight, deviceMicrogeneration, deviceSmartElectricMeter, deviceStorageBattery, deviceWaterHeater	***locan***

Attribute Name	Occurs in	Short Name
manufacturerDetailsLink	deviceRefrigerator	***maDLk***
manufacturerName	deviceRefrigerator,	***manNe***
measuringScope	deviceThermostat	***meaSe***
osVersion	deviceRefrigerator	***oseVn***
protocol	deviceElectricVehicleCharger, deviceLight, deviceMicrogeneration, deviceSmartElectricMeter, deviceStorageBattery, deviceWaterHeater	***protI***
supportURL	deviceRefrigerator	***suURL***
systemTime	deviceRefrigerator	***sysTe***

In protocol bindings resource attributes names for properties of module classes shall be translated into short names of table 6.3.3-2.

Table 6.3.3-2: Resource attribute short names (ModuleClass properties)

Attribute Name	Occurs in	Short Name
chargingCapacity	electricVehicleConnector	***chaCy***
dischargingCapacity	electricVehicleConnector	***disCy***
electricEnergy	battery	***eleEy***
material	battery	***matel***
voltage	battery	***volte***

In protocol bindings resource attributes names for data points of module classes shall be translated into short names of tables 6.3.3-3 and 6.3.3-4.

Table 6.3.3-3: Resource attribute short names (ModuleClass data points) (1/2)

Attribute Name	Occurs in	Short Name
absoluteEnergyConsumption	energyConsumption	***abECn***
absoluteStartTime	timer	***abSTe***
absoluteStopTime	timer	***abST0***
alarm	motionSensor, smokeSensor, temperatureAlarm, waterSensor	***alarm***
alarmStatus	alarmSpeaker	***alaSs***
availableChannels	televisionChannel	***avaCs***
bath	hotWaterSupply	***bath***
batteryThreshold	battery	***batTd***
blue	colour	***blue***
bone	bioelectricalImpedanceAnalysis	***bone***
brightness	brightness	***brigs***
capacity	battery	***capay***
channelId	televisionChannel	***chaId***
charging	battery	***charg***
code	faultDetection	***code***
colourSaturation	colourSaturation	***colSn***
current	energyConsumption	***currt***
currentDate	clock	***curDe***
currentTemperature	temperature	***curT0***
currentTime	clock	***curTe***
defrost	refrigeration	***defrt***
description	faultDetection	***descn***
desiredHumidity	relativeHumidity	***desHy***
detectedTime	smokeSensor	***detTe***
diastolicPressure	oximeter	***diaPe***
directionAuto	wind	***dirAo***
directionDown	wind	***dirDn***
directionLeft	wind	***dirLt***
directionRight	wind	***dirRt***
directionUp	wind	***dirUp***
discharging	battery	***discg***
doorState	doorStatus	***dooSe***
duration	recorder	***duran***
estimatedTimeToEnd	timer	***eTTEd***

Attribute Name	Occurs in	Short Name
fat	bioElectricalImpedanceAnalysis	***fat***
filterLifetime	refrigeration	***filLe***
frequency	energyConsumption	***freqy***
green	colour	***green***
height	height	***heigt***
inputSourceID	audioVideoInput	***inSId***
kcal	bioElectricalImpedanceAnalysis	***kcal***
keyNumber	keypad	***keyNr***
level	battery	***level***
light	alarmSpeaker	***light***
liquidLevel	waterLevel	***liqLl***
lowBattery	battery	***lowBy***
lqi	signalStrength	***lqi***
maxValue	audioVolume, temperature	***maxVe***
minValue	temperature	***minVe***
multiplyingFactors	energyConsumption, energyGeneration	***mulFs***
muscle	bioElectricalImpedanceAnalysis	***musce***
muteEnabled	audioVolume	***mutEd***
openAlarm	doorStatus	***opeAm***
openDuration	doorStatus	***opeDn***
operationMode	runMode	***opeMe***
oxygenSaturation	oximeter	***oxySn***

Table 6.3.3-4: Resource attribute short names (ModuleClass data points) (2/2)

Attribute Name	Occurs in	Short Name
power	energyConsumption	***power***
powerGenerationData	energyGeneration	***poGDa***
powerSaveEnabled	powerSave	***poSEd***
powerState	binarySwitch	***powSe***
previousChannel	televisionChannel	***preCl***
pulseRate	oximeter	***pulRe***
pushed	pushButton	***pusBn***
rapidCool	refrigeration	***rapCl***
rapidFreeze	refrigeration	***rapFe***
red	colour	***red***
referenceTimer	timer	***refTr***
relativeHumidity	relativeHumidity	***relHy***
resistance	bioElectricalImpedanceAnalysis	***resie***
rinseLevel	rinseLevel	***rinLl***
roundingEnergyConsumption	energyConsumption	***roECn***
roundingEnergyGeneration	energyGeneration	***roEGn***
rssi	signalStrength	***rssi***
runningTime	timer	***runTe***
sensitivity	motionSensor	***sensy***
significantDigits	energyConsumption, energyGeneration	***sigDs***
silentTime	motionSensor	***silTe***
status	boiler, electricVehicleConnector, faultDetection, hotWaterSupply	***stats***
stepValue	audioVolume, temperature	***steVe***
strength	spinLevel, waterFlow, wind	***streh***
supportedInputSources	audioVideoInput	***sulSs***
supportedModes	runMode	***supMs***
systolicPressure	oximeter	***sysPe***
targetDuration	timer	***tarDn***
targetTemperature	temperature	***tarTe***
targetTimeToStart	timer	***tTTSt***
targetTimeToStop	timer	***tTTSp***
temperature	temperatureAlarm	***tempe***
temperatureThreshhold	temperatureAlarm	***temTd***
tone	alarmSpeaker	***tone***
turboEnabled	turbo	***turEd***
unit	temperature	***unit***

Attribute Name	Occurs in	Short Name
visceraFat	bioElectricalImpedanceAnalysis	***visFt***
voltage	energyConsumption	***volte***
volumePercentage	audioVolume	***volPe***
water	bioElectricalImpedanceAnalysis	***water***
weight	weight	***weigt***

6.4 containerDefinition values

6.4.1 Introduction

Each specialization has a *containerDefinition* attribute which can be used as an unique identifier and contains the information of the resource. In this clause, the detailed values of *containerDenifition* attributes in every specializations for the home appliance information models are given.

6.4.2 Device models

The *containerDefinition* attribute of specializations for device models shall have the values specified in table 6.4.2-1.

Table 6.4.2-1: containerDefinition values of specializations for device models

Resource Type Name	***containerDefinition* value**
deviceAirConditioner	"org.onem2m.home.device.deviceAirConditioner"
deviceClothesWasher	"org.onem2m.home.device.deviceClothesWasher"
deviceElectricVehicleCharger	"org.onem2m.home.device.deviceElectricVehicleCharger"
deviceLight	"org.onem2m.home.device.deviceLight"
deviceMicrogeneration	"org.onem2m.home.device.deviceMicrogeneration"
deviceOven	"org.onem2m.home.device.deviceOven"
deviceRefrigerator	"org.onem2m.home.device.deviceRefrigerator"
deviceRobotCleaner	"org.onem2m.home.device.deviceRobotCleaner"
deviceSmartElectricMeter	"org.onem2m.home.device.deviceSmartElectricMeter"
deviceStorageBattery	"org.onem2m.home.device.deviceStorageBattery"
deviceTelevision	"org.onem2m.home.device.deviceTelevision"
deviceThermostat	"org.onem2m.home.device.deviceThermostat"
deviceWaterHeater	"org.onem2m.home.device.deviceWaterHeater"

6.4.3 ModuleClasses

The *containerDefinition* attribute of specializations for module classes shall have the values specified in table 6.4.3-1.

Table 6.4.3-1: containerDefinition values of specializations for module classes

Resource Type Name	***containerDefinition* value**
alarmSpeaker	"org.onem2m.home.moduleclass.alarmSpeaker"
audioVideoInput	"org.onem2m.home.moduleclass.audioVideoInput"
audioVolume	"org.onem2m.home.moduleclass.audioVolume"
battery	"org.onem2m.home.moduleclass.battery"
binarySwitch	"org.onem2m.home.moduleclass.binarySwitch"
bioElectricalImpedanceAnalysis	"org.onem2m.home.moduleclass.bioElectricalImpedanceAnalysis"
boiler	"org.onem2m.home.moduleclass.boiler"
brightness	"org.onem2m.home.moduleclass.brightness"
clock	"org.onem2m.home.moduleclass.clock"
colour	"org.onem2m.home.moduleclass.colour"
colourSaturation	"org.onem2m.home.moduleclass.colourSaturation"
customTemperature	"org.onem2m.home.moduleclass.customTemperature"
doorStatus	"org.onem2m.home.moduleclass.doorStatus"
electricVehicleConnector	"org.onem2m.home.moduleclass.electricVehicleConnector"
energyConsumption	"org.onem2m.home.moduleclass.energyConsumption"
energyGeneration	"org.onem2m.home.moduleclass.energyGeneration"

Resource Type Name	*containerDefinition* value
faultDetection	"org.onem2m.home.moduleclass.faultDetection"
freshTemperature	"org.onem2m.home.moduleclass.freshTemperature"
frozenTemperature	"org.onem2m.home.moduleclass.frozenTemperature"
height	"org.onem2m.home.moduleclass.height"
hotWaterSupply	"org.onem2m.home.moduleclass.hotWaterSupply"
keypad	"org.onem2m.home.moduleclass.keypad"
motionSensor	"org.onem2m.home.moduleclass.motionSensor"
oximeter	"org.onem2m.home.moduleclass.oximeter"
powerSave	"org.onem2m.home.moduleclass.powerSave"
pushButton	"org.onem2m.home.moduleclass.pushButton"
recorder	"org.onem2m.home.moduleclass.recorder"
refrigeration	"org.onem2m.home.moduleclass.refrigeration"
relativeHumidity	"org.onem2m.home.moduleclass.relativeHumidity"
rinseLevel	"org.onem2m.home.moduleclass.rinseLevel"
runMode	"org.onem2m.home.moduleclass.runMode"
signalStrength	"org.onem2m.home.moduleclass.signalStrength"
smokeSensor	"org.onem2m.home.moduleclass.smokeSensor "
spinLevel	"org.onem2m.home.moduleclass.spinLevel"
televisionChannel	"org.onem2m.home.moduleclass.televisionChannel"
temperature	"org.onem2m.home.moduleclass.temperature"
temperatureAlarm	"org.onem2m.home.moduleclass.temperatureAlarm"
timer	"org.onem2m.home.moduleclass.timer"
turbo	"org.onem2m.home.moduleclass.turbo"
waterFlow	"org.onem2m.home.moduleclass.waterFlow"
waterLevel	"org.onem2m.home.moduleclass.waterLevel"
waterSensor	"org.onem2m.home.moduleclass.waterSensor"
weight	"org.onem2m.home.moduleclass.weight"
wind	"org.onem2m.home.moduleclass.wind"

6.4.4 Actions

The *containerDefinition* attribute of specializations for actions shall have the values specified in table 6.4.4-1.

Table 6.4.4-1: containerDefinition values of specializations for actions

Resource Type Name	*containerDefinition* value
activateClockTimer	"org.onem2m.home.moduleclass.timer.activateclocktimer"
deactivateClockTimer	"org.onem2m.home.moduleclass.timer.deactivateclocktimer"
downChannel	"org.onem2m.home.moduleclass.televisionchannel.downchannel"
downVolume	"org.onem2m.home.moduleclass.audiovolume.downvolume"
toggle	"org.onem2m.home.moduleclass.binaryswitch.timer"
upChannel	"org.onem2m.home.moduleclass.televisionchannel.upchannel"

6.5 XSD definitions

6.5.1 Introduction

The present clause provides a list of files which define data types in XSD for Device models, ModuleClasss, and Actions.

Generation process of XSD file is explained in annex A using some examples.

6.5.2 XSD definitions for Device models

The XSD definitions for Device models are listed in table 6.5.2-1.

Table 6.5.2-1: Data type definition of Device models

Device model	File Name	Note
deviceAirConditioner	HD-deviceAirConditioner-v<TS-version>.xsd	
deviceClothesWasher	HD-deviceClothesWasher-v<TS-version>.xsd	
deviceElectricVehicleCharger	HD-deviceElectricVehicleCharger-v<TS-version>.xsd	
deviceLight	HD-deviceLight-v<TS-version>.xsd	
deviceMicrogeneration	HD-deviceMicrogeneration-v<TS-version>.xsd	
deviceOven	HD-deviceOven-v<TS-version>.xsd	
deviceRefrigerator	HD-deviceRefrigerator-v<TS-version>.xsd	
deviceRobotCleaner	HD-deviceRobotCleaner-v<TS-version>.xsd	
deviceStorageBattery	HD-deviceStroageBattery-v<TS-version>.xsd	
deviceSmartElectricMeter	HD-deviceSmartElectricMeter-v<TS-version>.xsd	
deviceTelevision	HD-deviceTelevision-v<TS-version>.xsd	
deviceThermostat	HD-deviceThermostat-v<TS-version>.xsd	
deviceWaterHeater	HD-deviceWaterHeater-v<TS-version>.xsd	
NOTE: The string '<TS-version>' shall be interpreted as the version of the present document.		

6.5.3 XSD definitions for ModuleClass

The XSD definitions for ModuleClass are listed in table 6.5.3-1.

Table 6.5.3-1: Data type definition of ModuleClasses

ModuleClass ID	File Name	Note
alarmSpeaker	HD-mod-alarmSpeaker-v<TS-version>.xsd	
audioVideoInput	HD-mod-audioVideoInput-v<TS-version>.xsd	
audioVolume	HD-mod-audioVolume-v<TS-version>.xsd	
battery	HD-mod-battery-v<TS-version>.xsd	
binarySwitch	HD-mod-binarySwitch-v<TS-version>.xsd	
bioElectricalImpedanceAnalysis	HD-mod-bioElectricalImpedanceAnalysis-v<TS-version>.xsd	
boiler	HD-mod-boiler-v<TS-version>.xsd	
brightness	HD-mod-brightness-v<TS-version>.xsd	
clock	HD-mod-clock-v<TS-version>.xsd	
colour	HD-mod-colour-v<TS-version>.xsd	
colourSaturation	HD-mod-colourSaturation-v<TS-version>.xsd	
doorStatus	HD-mod-doorStatus-v<TS-version>.xsd	
electricVehicleConnector	HD-mod-electricVehicleConnector-v<TS-version>.xsd	
energyConsumption	HD-mod-energyConsumption-v<TS-version>.xsd	
energyGeneration	HD-mod-energyGeneration-v<TS-version>.xsd	
faultDetection	HD-mod-faultDetection-v<TS-version>.xsd	
height	HD-mod-height-v<TS-version>.xsd	
hotWaterSupply	HD-mod-hotWaterSupply-v<TS-version>.xsd	
keypad	HD-mod-keypad-v<TS-version>.xsd	
motionSensor	HD-mod-motionSensor-v<TS-version>.xsd	
oximeter	HD-mod-oximeter-v<TS-version>.xsd	
powerSave	HD-mod-powerSave-v<TS-version>.xsd	
pushButton	HD-mod-pushButton-v<TS-version>.xsd	
recorder	HD-mod-recorder-v<TS-version>.xsd	
refrigeration	HD-mod-refrigeration-v<TS-version>.xsd	
relativeHumidity	HD-mod-relativeHumidity-v<TS-version>.xsd	
rinseLevel	HD-mod-rinseLevel-v<TS-version>.xsd	
runMode	HD-mod-runMode-v<TS-version>.xsd	
signalStrength	HD-mod-signalStrength-v<TS-version>.xsd	
smokeSensor	HD-mod-smokeSensor-v<TS-version>.xsd	
spinLevel	HD-mod-spinLevel-v<TS-version>.xsd	
televisionChannel	HD-mod- HD-mod--v<TS-version>.xsd	
temperature	HD-mod-temperature-v<TS-version>.xsd	
temperatureAlarm	HD-mod-temperatureAlarm-v<TS-version>.xsd	
timer	HD-mod-timer-v<TS-version>.xsd	
turbo	HD-mod-turbo-v<TS-version>.xsd	
waterFlow	HD-mod-waterFlow-v<TS-version>.xsd	

ModuleClass ID	File Name	Note
waterLevel	HD-mod-waterLevel-v<TS-version>.xsd	
waterSensor	HD-mod-waterSensor-v<TS-version>.xsd	
weight	HD-mod-weight-v<TS-version>.xsd	
wind	HD-mod-wind-v<TS-version>.xsd	
NOTE: The string '<TS-version>' shall be interpreted as the version of the present document.		

6.5.4 XSD definitions for Action

The XSD definitions for Actions are listed in table 6.5.4-1.

Table 6.5.4-1: Data type definition of Action

Action	File Name	Note
activateClockTimer	HD-act-activateClockTimer-v<TS-version><TS-version>.xsd	
deactivateClockTimer	HD-act-deactivateClockTimer-v<TS-version>.xsd	
downChannel	HD-act-downChannel-v<TS-version>.xsd	
downVolume	HD-act-downVolume-v<TS-version>.xsd	
toggle	HD-act-toggle-v<TS-version>.xsd	
upChannel	HD-act-upChannel-v<TS-version>.xsd	
upVolume	HD-act-upVolume-v<TS-version>.xsd	
NOTE: The string '<TS-version>' shall be interpreted as the version of the present document.		

7 Mapping with Other Information Models from External Organizations

This clause is intended to specify how the Home Appliance Information Model (HAIM) defined in the clause 5 can be mapped with existing external models from AllJoyn, OIC, ECHONET, etc. The mapping is to enable the interworking between the oneM2M system and external technologies at the information model level. This means a oneM2M native application which understands only oneM2M standardized HAIM can interact with non-oneM2M home appliances of different technologies in a consistent way without knowing the technology specific details. An IPE is responsible for translating the HAIM to/from technology specific information model bidirectional following the mapping specification in this clause. Using HAIM as a bridge, home appliances and applications of different technologies can also interact with each other via the oneM2M system (with IPEs).

The mapping details will be specified in future release. ModuleClasses, properties, device models and data types in HAIM may be further updated if gaps are identified for mapping.

8 Ontology for the Home Appliance Information Model aligned with oneM2M Base Ontology

Table 8-1 shows a mapping of the Home Appliance Information Model to the oneM2M Base Ontology in ETSI TS 118 112 [i.5].

The table only shows mapping of SDT concepts that are used to classify all concepts in the Home Appliance Information Model. Therefore, since any concept in the Home Appliance Information Model can be classified according to a specific SDT concept it also (transitively) maps to the related class of the oneM2M Base Ontology.

Table 8-1: Mapping between SDT concepts in the Home Appliance Information Model and the oneM2M Base Ontology

SDT Concept in the Home Appliance Information Model	Mapping relationship	Class in Base Ontology	Property in Base Ontology	Comment
SDT: Device	sub-class of	Device		
SDT: SubDevice	sub-class of	Device		The base ontology allows a Device to consist of (sub-) Devices.
SDT: Action	sub-class of	Operation		
SDT: Args (of an Action)	sub-class of	OperationInput		
SDT: ReturnType (of an Action)	sub-class of	OperationOutput		
SDT: Event	sub-class of	Operation		
SDT: Data (of an Event)	sub-class of	OutputDataPoint		
SDT: Module	sub-class of	Service		The base ontology allows a Service to have subServices. Each SDT: Module implements one SDT: ModuleClass. Therefore SDT: Module can be considered a subclass of SDT: ModuleClass and therefore subclass of oneM2M: Service. See note.
SDT: ModuleClass	sub-class of	Service		See note
SDT: UnitOfMeasure	sub-class of	MetaData		
SDT: DataPoint	sub-class of	InputDataPoint		If SDT: DataPoint is writable.
SDT: DataPoint	sub-class of	OutputDataPoint		If SDT :DataPoint is readable.
SDT: Property (of a Device)	sub-class of	ThingProperty		
SDT: Property (of a ModuleClass)	sub-class of	Aspect		Aspect (of the Functionality).
SDT: SimpleType	sub-property of		hasDataType	The base ontology's SimpleTypeVariable class has data properties: • hasDataType; • hasDataRestriction.
SDT: Constraint	sub-property of		hasDataRestriction	
NOTE: In RESTful technologies the Service (i.e. the electronic representation of a Functionality in a network) is implicitly bound to its Functionality by the naming of the used resources (e.g. the Functionality of ModuleClass "AudioVolume" is implemented as a Service through CRUD operations on a [audioVolume] <flexContainer> specialization).				

Annex A (informative): Resource Mapping Examples

A.1 Introduction

The AE may construct oneM2M resource tree on hosting CSE as the mapping of associated device, and each XSD definition for the device information models is generated following 'Resource Mapping Rule' in clause 6.2.

The present clause explains how to use the oneM2M resource tree to map Device model for each device (see clause 5.4).

A.2 Example for Device model 'deviceAirConditioner'

The present clause explains the creation process for the device typed 'deviceAirConditioner' (see clause 5.4.1 for device model definition of 'airConditioner').

Using the definition, 'deviceAirConditioner' model is mapped to [deviceAirConditioner] resource which is a specialization of <flexContainer> resource (see figure A.2-1).

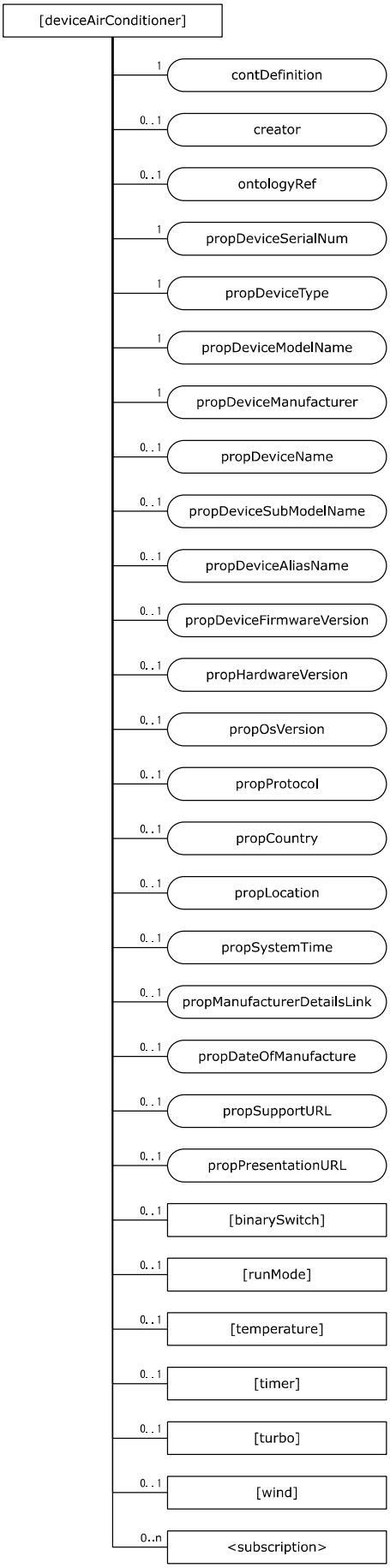

Figure A.2-1: Structure of *[deviceAirConditioner]* resource

The AE creates the [deviceAirConditioner] specialization of <flexContainer> resource for the Device model [deviceAirConditioner] resource.

The [deviceAirConditioner] resource contains the child resource specified in table A.2-1.

Table A.2-1: Child resources of *[deviceAirConditioner]* resource

Child Resources of *[airConditioner]*	Child Resource Type	Multiplicity	Description
[variable]	*<flexContainer> as defined in the specialization [binarySwitch]*	0..1	This resource is used to map 'binarySwith' ModuleClass defined in clause 5.
[variable]	*<flexContainer> as defined in the specialization [runMode]*	0..1	This resource is used to map 'runMode' ModuleClass defined in clause 5.
[variable]	*<flexContainer> as defined in the specialization [temperature]*	0..1	This resource is used to map 'temperature' ModuleClass defined in clause 5.
[variable]	*<flexContainer> as defined in the specialization [timer]*	0..1	This resource is used to map 'timer' ModuleClass defined in clause 5.
[variable]	*<flexContainer> as defined in the specialization [turbo]*	0..1	This resource is used to map 'turbo' ModuleClass defined in clause 5.
[variable]	*<flexContainer> as defined in the specialization [wind]*	0..1	This resource is used to map 'wind' ModuleClass defined in clause 5.
[variable]	*<subscription>*	0..n	See clause 9.6.8 in ETSI TS 118 101 [i.3].

The [deviceAirConditioner] specialization of <flexContainer> will contain [customAttributes] with variable name for each Properties. Thus, attributes of [deviceAirConditioner] resource are defined in table A.2-2.

Table A.2-2: Attributes of *[deviceAirConditioner]* resource

Attributes of *[airConditioner]*	Multiplicity	RW/ RO/ WO	Description
resourceType	1	RO	See clause 9.6.1.3 in ETSI TS 118 101 [i.3].
resourceID	1	RO	See clause 9.6.1.3 in ETSI TS 118 101 [i.3].
resourceName	1	RO	See clause 9.6.1.3 in ETSI TS 118 101 [i.3].
parentID	1	RO	See clause 9.6.1.3 in ETSI TS 118 101 [i.3].
expirationTime	1	RW	See clause 9.6.1.3 in ETSI TS 118 101 [i.3].
accessControlPolicyIDs	0..1 (L)	RW	See clause 9.6.1.3 in ETSI TS 118 101 [i.3].
creationTime	1	RO	See clause 9.6.1.3 in ETSI TS 118 101 [i.3].
lastModifiedTime	1	RO	See clause 9.6.1.3 in ETSI TS 118 101 [i.3].
labels	0..1	RW	See clause 9.6.1.3 in ETSI TS 118 101 [i.3].
containerDefinition	1	WO	The value is "org.onem2m.home.device.airconditioner".
Creator	0..1 (L)	RW	See clause 9.6.35 in ETSI TS 118 101 [i.3].
ontologyRef	0..1 (L)	RW	See clause 9.6.35 in ETSI TS 118 101 [i.3].
propDeviceSerialNum	1	RO	This [customAttribute] is used to map Property 'deviceSerialNum'.
propDeviceType	1	RO	This [customAttribute] is used to map Property 'deviceType'.
propDeviceModelName	1	RO	This [customAttribute] is used to map Property 'deviceModelName'.
propDeviceManifacture		RO	This [customAttribute] is used to map Property 'devicManufacture'.
propDeviceName	0..1	RO	This [customAttribute] is used to map Property 'deviceName'.
propDeviceSubModelName	0..1	RO	This [customAttribute] is used to map Property 'deviceSubModelName'.
propDeviceAliasName	0..1	RO	This [customAttribute] is used to map Property 'deviceAliasName'.
propDeviceFirmwareVersion	0..1	RO	This [customAttribute] is used to map Property 'deviceFirmwareVersion'.
propDeviceHardwareVersion	0..1	RO	This [customAttribute] is used to map Property 'deviceHardwareVersion'.
propOsVersion	0..1	RO	This [customAttribute] is used to map Property 'osVersion'.
propLocation	0..1	RO	This [customAttribute] is used to map Property 'location'.
propDateOfManufacture	0..1	RO	This [customAttribute] is used to map Property 'dateOfManufacture'.
propSupportURL	0..1	RO	This [customAttribute] is used to map Property 'SupportURL'.
propPresentationURL	0..1	RO	This [customAttribute] is used to map Property 'presentationURL'.

A.3 Example of ModuleClass 'binarySwitch'

The [*binarySwitch*] resource is used to share information regarding the modeled binary switch module as a ModuleClass. The [*binarySwitch*] resource is a specialization of the <*flexContainer*> resource.

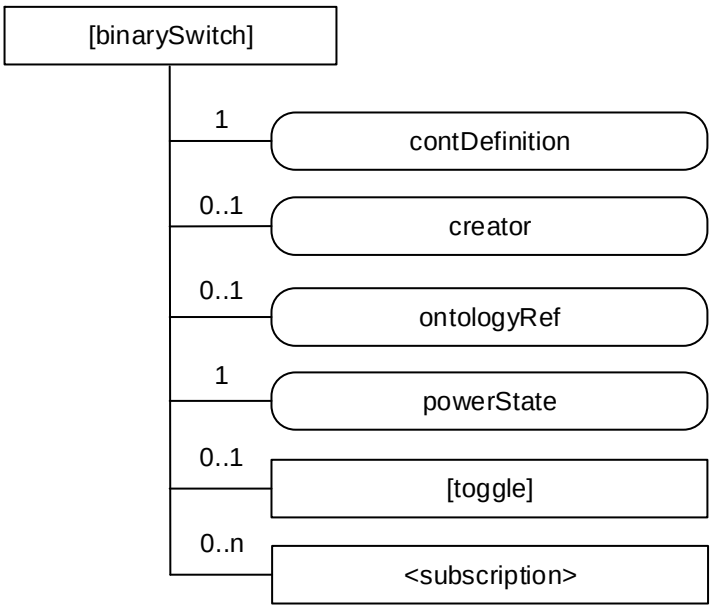

Figure A.3-1: Structure of *[binarySwitch]* resource

The *[binarySwitch]* resource contains the child resource specified in table A.3-1.

Table A.3-1: Child resources of *[binarySwitch]* resource

Child Resources of *[binarySwitch]*	Child Resource Type	Multiplicity	Description
[variable]	*<flexContainer> as defined in the specialization [toggle]*	0..1	This resource is used to map 'toggle' Action defined in clause 5.3.5.
[variable]	*<subscription>*	0..n	See clause 9.6.8 in ETSI TS 118 101 [i.3]

The *[binarySwitch]* resource contains the attributes specified in table A.3-2.

Table A.3-2: Attributes of *[binarySwitch]* resource

Attributes of *[binarySwitch]*	Multiplicity	RW/ RO/ WO	Description
resourceType	1	RO	See clause 9.6.1.3 in ETSI TS 118 101 [i.3]
resourceID	1	RO	See clause 9.6.1.3 in ETSI TS 118 101 [i.3]
resourceName	1	RO	See clause 9.6.1.3 in ETSI TS 118 101 [i.3]
parentID	1	RO	See clause 9.6.1.3 in ETSI TS 118 101 [i.3]
expirationTime	1	RW	See clause 9.6.1.3 in ETSI TS 118 101 [i.3]
accessControlPolicyIDs	0..1 (L)	RW	See clause 9.6.1.3 in ETSI TS 118 101 [i.3]
creationTime	1	RO	See clause 9.6.1.3 in ETSI TS 118 101 [i.3]
lastModifiedTime	1	RO	See clause 9.6.1.3 in ETSI TS 118 101 [i.3]
labels	0..1	RW	See clause 9.6.1.3 in ETSI TS 118 101 [i.3]
containerDefinition	1	WO	The value is "org.onem2m.home.moduleclass.binaryswitch"
creator	0..1 (L)	RW	See clause 9.6.35 in ETSI TS 118 101 [i.3]
ontologyRef	0..1 (L)	RW	See clause 9.6.35 in ETSI TS 118 101 [i.3]
powerState	1	RW	See clause 5.3.5

A.4 Example of Action 'toggle'

The [*toggle*] resource is used to share information regarding the modeled toggle as an Action. The [*toggle*] resource is a specialization of the <*flexContainer*> resource.

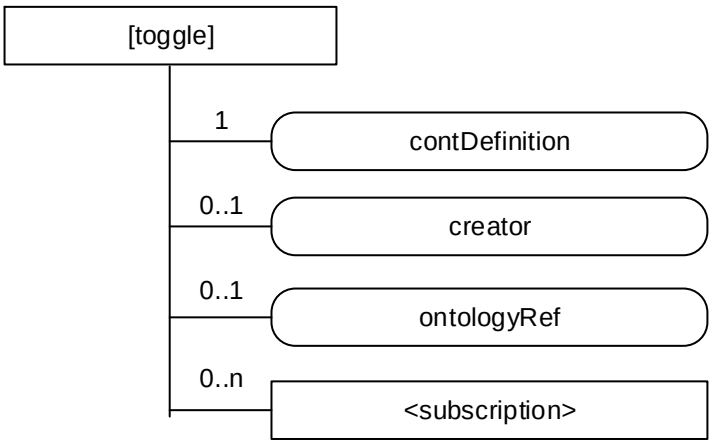

Figure A.4-1: Structure of *[toggle]* resource

The *[toggle]* resource contains the child resource specified in table A.4-1.

Table A.4-1: Child resources of *[toggle]* resource

Child Resources of *[toggle]*	Child Resource Type	Multiplicity	Description
[variable]	*<subscription>*	0..n	See clause 9.6.8 in ETSI TS 118 101 [i.3]

The *[toggle]* resource contains the attributes specified in table A.4-2.

Table A.4-2: Attributes of *[toggle]* resource

Attributes of *[toggle]*	Multiplicity	RW/ RO/ WO	Description
resourceType	1	RO	See clause 9.6.1.3 in ETSI TS 118 101 [i.3]
resourceID	1	RO	See clause 9.6.1.3 in ETSI TS 118 101 [i.3]
resourceName	1	RO	See clause 9.6.1.3 in ETSI TS 118 101 [i.3]
parentID	1	RO	See clause 9.6.1.3 in ETSI TS 118 101 [i.3]
expirationTime	1	RW	See clause 9.6.1.3 in ETSI TS 118 101 [i.3]
accessControlPolicyIDs	0..1 (L)	RW	See clause 9.6.1.3 in ETSI TS 118 101 [i.3]
creationTime	1	RO	See clause 9.6.1.3 in ETSI TS 118 101 [i.3]
lastModifiedTime	1	RO	See clause 9.6.1.3 in ETSI TS 118 101 [i.3]
labels	0..1	RW	See clause 9.6.1.3 in ETSI TS 118 101 [i.3]
containerDefinition	1	WO	The value is "org.onem2m.home.moduleclass.binaryswitch.toggle"
creator	0..1 (L)	RW	See clause 9.6.35 in ETSI TS 118 101 [i.3]
ontologyRef	0..1 (L)	RW	See clause 9.6.35 in ETSI TS 118 101 [i.3]

History

Document history		
V2.0.0	September 2016	Publication
V2.0.2	March 2020	Publication