

SensorManagement Sensor DataModel Service

For UPnP Version 1.0

Status: Standardized DCP (SDCP)

Date: February 1, 2015

Document Version: 1.0

Service Template Version: 2.00

This Standardized DCP has been adopted as a Standardized DCP by the Steering Committee of the UPnP Forum, pursuant to Section 2.1(c)(ii) of the UPnP Forum Membership Agreement. UPnP Forum Members have rights and licenses defined by Section 3 of the UPnP Forum Membership Agreement to use and reproduce the Standardized DCP in UPnP Compliant Devices. All such use is subject to all of the provisions of the UPnP Forum Membership Agreement.

THE UPNP FORUM TAKES NO POSITION AS TO WHETHER ANY INTELLECTUAL PROPERTY RIGHTS EXIST IN THE STANDARDIZED DCPS. THE STANDARDIZED DCPS ARE PROVIDED "AS IS" AND "WITH ALL FAULTS". THE UPNP FORUM MAKES NO WARRANTIES, EXPRESS, IMPLIED, STATUTORY, OR OTHERWISE WITH RESPECT TO THE STANDARDIZED DCPS, INCLUDING BUT NOT LIMITED TO ALL IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT AND FITNESS FOR A PARTICULAR PURPOSE, OF REASONABLE CARE OR WORKMANLIKE EFFORT, OR RESULTS OR OF LACK OF NEGLIGENCE.

© 2015 UPnP Forum. All Rights Reserved.

Authors <sup>a</sup>	Company
Clarke Stevens	Cablelabs
Jangwook Park (Vice-Chair)	LGE
Paul Jeon (Vice-Chair)	LGE
Russell Berkoff (Chair)	Samsung Electronics
Danilo Santos	Signove
Gerhard Mekenkamp	TPVision
Wouter van der Beek	Cisco Systems Inc
<sup>a</sup> The UPnP forum in no way guarantees the accuracy or completeness of this author list and in no way implies any rights for or support from those members listed. This list is not the specifications' contributor list that is kept on the UPnP Forum's website.	

## CONTENTS

1	Scope.....	5
2	Normative References .....	5
3	Terms, Definitions and Abbreviations .....	6
4	Notations and conventions .....	6
4.1	Notation .....	6
4.2	Data Types.....	7
4.3	Vendor-defined Extensions.....	7
4.3.1	Extended Backus-Naur Form .....	7
	SensorManagement ConfigurationManagement Service Profile .....	9
4.4	Service Type .....	9
4.5	SensorManagement Device Architecture .....	9
4.5.1	Sensor Description .....	9
4.5.2	Sensor Data Interface.....	9
4.5.3	Sensor Data Records .....	9
4.5.4	SensorURNs.....	9
4.5.5	Sensor Required DataItems.....	9
4.5.6	Sensor Normative Type Identifiers .....	10
4.5.7	Sensor Event Model .....	11
4.6	State Variables.....	13
4.7	Actions .....	13
4.7.1	Introduction .....	13
4.7.2	<a href="#"><u>GetSupportedDataModels()</u></a> .....	14
4.7.3	<a href="#"><u>GetSelectedValues()</u></a> .....	15
4.7.4	<a href="#"><u>SetValues()</u></a> .....	15
4.7.5	<a href="#"><u>CreateInstance()</u></a> .....	15
4.7.6	<a href="#"><u>DeleteInstance()</u></a> .....	15
4.7.7	<a href="#"><u>SetAttributes()</u></a> .....	16
	Annex A SensorManagement General Data Model (normative) .....	17
	Annex B Required Sensor DataItem(s) (normative) .....	25
	Annex C Common Device Identifiers (normative) .....	26
	Annex D SensorManagement IEEE-11073 Personal Health Devices .....	31
	Annex E Common DataItem Names (Normative) .....	44
	Annex F Location.....	76
	Figure D.1 — Blood Pressure Monitor – Medical Device System (Informative) .....	31
	Figure D.2 — Medical Device System with PM-store Object (informative) .....	39
	Table 1 — EBNF Operators .....	8
	Table 0-1 — Sensor URN [identifier-type] values .....	10
	Table 0-2 — Sensor URN [identifier-type-dependent] values.....	10
	Table 0-3 — <SensorEvents> event= attribute allowed values .....	12
	Table 0-4 — State Variables for Eventing .....	13
	Table 0-5 — Actions .....	14

## SensorDataModel:1

Table 0-6 — SensorManagement General Parameters.....	17
Table A.1 — SensorEventEnable parameter allowed values .....	20
Table A.7 — Sensor Permissions.....	23
Table B.1 — SensorManagement required Dataltem(s).....	25
Table C.1-1 — Common Device Identifiers.....	26
Table D.1 — IEEE-11073 specific Medical Device System Parameters .....	32
Table D.2 — IEEE-11073 Medical Device Sensor Parameters .....	35
Table D.3 — Default Dataltem(s) for IEEE-11073 Medical Object Class Sensors .....	36
Table D.4 — Dataltem(s) for Medical Object Class Attributes.....	37
Table D.5 — Dataltem(s) for Medical Object Class Observations .....	38
Table D.6 — IEEE-11073 Persistent Metric Store Parameter Nodes .....	39
Table D.7 — IEEE-11073 Persistent Metric Segment Parameter Nodes .....	41
Table D.8 — IEEE-11073 Persistent Metric Segment Dataltem(s).....	42
Table E.3-10 — Dataltem <u>Acceleration</u> definition .....	46
Table E.3-11 — Dataltem <u>Angle</u> definition .....	47
Table E.3-12 — Dataltem <u>Area</u> definition .....	47
Table E.3-13 — Dataltem <u>Brightness</u> definition.....	48
Table E.3-14 — Dataltem <u>ColorRGB</u> definition.....	49
Table E.3-15 — Dataltem <u>ControlValve</u> definition .....	49
Table E.3-16 — Dataltem <u>Count</u> definition .....	50
Table E.3-17 — Dataltem <u>Current</u> definition.....	50
Table E.3-18 — Dataltem <u>Direction</u> definition.....	51
Table E.3-19 — Dataltem <u>Flow</u> definition .....	52
Table E.3-20 — Dataltem <u>FlowDirection</u> definition .....	52
Table E.3-21 — Dataltem <u>Frequency</u> definition .....	53
Table E.3-22 — Dataltem <u>GPS</u> definition .....	53
Table E.3-23 — Dataltem <u>Humidity</u> definition.....	54
Table E.3-24 — Dataltem <u>Mass</u> definition .....	55
Table E.3-25 — Dataltem <u>Moisture</u> definition .....	55
Table E.3-26 — Dataltem <u>Percentage</u> definition.....	56
Table E.3-27 — Dataltem <u>Position</u> definition .....	57
Table E.3-28 — Dataltem <u>Power</u> definition.....	57
Table E.3-29 — Dataltem <u>PowerSwitch</u> definition .....	58
Table E.3-30 — Dataltem <u>Presence</u> definition.....	58
Table E.3-31 — Dataltem <u>Pressure</u> definition .....	59
Table E.3-32 — Dataltem <u>Proximity</u> definition.....	60
Table E.3-33 — Dataltem RelPosition definition.....	60
Table E.3-34 — Dataltem <u>RelProximity</u> definition.....	61
Table E.3-35 — Dataltem <u>RelSpeed</u> definition .....	62
Table E.3-36 — Dataltem <u>Saturation</u> definition .....	62
Table E.3-37 — Dataltem <u>Speed</u> definition .....	63

## SensorDataModel:1

Table E.3-38 — Dataltem <u>SpeedDirection</u> definition.....	64
Table E.3-39 — Dataltem <u>Switch</u> definition .....	64
Table E.3-40 — Dataltem <u>Temperature</u> definition.....	65
Table E.3-41 — Dataltem <u>Vibration</u> definition.....	66
Table E.3-42 — Dataltem <u>Voltage</u> definition.....	66
Table E.3-43 — Dataltem <u>Voltage_dc</u> definition .....	67
Table E.3-44 — Dataltem <u>Volume</u> definition.....	67
Table E.3-45 — Dataltem <u>WRDirection</u> definition .....	68
Table E.3-46 Dataltem <u>BatteryState</u> definition.....	69
Table E.3-47 Dataltem <u>CleanerState</u> definition .....	70
Table E.3-48 Dataltem <u>CloseToDevice</u> definition .....	71
Table E.3-49 Dataltem <u>DryingState</u> definition.....	71
Table E.3-50 Dataltem <u>Location</u> definition.....	72
Table E.3-51 Dataltem <u>Lock</u> definition .....	73
Table E.3-52 Dataltem <u>MagnetoMeter</u> definition .....	73
Table E.3-53 Dataltem WashProgram definition.....	74
Table E.3-54 Dataltem <u>WashState</u> definition.....	75

## 1 Scope

This part of Publicly Available Specification ISO/IEC 29341 specifies Sensor Data Models. It is intended to be an adjunct document to the Sensor Management Device [11] and DataStore Service [13] specifications.

## 2 Normative References

The following referenced documents are indispensable for the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

[1] UPnP Device Architecture, version 1.0, UPnP Forum, June 13, 2000. Available at: [http://upnp.org/specs/arch/UPnPDA10\\_20000613.pdf](http://upnp.org/specs/arch/UPnPDA10_20000613.pdf). Latest version available at: <http://upnp.org/specs/arch/UPnP-arch-DeviceArchitecture-v1.0.pdf>.

[2] ISO 8601 Data elements and interchange formats – Information interchange -- Representation of dates and times, International Standards Organization, December 21, 2000. Available at: [ISO 8601:2000](http://www.iso.org/iso/8601).

[3] IETF RFC 2119, Key words for use in RFCs to Indicate Requirement Levels, S. Bradner, 1997. Available at: <http://www.faqs.org/rfcs/rfc2119.html>.

[4] HyperText Transport Protocol – HTTP/1.1, R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, T. Berners-Lee, June 1999. Available at: <http://www.ietf.org/rfc/rfc2616.txt>.

[5] IETF RFC 3339, Date and Time on the Internet: Timestamps, G. Klyne, Clearswift Corporation, C. Newman, Sun Microsystems, July 2002. Available at: <http://www.ietf.org/rfc/rfc3339.txt>.

[6] Extensible Markup Language (XML) 1.0 (Third Edition), François Yergeau, Tim Bray, Jean Paoli, C. M. Sperberg-McQueen, Eve Maler, eds., W3C Recommendation, February 4, 2004. Available at: <http://www.w3.org/TR/2004/REC-xml-20040204>.

[7] XML Schema Part 2: Data Types, Second Edition, Paul V. Biron, Ashok Malhotra, W3C Recommendation, 28 October 2004. Available at: <http://www.w3.org/TR/2004/REC-xmlschema-2-20041028>.

[8] ISO/IEEE-11073-20601 Health informatics - Personal health device communication - Part 20601: Application Profile - Optimized exchange protocol, 2010. Available at: <http://www.iso.org/iso/search.htm?qt=11073&searchSubmit=Search&sort=rel&type=simple&published=true>

[9] ISO/IEEE-11073-104zz Health informatics -- Personal health device communication -- Part 104zz: Device specialization -- <see specific specialization>. Available at: <http://www.iso.org/iso/search.htm?qt=11073&searchSubmit=Search&sort=rel&type=simple&published=true>

[10] UPnP Sensor and DataStore Architecture Overview, UPnP Forum, July 1, 2013. Available at: <http://www.upnp.org/specs/smg/UPnP-smgt-SensorManagementArchitectureOverview-v1-20130701.pdf>. Latest version available at: <http://www.upnp.org/specs/ehs/UPnP-smgt-SensorManagementArchitectureOverview-v1.pdf>.

[11] UPnP SensorManagement:1 Device, UPnP Forum July 1, 2013. Available at: <http://www.upnp.org/specs/smg/UPnP-smgt-SensorManagement-v1-Device-20130701.pdf>. Latest version available at: <http://www.upnp.org/specs/smg/UPnP-smgt-SensorManagement-v1-Device.pdf>.

## SensorDataModel:1

- [12] UPnP SensorGenericTransport:1 Service, UPnP Forum July 1, 2013. Available at: <http://www.upnp.org/specs/smgt/UPnP-smgt-SensorTransportGeneric-v1-Service-20130701.pdf>. Latest version available at: <http://www.upnp.org/specs/smgt/UPnP-smgt-SensorTransportGeneric-v1-Service.pdf>.
- [13] UPnP DataStore:1 Service, UPnP Forum, July 1, 2013. Available at: <http://www.upnp.org/specs/smgt/UPnP-smgt-DataStore-v1-Service-20130701.pdf>. Latest version available at: <http://www.upnp.org/specs/smgt/UPnP-smgt-DataStore-v1-Service.pdf>.
- [14] UPnP SensorManagement Sensor DataModel Service, UPnP Forum, July 1, 2013. Available at: <http://www.upnp.org/specs/smgt/UPnP-smgt-SensorDataModel-v1-Service-20130701.pdf>. Latest version available at: <http://www.upnp.org/specs/smgt/UPnP-smgt-SensorDataModel-v1-Service.pdf>.
- [15] UPnP DeviceProtection:1 Service, UPnP Forum, February 24, 2011. Available at: <http://www.upnp.org/specs/gw/UPnP-gw-DeviceProtection-v1-Service-20110224.pdf>. Latest version available at: <http://www.upnp.org/specs/gw/UPnP-gw-DeviceProtection-v1-Service.pdf>.
- [16] UPnP ConfigurationManagement:2 Service, UPnP Forum, December 31, 2010. Available at: <http://www.upnp.org/specs/dm/UPnP-dm-ConfigurationManagement-v2-Service-20120216.pdf>. Latest version available at: <http://www.upnp.org/specs/dm/UPnP-dm-ConfigurationManagement-v2-Service.pdf>.
- [17] XML Schema UPnP SensorManagement Sensor DataModel Events, UPnP Forum, July 1, 2013. Available at: <http://www.upnp.org/schemas/smgt/sdmevent-v1-20130701.xsd>. Latest version available at: <http://www.upnp.org/schemas/smgt/sdmevent.xsd>.
- [18] XML Schema UPnP SensorManagement Sensor DataModel DataItem Description, UPnP Forum, July 1, 2013. Available at: <http://www.upnp.org/schemas/smgt/sdmdid-v1-20130701.xsd>. Latest version available at: <http://www.upnp.org/schemas/smgt/sdmdid.xsd>.
- [19] XML Schema UPnP SensorManagement DataRecord Information, UPnP Forum, July 1, 2013. Available at: <http://www.upnp.org/schemas/smgt/srecinfo-v1-20130701.xsd>. Latest version available at: <http://www.upnp.org/schemas/smgt/srecinfo.xsd>.
- [20] ISO/IEC 14977, *Information technology - Syntactic metalanguage - Extended BNF*, December 1996.

## 3 Terms, Definitions and Abbreviations

For the purposes of this document, the terms and definitions given in UPnP Device Architecture, version 1.0 [1] apply.

## 4 Notations and conventions

### 4.1 Notation

- Strings that are to be taken literally are enclosed in “double quotes”.
- Words that are emphasized are printed in *italic*.
- Keywords that are defined by the UPnP Working Committee are printed using the forum character style.
- Keywords that are defined by the UPnP Device Architecture are printed using the arch character style.

## SensorDataModel:1

- A double colon delimiter, "::", signifies a hierarchical parent-child (parent::child) relationship between the two objects separated by the double colon. This delimiter is used in multiple contexts, for example: Service::Action(), Action()::Argument, parentProperty::childProperty.

### 4.2 Data Types

This specification uses data type definitions from two different sources. The UPnP Device Architecture defined data types are used to define state variable and action argument data types [1]. The XML Schema namespace is used to define property data types [7].

For UPnP Device Architecture defined Boolean data types, it is strongly RECOMMENDED to use the value "0" for false, and the value "1" for true. The values "true", "yes", "false", or "no" MAY also be used but are NOT RECOMMENDED. The values "yes" and "no" are deprecated and MUST NOT be sent out by devices but MUST be accepted on input.

For XML Schema defined Boolean data types, it is strongly RECOMMENDED to use the value "0" for false, and the value "1" for true. The values "true", "yes", "false", or "no" MAY also be used but are NOT RECOMMENDED. The values "yes" and "no" are deprecated and MUST NOT be sent out by devices but MUST be accepted on input.

### 4.3 Vendor-defined Extensions

Whenever vendors create additional vendor-defined state variables, actions or properties, their assigned names and XML representation MUST follow the naming conventions and XML rules as specified in [1], Clause 2.5, "Description: Non-standard vendor extensions".

#### 4.3.1 Extended Backus-Naur Form

Extended Backus-Naur Form is used in this document for a formal syntax description of certain constructs. The usage here is according to the reference

[20]

[20] ].

##### 4.3.1.1 Typographic conventions for EBNF

Non-terminal symbols are unquoted sequences of characters from the set of English upper and lower case letters, the digits "0" through "9", and the hyphen ("-"). Character sequences between 'single quotes' are terminal strings and shall appear literally in valid strings. Character sequences between (\*comment delimiters\*) are English language definitions or supplementary explanations of their associated symbols. White space in the EBNF is used to separate elements of the EBNF, not to represent white space in valid strings. White space usage in valid strings is described explicitly in the EBNF. Finally, the EBNF uses the following operators:

**Table 1 — EBNF Operators**

<b>Operator</b>	<b>Semantics</b>
<b>::=</b>	<b>definition</b> – the non-terminal symbol on the left is defined by one or more alternative sequences of terminals and/or non-terminals to its right.
<b> </b>	<b>alternative separator</b> – separates sequences on the right that are independently allowed definitions for the non-terminal on the left.
<b>*</b>	<b>null repetition</b> – means the expression to its left may occur zero or more times.
<b>+</b>	<b>non-null repetition</b> – means the expression to its left shall occur at least once and may occur more times.
<b>[ ]</b>	<b>optional</b> – the expression between the brackets is allowed.
<b>( )</b>	<b>grouping</b> – groups the expressions between the parentheses.
<b>-</b>	<b>character range</b> – represents all characters between the left and right character operands inclusively.



## **SensorManagement ConfigurationManagement Service Profile**

### **4.4 Service Type**

This specification reuses the ConfigurationManagement service exactly as specified in ConfigurationManagement service [16]:

**urn:schemas-upnp-org:service:ConfigurationManagement:2**

SensorManagement Sensor DataModel service defined in this specification refers to the same service type.

### **4.5 SensorManagement Device Architecture**

The UPnP SensorManagement device [11] provides UPnP client access to Sensors and Actuators. These devices may be connected directly to the UPnP network or can be connected to non-UPnP based networks which are bridged by this device into the UPnP network. The SensorManagement device includes a UPnP ConfigurationManagement service [16] which describes Sensors to UPnP based clients.

#### **4.5.1 Sensor Description**

The UPnP SensorManagement Sensor DataModel service provides a set of uniform Sensor Properties as defined by Annex A, "SensorManagement General Data Model". These properties assist UPnP clients to identify sensors they may be capable of supporting. In addition to uniform Sensor properties described by the General Sensor Data Model, this specification also can reference additional sensor properties which are defined by the Sensor's parent ecosystem.

#### **4.5.2 Sensor Data Interface**

The UPnP SensorManagement device [11] also includes a UPnP SensorTransportGeneric service [12]. This service treats Sensors and Actuators as abstract data sources and sinks. While the SensorManagement device performs much of the interfacing tasks for Sensors, the actual data formats delivered by the Sensors are largely determined by the parent ecosystems for each target Sensor or Actuator. UPnP describes Sensor and Actuator data inputs/outputs as named Dataltem(s). Sensor Dataltem(s) are described by providing a name, format, encoding (see SensorManagement UPnP Sensor and DataStore Architecture Overview [10], subclause 4.3, "Dataltem Semantics" for additional information).

#### **4.5.3 Sensor Data Records**

A UPnP client may transfer data to/from a Sensor(s) supported by the SensorManagement device using actions in the included UPnP SensorTransportGeneric service [12]. When requesting data, the UPnP client specifies an XML document conforming to the XML Schema UPnP SensorManagement DataRecord Information [19]. This document identifies a series of one or more Dataltem(s) supported by the target Sensor. The UPnP client may either read or write the Sensor directly via SOAP actions or may establish a transport connection allowing the Sensor to deliver DataRecord(s) directly to a transport client.

#### **4.5.4 SensorURNs**

A SensorURN identifies a set of Dataltem(s) a Sensor supports. See subclause 4.5.6, "Sensor Normative Type Identifiers" for requirements on construction of SensorURN values.

#### **4.5.5 Sensor Required Dataltems**

Implementation of certain Dataltem(s) is required by the SensorManagement specification. These Dataltem(s) shall be available for all Sensor URNs whether or not explicitly defined by the corresponding SensorURN definition. The required Dataltem(s) are listed in Annex B "Required Sensor Dataltems".

#### 4.5.6 Sensor Normative Type Identifiers

UPnP SensorManagement defines a uniform template for generating normative identifiers. These identifiers include SensorCollectionType, SensorType and SensorURNs. The following template is used:

SensorIdentifier ::= "urn:" upnp-org ":" [identifier-type] ":" [generic-device-type] ":" [sensor-vendor-identifier] ":" [sensor-vendor-spec] ":" [vendor-identifier] : [vendor-model-identifier] ":" [identifier-type-dependent]

SensorIdentifier(s) using the [upnp-org] domain shall use the following [identifier-type] values:

**Table 0-1 — Sensor URN [identifier-type] values**

Sensor Identifier	[identifier-type] value
Sensor Collection Type	<u>smgt-sct</u>
Sensor Type	<u>smgt-st</u>
Sensor URN	<u>smgt-surn</u>

The [generic-device-identifiers] field is descriptive. UPnP SensorManagement defines device-identifiers for various common devices (see Annex C, "Common Device Identifiers"). If one of these values is suitable, then the UPnP defined value should be used. However, if a sensor application is unique, vendor-specific extensions are permitted.

The [sensor-vendor-identifier] shall contain the ICANN domain-name of the organization or vendor which defines the sensor specification. This name shall identify the vendor which defines the sensors for the device rather than the final integrator of the device.

The [sensor-vendor-spec] shall identify the specification the sensor conforms to.

The [vendor-identifier] shall contain the ICAN domain-name of the device vendor. This domain-name shall identify the vendor which the customer would normally identify as the manufacturer of the device.

The [vendor-model-identifier] shall provide vendor specific identifying information for the device.

The [identifier-type-dependent] field is defined based on the type of SensorIdentifier as follows:

**Table 0-2 — Sensor URN [identifier-type-dependent] values**

[identifier-type-dependent]	[identifier-types]	Description
<u>monitor</u>	<u>smgt-st, smgt-surn</u>	Dataltem(s) or Sensor(s) intended to periodically monitor or record device performance over a period of time.
<u>status</u>	<u>smgt-st, smgt-surn</u>	Dataltem(s) or Sensor(s) intended to provide current device status or settings.
<u>datafeed</u>	<u>smgt-st, smgt-surn</u>	Dataltem(s) or Sensor(s) which to provide a continuous source of sensor data.
<u>telemetry</u>	<u>smgt-st, smgt-surn</u>	Dataltem(s) or Sensor(s) intended to report data obtained by a Sensor.
<u>setting</u>	<u>smgt-st, smgt-surn</u>	Dataltem(s) or Sensor(s) intended to modify end-user settable characteristics of a device.
<u>control</u>	<u>smgt-st, smgt-surn</u>	Dataltem(s) or Sensor(s) intended to control aspects of a device not typically set by end-users
<u>alarms</u>	<u>smgt-st, smgt-surn</u>	Dataltem(s) or Sensor(s) intended to convey abnormal conditions for a device.

#### 4.5.7 Sensor Event Model

The SensorManagement event model generates two categories of events.

- Sensor Configuration events

Sensor Configuration events indicate addition or removal of SensorCollection(s) or Sensor(s) within a collection. Reporting of Sensor and SensorCollection configuration events are always enabled.

- Sensor Specific events

Sensor Specific events indicate state changes to Sensor(s) such as data availability or transport connection errors. Reporting of Sensor events may be enabled on a per-Sensor basis via the [SensorEventsEnable](#) parameter.

Sensor events are reported via the [SensorEvents](#) parameter which provides an XML document conforming to XML Schema UPnP SensorManagement Sensor DataModel Events [17]. This parameter shall support the [EventOnChange](#) attribute value which shall be enabled by default and the [AlarmOnChange](#) attribute value which shall also be enabled by default.

Changes to the [SensorEvents](#) parameter shall be sent to subscribed control points via the ConfigurationManagement service [ConfigurationUpdate](#) state variable. The SensorManagement ConfigurationManagement service shall implement and set the [AlarmsEnabled](#) state variable to ("[1](#)") by default.

To maintain privacy and security sensor collection(s) and sensor(s) are only identified by their [CollectionID](#) and [SensorID](#) values. In order to obtain additional information a UPnP control point would need sufficient privileges to read to the corresponding Data Model nodes which describe the sensor collection or sensor.

The SensorEvents parameter value shall be set to an XML document conforming to the UPnP SensorManagement Sensor DataModel Events schema [17] and as described below:

```
<?xml version="1.0" encoding="utf-8"?>
<SensorEvents
  xmlns="urn:schemas-upnp-org:smgt:sdmevent"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="urn:schemas-upnp-org:smgt:sdmevent
http://www.upnp.org/schemas/smgt/sdmevent-v1.xsd">
  <sensorevent
    collectionID="sensorCollectionID"
    sensorID="sensorID"
    event="sensor event or sensor collection event name "/>
    ...
  </SensorEvents>
```

<?xml>

Required. Case Sensitive

<SensorEvents>

Required. Shall include the namespace declaration for the SensorEvents XML Schema ("urn:schemas-upnp-org:smgt:sdmevent"). Shall include the zero or more of the following element and attributes:

<sensorevent>

Required. Shall occur zero or more times. Indicates that a sensor or sensor collection event has occurred.

collectionID

Required. Indicates the [CollectionID](#) for the reported event.

sensorID

Required. Indicates the [SensorID](#) for the reported event. If there is no corresponding [SensorID](#) for a reported event this attribute shall be set to "".

## SensorDataModel:1

event

Required. Indicates the type of sensor related event. The `event` attribute shall use one of the values as detailed in the following table.

**Table 0-3 — <SensorEvents> event= attribute allowed values**

| SensorEvent Name                                | Description  |
|---|--|
| <u><a href="#">SensorCollectionAdded</a></u>    | A new sensor collection has been added.<br>Reporting of this event is enabled by default.  |
| <u><a href="#">SensorCollectionRemoved</a></u>  | A previously reported sensor collection has been removed.<br>Reporting of this event is enabled by default.  |
| <u><a href="#">SensorAdded</a></u>              | A new sensor has been added.<br>Reporting of this event is enabled by default.   |
| <u><a href="#">SensorRemoved</a></u>            | A previously reported sensor has been removed.<br>Reporting of this event is enabled by default.   |
| <u><a href="#">SOAPDataAvailable</a></u>        | The sensor has data available to be read by the <u><a href="#">ReadSensor()</a></u> SOAP action.<br>Reporting of this event is disabled by default. Reporting of this event can be enabled by setting the <u><a href="#">SensorEventsControl</a></u> parameter to "SOAPDataAvailableEnable,1".           |
| <u><a href="#">SOAPDataOverrun</a></u>          | The sensor has discarded data which was not read by any SOAP client.<br>Reporting of this event is disabled by default. Reporting of this event can be enabled by setting the <u><a href="#">SensorEventControl</a></u> parameter to "SOAPDataOverrunEnable,1".  |
| <u><a href="#">TransportDataAvailable</a></u>   | The sensor has received data available to be sent to one or more transport connection for this sensor.<br>Reporting of this event is disabled by default. Reporting of this event can be enabled by setting the <u><a href="#">SensorEventControl</a></u> parameter to "TransportDataAvailableEnable,1". |
| <u><a href="#">TransportDataOverrun</a></u>     | The sensor has discarded data which was not read by any active transport connection.<br>Reporting of this event is disabled by default. Reporting of this event can be enabled by setting the <u><a href="#">SensorEventControl</a></u> parameter to "TransportDataOverrunEnable,1".                     |
| <u><a href="#">TransportConnectionError</a></u> | An error has occurred on one or more active transport connections for this sensor.<br>Reporting of this event is disabled by default. Reporting of this event can be enabled by setting the <u><a href="#">SensorEventControl</a></u> parameter to "TransportConnectionErrorEnable,1".                   |
| <u><a href="#">SensorAvailable</a></u>          | A currently configured sensor has become available.<br>Reporting of this event is disabled by default. Reporting of this event can be enabled by setting the <u><a href="#">SensorEventControl</a></u> parameter to "SensorAvailabilityEnable,1".  |
| <u><a href="#">SensorUnavailable</a></u>        | A currently configured sensor has become unavailable,<br>Reporting of this event is disabled by default. Reporting of this event can be enabled by setting the <u><a href="#">SensorEventControl</a></u> parameter to "SensorAvailabilityEnable,1".  |

## SensorDataModel:1

### 4.6 State Variables

The SensorManagement device [11] reuses some of the state variables defined in the ConfigurationManagement service [16].

The Table 0-4 below lists all the eventable state variables used in the SensorManagement device. These state variables are defined in the CMS. The table also indicates the required/allowed constraint for the SensorManagement device.

All the required state variables from the CMS shall be implemented by SensorManagement, as specified in [1]. It is up to the implementation to choose whether to support allowed state variables.

The AttributeValuesUpdate state variable should be supported if the CMS implementation support the SetAttributeValues() for changing attribute values (i.e. the values of EventOnChange and Version, when it is also supported).

The SensorManagement device supports the A\_ARG\_TYPE state variables defined in the CMS depending on the required/allowed constraint of the respective actions for the SensorManagement. The A\_ARG\_TYPE\_ state variables are not listed in this document, refer to the ConfigurationManagement service [16] for the details.

**Table 0-4 — State Variables for Eventing**

| Variable Name  | CMS<br>R/A <sup>a</sup> | SensorManagement<br>R/A <sup>a</sup> |
|--|-------------------------|--------------------------------------|
| <u>ConfigurationUpdate</u>   | <u>R</u>                | <u>R</u>                             |
| <u>CurrentConfigurationVersion</u>   | <u>R</u>                | <u>R</u>                             |
| <u>SupportedDataModelsUpdate</u>   | <u>R</u>                | <u>R</u>                             |
| <u>SupportedParametersUpdate</u>   | <u>R</u>                | <u>R</u>                             |
| <u>AttributeValuesUpdate</u>   | <u>A</u>                | <u>A</u>                             |
| <u>InconsistentStatus</u>  | <u>A</u>                | <u>A</u>                             |
| <u>AlarmsEnabled</u>   | <u>CR</u>               | <u>R</u>                             |
| <sup>a</sup> <u>R</u> = required, <u>A</u> = allowed, <u>CR</u> = conditionally required, <u>CA</u> = conditionally allowed, <u>X</u> = Non-standard, add <u>-</u> when deprecated (e.g., <u>R-D</u> , <u>A-D</u> ). |                         |                                      |

Note: For first-time reader, it may be more insightful to read the theory of operations first and then the action definitions before reading the state variable definitions.

### 4.7 Actions

#### 4.7.1 Introduction

The SensorManagement device reuses the actions defined in the CMS. Some of the actions which are allowed in CMS are required or conditionally required for SensorManagement.

Table 0-5 below lists all the CMS actions, with the following additional information:

- The column “CMS R/A” indicates whether the action is required or allowed in ConfigurationManagement service [16]; all the required actions from the CMS shall be implemented by the device supporting SensorManagement.
- The column “SensorManagement R/A” indicates whether the action is either required or allowed for the SensorManagement Sensor DataModel service [14]; some of the allowed actions are required for SensorManagement.

### SensorDataModel:1

- The column “SensorManagement CP R/A” indicates whether the SensorManagement CP shall support the action to be fully compliant with the SensorManagement device specifications.

Refer to ConfigurationManagement service [16] for more detailed information on each action. The remaining subclauses of 4.7 give additional information on some of the actions as used by the SensorManagement profile.

**Table 0-5 — Actions**

| Name   | CMS<br>R/A <sup>a</sup> | Sensor<br>Management<br>R/A <sup>b</sup> | Sensor<br>Management<br>CP<br>R/A <sup>c</sup> |
|--|-------------------------|--|--|
| <u>GetSupportedDataModels()</u>  | <u>R</u>                | <u>R</u>                                 | <u>R</u>                                       |
| <u>GetSupportedParameters()</u>  | <u>R</u>                | <u>R</u>                                 | <u>R</u>                                       |
| <u>GetInstances()</u>  | <u>R</u>                | <u>R</u>                                 | <u>R</u>                                       |
| <u>GetValues()</u>   | <u>R</u>                | <u>R</u>                                 | <u>R</u>                                       |
| <u>GetSelectedValues()</u>   | <u>A</u>                | <u>A</u>                                 | <u>A</u>                                       |
| <u>SetValues()</u>   | <u>A</u>                | <u>R</u>                                 | <u>A</u>                                       |
| <u>CreateInstance()</u>  | <u>A</u>                | <u>CR</u>                                | <u>A</u>                                       |
| <u>DeleteInstance()</u>  | <u>A</u>                | <u>CR</u>                                | <u>A</u>                                       |
| <u>GetAttributes()</u>   | <u>R</u>                | <u>R</u>                                 | <u>A</u>                                       |
| <u>SetAttributes()</u>   | <u>A</u>                | <u>A</u>                                 | <u>A</u>                                       |
| <u>GetInconsistentStatus()</u>   | <u>A</u>                | <u>A</u>                                 | <u>A</u>                                       |
| <u>GetConfigurationUpdate()</u>  | <u>R</u>                | <u>R</u>                                 | <u>A</u>                                       |
| <u>GetCurrentConfigurationVersion()</u>  | <u>R</u>                | <u>R</u>                                 | <u>A</u>                                       |
| <u>GetSupportedDataModelsUpdate()</u>  | <u>R</u>                | <u>R</u>                                 | <u>A</u>                                       |
| <u>GetSupportedParametersUpdate()</u>  | <u>R</u>                | <u>R</u>                                 | <u>A</u>                                       |
| <u>GetAttributeValuesUpdate()</u>  | <u>A</u>                | <u>A</u>                                 | <u>A</u>                                       |
| <p><sup>a</sup> For a device this column indicates whether the action shall be implemented or not, where <u>R</u> = required, <u>A</u> = allowed, <u>CR</u> = conditionally required, <u>CA</u> = conditionally allowed, <u>X</u> = Non-standard, add <u>-D</u> when deprecated (e.g., <u>R-D</u>, <u>A-D</u>).</p> <p><sup>b</sup> For a device this column indicates whether the action shall be implemented or not, where <u>R</u> = required, <u>A</u> = allowed, <u>CR</u> = conditionally required, <u>CA</u> = conditionally allowed, <u>X</u> = Non-standard, add <u>-D</u> when deprecated (e.g., <u>R-D</u>, <u>A-D</u>).</p> <p><sup>c</sup> For a control point this column indicates whether a control point shall be capable of invoking this action, where <u>R</u> = required, <u>A</u> = allowed, <u>CR</u> = conditionally required, <u>CA</u> = conditionally allowed, <u>X</u> = Non-standard, add <u>-D</u> when deprecated (e.g., <u>R-D</u>, <u>A-D</u>).</p> |                         |  |  |

#### **4.7.2 GetSupportedDataModels()**

The SensorManagement implementation shall provide an XML document compliant with the <SupportedDataModels> root element per the CMS namespace (urn:schemas-upnp-org:dm:cms). This element shall contain a <SubTree> child element which describing the SensorManagement Sensor DataModel as follows:

```
<?xml version="1.0" encoding="UTF-8"?>
<cms:SupportedDataModels
  xmlns:cms="urn:schemas-upnp-org:dm:cms"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="urn:schemas-upnp-org:dm:cms
    http://www.upnp.org/schemas/dm/cms.xsd">
  <SubTree>
    <URI>
      urn:upnp-org:smgt:1
    </URI>
  </SubTree>
</cms:SupportedDataModels>
```

## SensorDataModel:1

```
<Location>
  /UPnP/SensorMgt
</Location>
<URL>
  http://www.upnp.org/specs/smgt/UPnP-smgt-SensorDataModel-v1-Service.pdf
</URL>
<Description>
  ... device vendor descriptive text ...
</Description>
</SubTree>
</SupportedDataModels>
```

<?xml>

Required. Case Sensitive

<SupportedDataModels>

Required. Shall include the name space declaration for the CMS XML Schema ("urn:schemas-upnp-org:dm:cms").

<SubTree>

Required. Shall include a <SubTree> element with the following elements:

<URI>

Required. Implementations complying with this specification shall provide the value:

urn:upnp-org:smgt:1

<Location>

Required. Implementations complying with this specification shall provide the value:

/UPnP/SensorMgt

<URL>

Allowed. Implementations may use the following value default value corresponding to the <URI> value (above) or may provide a reference to device vendor supplied specifications:

http://www.upnp.org/specs/smgt/UPnP-smgt-SensorDataModel-v1-Service.pdf

<Description>

Allowed. Implementations may provide descriptive text for their SensorManagement implementation or may omit this element.

### **4.7.3 GetSelectedValues()**

The GetSelectedValues() action is an allowed action for the CMS, and also an allowed action in SensorManagement.

### **4.7.4 SetValues()**

The SetValues() action is an allowed action for the CMS, but from the SensorManagement perspective it is a required action and shall be implemented.

This action is required since the SensorManagement Sensor DataModel service supports Parameters with write access. This action allows a SensorManagement CP to change the values of such Parameters.

### **4.7.5 CreateInstance()**

The CreateInstance() action is an allowed action for the CMS, but the SensorManagement perspective it is a conditionally required action. This action shall be implemented if the SensorManagement device supports control point additions and/or deletions to a sensor's SensorGroups and/or SensorPermissions MultiInstance parameter nodes.

### **4.7.6 DeleteInstance()**

The DeleteInstance() action is an allowed action for the CMS, but the SensorManagement perspective it is a conditionally required action. This action shall be implemented if the

### SensorDataModel:1

SensorManagement device supports control point additions and/or deletions to a Sensor's SensorGroups and/or SensorPermissions MultilInstance parameter nodes.

#### **4.7.7    SetAttributes()**

The SetAttributes() action is an allowed action for the CMS, and is also an allowed action for SensorManagement.



## Annex A SensorManagement General Data Model (normative)

Table 0-6 — SensorManagement General Parameters

| Name   | Type           | Acc | Req | Reference | EOC | Ver |
|--|----------------|-----|-----|-----------|-----|-----|
| /UPnP/SensorMgt/   | SingleInstance | RO  | R   | A.1.1.1   |     | 0   |
| SensorEvents   | string         | RO  | R   | A.1.1.2   | 1   | 1   |
| SensorCollectionsNumberOfEntries   | int            | RO  | R   | A.1.1.3   |     | 0   |
|  |                |     |     |           |     |     |
| /UPnP/SensorMgt/SensorCollections/#!/  | MultiInstance  | RO  | R   | A.1.1.4   |     | 0   |
| CollectionID   | string         | RO  | R   | A.1.1.5   |     | 0   |
| CollectionType   | string         | RO  | R   | A.1.1.6   |     | 0   |
| CollectionFriendlyName   | string         | RW  | R   | A.1.1.7   |     | 0   |
| CollectionInformation  | string         | RW  | R   | A.1.1.8   |     | 0   |
| CollectionUniqueIdentifier   | string         | RO  | R   | A.1.1.9   |     | 0   |
| CollectionSpecific/  | SingleInstance | RO  | CR  | A.1.1.10  |     | 0   |
| SensorsNumberOfEntries   | int            | RO  | R   | A.1.1.11  |     | 0   |
|  |                |     |     |           |     |     |
| /UPnP/SensorMgt/SensorCollections/#!/Sensors/#!/                             | MultiInstance  | RO  | R   | A.1.1.12  |     | 0   |
| SensorID   | string         | RO  | R   | A.1.1.13  |     | 0   |
| SensorType   | string         | RO  | R   | A.1.1.14  |     | 0   |
| SensorUniqueIdentifier   | string         | RO  | A   | A.1.1.15  |     | 0   |
| SensorUpdateRequest  | boolean        | RW  | A   | A.1.1.15  |     | 0   |
| SensorPollingInterval  | Int            | RW  | A   | A.1.1.16  |     | 0   |
| SensorReportChangeOnly   | boolean        | RW  | A   | A.1.1.17  |     | 0   |
| SensorEventsEnable   | string         | RW  | A   | A.1.1.18  |     | 0   |
| SensorSpecific/  | SingleInstance | RO  | CR  | A.1.1.29  |     | 0   |
| SensorRelatedNumberOfEntries   | Int            | RO  | CA  | A.1.1.19  |     | 0   |
| SensorGroupsNumberOfEntries  | Int            | RO  | CA  | A.1.1.20  |     | 0   |
| SensorDefaultPermissionsNumberOfEntries                                      | Int            | RO  | CA  | A.1.1.21  |     | 0   |
| SensorURNsNumberOfEntries  | Int            | RO  | R   | A.1.1.30  |     | 0   |
|  |                |     |     |           |     |     |
| UPnP/SensorMgt/SensorCollections/#!/Sensors/#!/SensorRelated/#!/             | MultiInstance  | RO  | A   | A.1.1.22  |     | 0   |
| SensorPath   | string         | RO  | R   | A.1.1.23  |     | 0   |
|  |                |     |     |           |     |     |
| UPnP/SensorMgt/SensorCollections/#!/Sensors/#!/SensorGroups/#!/              | MultiInstance  | RO  | CR  | A.1.1.24  |     | 0   |
| SensorGroup  | string         | RW  | R   | A.1.1.25  |     | 0   |
|  |                |     |     |           |     |     |
| /UPnP/SensorMgt/SensorCollections/#!/Sensors/#!/SensorDefaultPermissions/#!/ | MultiInstance  | RW  | CA  | A.1.1.26  |     | 0   |
| SensorDefaultRole  | string         | RO  | R   | A.1.1.27  |     | 0   |
| SensorDefaultPermissions   | CSV(string)    | RO  | R   | A.1.1.28  |     | 0   |

## SensorDataModel:1

|   |              |    |   |          |  |   |
|---|--------------|----|---|----------|--|---|
|   |              |    |   |          |  |   |
| /UPnP/SensorMgt/SensorCollections/#/Sensors/#/SensorURNs/#/             | MultInstance | RO | R | A.1.1.31 |  | 0 |
| SensorURN   | string       | RO | R | A.1.1.32 |  | 0 |
| DataItemsNumberOfEntries  | Int          | RO | R | A.1.1.33 |  | 0 |
|   |              |    |   |          |  |   |
| /UPnP/SensorMgt/SensorCollections/#/Sensors/#/SensorURNs/#/DataItems/#/ | MultInstance | RO | R | A.1.1.34 |  | 0 |
| Name  | string       | RO | R | A.1.1.35 |  | 0 |
| Type  | string       | RO | R | A.1.1.36 |  | 0 |
| Encoding  | string       | RO | R | A.1.1.37 |  | 0 |
| Description   | string       | RO | R | A.1.1.38 |  | 0 |

## A.1 Introduction

This section describes SensorManagement Nodes and Parameters which are common to all sensor collections and sensors supported by the host SensorManagement device. Individual sensor collections appear as instances of the [/UPnP/SensorMgt/SensorCollections/](#) MultInstance node. Nodes which are sensor collection type specific are listed under the [/UPnP/SensorMgt/SensorCollections/#/CollectionSpecific/](#) node. Each [SensorCollection](#) node hosts zero or more [Sensors](#) nodes which appear as instances of the [/UPnP/SensorMgt/SensorCollections/#/Sensors/](#) MultInstance node. Nodes which are sensor type specific are listed under the [/UPnP/SensorMgt/SensorCollections/#/SensorSpecific/](#) node for each [Sensor](#) instance. Each [Sensor](#) supports zero or more [Sensor URNs](#) [/UPnP/SensorMgt/SensorCollections/#/Sensors/#/SensorURNs/#/SensorURNs](#) which identifies sets required and permitted DataItem(s) generated by each [Sensor](#).

### A.1.1 SensorManagement General Parameters

#### A.1.1.1 /UPnP/SensorMgt/

**Type:** SingleInstance

**Description:** The required [SensorMgt](#) SingleInstance node identifies the root node of the SensorManagement Sensor DataModel.

#### A.1.1.2 /UPnP/SensorMgt/SensorEvents

**Type:** string

**Description:** The required [SensorEvents](#) parameter shall provide an XML document conforming to the XML Schema UPnP SensorManagement Sensor DataModel Events [17]. This document shall contain zero or more <sensorEvent> elements where each element occurrence reports a unique event condition as indicated by the event attribute for the [SensorCollectionID](#) and [SensorID](#) identified by the collectionID and sensorID attributes. This parameter shall remain unchanged for a minimum event moderation period (200mS). If additional events have accumulated a new SensorEvents XML document shall replace the current document after any active event moderation period has expired.

#### A.1.1.3 /UPnP/SensorMgt/SensorCollectionsNumberOfEntries

**Type:** int

**Description:** The required [SensorCollectionsNumberOfEntries](#) parameter shall provide the number of [SensorCollections](#) MultInstance node entries.

#### A.1.1.4 /UPnP/SensorMgt/SensorCollections/

**Type:** MultInstance

**Description:** The required [SensorCollections](#) MultInstance node identifies zero or more collections containing sensors and/or actuators supported by this SensorManagement device.

## SensorDataModel:1

### **A.1.1.5 /UPnP/SensorMgt/SensorCollections/#!/CollectionID**

**Type:** string

**Description:** The required [CollectionID](#) parameter provides a unique identifier for the purpose of issuing SOAP actions or identifying the sensor collection associated with events reported by the SensorTransportGeneric service [12].

### **A.1.1.6 /UPnP/SensorMgt/SensorCollections/#!/CollectionType**

**Type:** string

**Description:** The required [CollectionType](#) parameter value describes the type of sensor for this [SensorCollections](#) node instance. See the subclause 4.5.5, "Sensor Normative Type Identifiers" for required and allowed values for this parameter.

### **A.1.1.7 /UPnP/SensorMgt/SensorCollections/#!/CollectionFriendlyName**

**Type:** string

**Description:** The required [CollectionFriendlyName](#) parameter provides a string identifying the sensor collection suitable for end-user display. The SensorManagement service is allowed to preset this parameter. The SensorManagement service shall allow this field to be written provided the requesting control point has sufficient privileges. The SensorManagement service should retain previously written values between reboots.

### **A.1.1.8 /UPnP/SensorMgt/SensorCollections/#!/CollectionInformation**

**Type:** string

**Description:** The required [CollectionInformation](#) parameter provides a string describing the sensor collection suitable for end-user display. The SensorManagement service is allowed to preset this parameter. The SensorManagement service shall permit this parameter to be written provided the requesting control point has sufficient privileges. The SensorManagement service should retain previously written values between reboots.

### **A.1.1.9 /UPnP/SensorMgt/SensorCollections/#!/CollectionUniqueIdentifier**

**Type:** string

**Description:** The required [CollectionUniqueIdentifier](#) parameter provides a string which uniquely identifies a sensor collection. For devices discoverable on a UPnP network, this value shall match the UDN of the corresponding UPnP Device. See sections describing specific classes of sensor collection(s) for other allowed values.

### **A.1.1.10 /UPnP/SensorMgt/SensorCollections/#!/CollectionSpecific/**

**Type:** SingleInstance

**Description:** The conditionally required [CollectionSpecific](#) SingleInstance node provides ecosystem specific sensor collection information. The child nodes of the [CollectionSpecific](#) node depend on the [CollectionType](#) parameter value. See the sections describing specific classes of sensor collection(s) for required and allowed nodes for specific sensor collection types.

### **A.1.1.11 /UPnP/SensorMgt/SensorCollections/SensorsNumberOfEntries**

**Type:** int

**Description:** The required [SensorsNumberOfEntries](#) parameter shall provide the number of [Sensor](#) MultiInstance node entries.

### **A.1.1.12 /UPnP/SensorMgt/SensorCollections/#!/Sensors/**

**Type:** MultiInstance

**Description:** The required [Sensors](#) MultiInstance node shall describe a zero or more sensor(s) or actuator(s) managed by this service.

### **A.1.1.13 /UPnP/SensorMgt/SensorCollections/#!/Sensors/#!/SensorID**

**Type:** string

### SensorDataModel:1

**Description:** The required SensorID parameter provides a unique identifier for the purpose of issuing SOAP actions to the sensor. See the A\_ARG\_TYPE\_SensorID in the SensorTransportGeneric service specification [12] for further usage information about this parameter. The SensorManagement device is permitted to change assigned SensorID values between service restarts.

#### **A.1.1.14 /UPnP/SensorMgt/SensorCollections/#!/Sensors/#!/SensorType**

**Type:** string

**Description:** The required SensorType parameter value describes the type of sensor for this Sensors node instance. See the subclause 4.5.5, "Sensor Normative Type Identifiers" for required and allowed values for this parameter.

#### **A.1.1.15 /UPnP/SensorMgt/SensorCollections/#!/Sensors/#!/SensorUpdateRequest**

**Type:** boolean

**Description:** The allowed SensorUpdateRequest parameter when written to ("1") shall cause the SensorManagement Sensor DataModel service to request an update from the sensor associated with this Sensors instance. When the SensorManagement Sensor DataModel service completes the update process, it shall reset this parameter to ("0"). See the sections describing specific classes of sensor collection(s) to determine if implementation of this parameter is required.

#### **A.1.1.16 /UPnP/SensorMgt/SensorCollections/#!/Sensors/#!/SensorPollingInterval**

**Type:** int

**Description:** The allowed SensorPollingInterval parameter when written to a non-zero value (milliseconds) shall cause SensorManagement service to request periodic updates from the sensor associated with this Sensors instance.

#### **A.1.1.17 /UPnP/SensorMgt/SensorCollections/#!/Sensors/#!/SensorReportChangeOnly**

**Type:** int

**Description:** This allowed SensorReportChangeOnly parameter when written to a ("1") in addition to a non-zero SensorPollingInterval parameter shall cause the SensorManagement service to suppress reporting of identical sensor readings. The setting of this property shall not suppress sensor updates requested by setting the SensorUpdateRequest parameter.

#### **A.1.1.18 /UPnP/SensorMgt/SensorCollections/#!/Sensors/#!/SensorEventsEnable**

**Type:** string

**Description:** The required SensorsEventsEnable parameter shall enable reporting of events for the indicated sensor. On read of this parameter a CSV of pairs of supported event names and a boolean value with ("1") indicating that the corresponding event is to be reported. On write a CSV of an event name and boolean value indicating the new enable/disable setting for a supported named event shall be accepted. Sensor events are reported by the SensorEvents XML document provided by the UPnP/SensorMgt/SensorEvents parameter.

The following SensorEventsEnable names are defined:

**Table A.1 — SensorEventEnable parameter allowed values**

| SensorEventEnable Name              | Description  |
|-------------------------------------|--|
| <u>SOAPDataAvailableEnable</u>      | Controls reporting of SOAPDataAvailable events indicating sensor data is available for reading by the ReadSensor() action of the SensorTransportGeneric service. |
| <u>SOAPDataOverrunEnable</u>        | Controls reporting of SOAPDataOverrun events indicating the sensor has discarded data which was not read by any SOAP client.                                     |
| <u>TransportDataAvailableEnable</u> | Controls reporting of TransportDataAvailable events indicating a sensor with one or more active transport connections has received data.                         |

## SensorDataModel:1

|   |  |
|---|--|
| <u><a href="#">TransportDataOverrunEnable</a></u>     | Controls reporting of TransportDataOverrun events indicating a sensor with one or more active transport connections has discarded received data it was unable to forward to transport clients. |
| <u><a href="#">TransportConnectionErrorEnable</a></u> | Controls reporting of TransportConnectionError events indicating a sensor with one or more active transport connections has detected a connection error.                                       |
| <u><a href="#">SensorAvailabilityEnable</a></u>       | Controls reporting of SensorAvailable and SensorUnavailable events indicating a previously reported sensor has become accessible or inaccessible.  |

The following example shows a sample SensorEventsEnable parameter and changing one of the reported event settings:

### **Request :**

```
GetValues(  
  "<?xml version='1.0' encoding='UTF-8'?>  
    <ContentPathList xmlns='urn:schemas-upnp-org:dm:cms'  
      xmlns:xsi='http://www.w3.org/2001/XMLSchema-instance'  
      xsi:schemaLocation='urn:schemas-upnp-org:dm:cms  
        http://www.upnp.org/schemas/dm/cms.xsd'>  
      <ContentPath>  
        /UPnP/SensorMgt/SensorCollections/1/Sensors/1/SensorEventsEnable  
      </ContentPath>  
    </ContentPathList>" )
```

### **Response :**

```
GetValues(  
  "<?xml version='1.0' encoding='UTF-8'?>  
    <ParameterValueList xmlns='urn:schemas-upnp-org:dm:cms'  
      xmlns:xsi='http://www.w3.org/2001/XMLSchema-instance'  
      xsi:schemaLocation='urn:schemas-upnp-org:dm:cms  
        http://www.upnp.org/schemas/dm/cms.xsd'>  
      <Parameter>  
        <ParameterPath>  
          /UPnP/SensorMgt/SensorCollections/1/Sensors/1/SensorEventsEnable  
        </ParameterPath>  
        <Value>  
          SOAPDataAvailableEnable,0,SOAPDataOverrunEnable,0,  
          TransportDataAvailableEnable,0,TransportDataOverrunEnable,0,  
          TransportConnectionErrorEnable,0  
        </Value>  
      </Parameter>  
    </ParameterValueList>" )
```

A SOAP client which requests notification of SensorDataAvailable events updates this parameter as follows:

### **Request :**

```
SetValues(  
  "<?xml version='1.0' encoding='UTF-8'?>  
    <ParameterValueList xmlns='urn:schemas-upnp-org:dm:cms'  
      xmlns:xsi='http://www.w3.org/2001/XMLSchema-instance'  
      xsi:schemaLocation='urn:schemas-upnp-org:dm:cms  
        http://www.upnp.org/schemas/dm/cms.xsd'>  
      <Parameter>  
        <ParameterPath>  
          /UPnP/SensorMgt/SensorCollections/1/Sensors/1/SensorEventsEnable  
        </ParameterPath>  
        <Value>  
          SOAPDataAvailableEnable,1  
        </Value>  
      </Parameter>  
    </ParameterValueList>" )
```

### **Response :**

## SensorDataModel:1

```
SetValues(  
    "ChangesCommitted")
```

### **A.1.1.19 /UPnP/SensorMgt/SensorCollections/#!/Sensors/#!/SensorsRelatedNumberOfEntries**

**Type:** int

**Description:** The conditionally required SensorsRelatedNumberOfEntries parameter shall provide the number of SensorsRelated MultInstance node entries. This parameter shall be implemented if the SensorsRelated MultInstance node is supported.

### **A.1.1.20 /UPnP/SensorMgt/SensorCollections/#!/Sensors/#!/SensorGroupsNumberOfEntries**

**Type:** int

**Description:** The conditionally required SensorsGroupsOfEntries parameter shall provide the number of SensorGroups MultInstance node entries. This parameter shall be implemented if the SensorGroups MultInstance node is supported.

### **A.1.1.21 /UPnP/SensorMgt/SensorCollections/#!/Sensors/#!/SensorDefaultPermissionsNumberOfEntries**

**Type:** int

**Description:** The conditionally required SensorsPermissionNumberOfEntries parameter shall provide the number of SensorPermissions MultInstance node entries. . This parameter shall be implemented if the SensorDefaultPermissions MultInstance node is supported

### **A.1.1.22 /UPnP/SensorMgt/SensorCollections/#!/Sensors/#!/SensorsRelated/**

**Type:** MultInstance

**Description:** The allowed SensorsRelated MultInstance node includes zero or more SensorPath parameter nodes identifying other sensors which have an operational relationship to this sensor. Note: This differs from the SensorGroups node where sensors are grouped for classification and device protection.

### **A.1.1.23 /UPnP/SensorMgt/SensorCollections/#!/Sensors/#!/SensorsRelated/#!/SensorPath**

**Type:** string

**Description:** The required SensorPath parameter provides a relative path to a sensor operationally related to this sensor. This path shall start at the SensorCollections MultInstance node and shall end at the Sensors/#!/ instance node identifying the sensor being referenced.

### **A.1.1.24 /UPnP/SensorMgt/SensorCollections/#!/Sensors/#!/SensorGroups/**

**Type:** MultInstance

**Description:** The conditionally required SensorGroups MultInstance node includes zero or more SensorGroup parameter nodes identifying SensorGroup(s) this sensor is participating in. The SensorGroups MultInstance node shall be implemented when the DeviceProtection feature is supported.

Note: See Sensor and DataStore Architecture Overview [10], subclause 4.1.9 "Sensor Protection Model" for further details on usage of SensorGroup(s) for DeviceProtection.

### **A.1.1.25 /UPnP/SensorMgt/SensorCollections/#!/Sensors/#!/SensorGroups/#!/SensorGroup**

**Type:** string

**Description:** The required SensorGroup parameter identifies an individual SensorGroup the corresponding Sensor is participating in. An instance of the SensorGroup node shall be included for each sensor group the sensor is a member of.

## SensorDataModel:1

**A.1.1.26 /UPnP/SensorMgt/SensorCollections/#!/Sensors/#!/SensorDefaultPermissions/**  
**Type:** MultilInstance

**Description:** The conditionally allowed [SensorPermissions](#) MultilInstance node lists sensor permissions for default DeviceProtection roles. This node may be implemented if the SensorManagement device provides DeviceProtection support and is otherwise prohibited.

**A.1.1.27 /UPnP/SensorMgt/SensorCollections/#!/Sensors/#!/SensorPermissions/#!/SensorDefaultRole**

**Type:** string

**Description:** The required [SensorDefaultRole](#) parameter identifies a default DeviceProtection role. The allowed values for this parameter are the default DeviceProtection roles: "[Public](#)" or "[Basic](#)".

**A.1.1.28 /UPnP/SensorMgt/SensorCollections/#!/Sensors/#!/SensorPermissions/#!/SensorDefaultPermissions**

**Type:** CSV(string)

**Description:** The required [SensorDefaultPermissions](#) parameter lists SensorPermissions assigned to the corresponding default DeviceProtection role identified by the [SensorDefaultRole](#) parameter. This parameter shall be supported if the SensorManagement device provides DeviceProtection support. The following allowed values are defined.

**Table A.7 — Sensor Permissions**

| Permission                         | Description   |
|------------------------------------|---|
| <a href="#">smgt:ReadSensor</a>    | A control point is permitted to issue <a href="#">ReadSensor()</a> actions to the corresponding Sensor.   |
| <a href="#">smgt:WriteSensor</a>   | A control point is permitted to issue <a href="#">WriteSensor()</a> actions to the corresponding Sensor.  |
| <a href="#">smgt:ConnectSensor</a> | A control point is permitted to issue <a href="#">ConnectSensor()</a> and <a href="#">DisconnectSensor()</a> actions to the corresponding Sensor. |
| <a href="#">smgt:CommandSensor</a> | A control point is permitted to modify SensorManagement properties in the DataModel.  |
| <a href="#">smgt:ViewSensor</a>    | A control point is permitted to view SensorManagement properties for this Sensor.   |

**A.1.1.29 /UPnP/SensorMgt/SensorCollections/#!/Sensors/#!/SensorSpecific/**

**Type:** SingleInstance

**Description:** The required [SensorSpecific](#) SingleInstance node provides ecosystem specific Sensor information. The allowed and required children of this node depend on the specific type of sensor as indicated by the [SensorType](#) parameter. Values prefixed by "smgt:" are defined within this specification. See the sections describing specific classes of sensor collection(s) for required and allowed nodes for specific sensor types.

**A.1.1.30 /UPnP/SensorMgt/SensorCollections/#!/Sensors/#!/SensorsURNsNumberOfEntries**

**Type:** int

**Description:** The required [SensorURNsNumberOfEntries](#) parameter shall provide the number of [SensorURNs](#) MultilInstance node entries.

**A.1.1.31 /UPnP/SensorMgt/SensorCollections/#!/Sensors/#!/SensorURNs/**

**Type:** MultilInstance

**Description:** The required [SensorURNs](#) MultilInstance node lists URN values for sets of DataItem(s) this sensor provides (or actuator accepts).

**A.1.1.32 /UPnP/SensorMgt/SensorCollections/#!/Sensors/#!/SensorURNs/#!/SensorURN**

**Type:** string



### SensorDataModel:1

**Description:** The required SensorURN parameter node identifies the URN value for each set of sensor Dataltem(s). See the sections describing specific classes of sensor collection(s) for required and allowed URN values for sensor types.

#### **A.1.1.33 /UPnP/SensorMgt/SensorCollections/#!/Sensors/#!/SensorURNs/DataltemsNumberOfEntries**

**Type:** int

**Description:** The required DataltemsNumberOfEntries parameter shall provide the number of Dataltems MultilInstance node entries.

#### **A.1.1.34 /UPnP/SensorMgt/SensorCollections/#!/Sensors/#!/SensorURNs/#!/Dataltems/**

**Type:** MultilInstance

**Description:** The required Dataltems MultilInstance node identifies sets of Dataltem(s) supported for the corresponding SensorURN parameter value.

#### **A.1.1.35 /UPnP/SensorMgt/SensorCollections/#!/Sensors/#!/SensorURNs/#!/Dataltems/#!/Name**

**Type:** MultilInstance

**Description:** The required Name parameter identifies the name of an individual Dataltem. See SensorManagement Architecture Overview [10] subclause 4.3, "Dataltem semantics" for further information.

#### **A.1.1.36 /UPnP/SensorMgt/SensorCollections/#!/Sensors/#!/SensorURNs/#!/Dataltems/#!/Type**

**Type:** MultilInstance

**Description:** The required Type parameter identifies the type of an individual Dataltem described by the corresponding SensorURN parameter value. See SensorManagement Architecture Overview [10] subclause 4.3, "Dataltem semantics" for further information.

#### **A.1.1.37 /UPnP/SensorMgt/SensorCollections/#!/Sensors/#!/SensorURNs/#!/Dataltems/#!/Encoding**

**Type:** MultilInstance

**Description:** The required Encoding parameter identifies the encoding of an individual Dataltem described by the corresponding SensorURN parameter value. See SensorManagement Architecture Overview [10] subclause 4.3, "Dataltem semantics" for further information.

#### **A.1.1.38 /UPnP/SensorMgt/SensorCollections/#!/Sensors/#!/SensorURNs/#!/Dataltems/#!/Description**

**Type:** MultilInstance

**Description:** The required Description parameter shall provide a string containing an XML document which describes the named Dataltem or an empty string if no Dataltem Description document is available. This XML document shall comply with the Dataltem Description schema. See SensorManagement Architecture Overview [10] subclause 4.3, "Dataltem semantics" for further information.



## Annex B Required Sensor Dataltem(s) (normative)

### B.1 Introduction

The following Sensor Dataltems shall be supported for all SensorURNs.

**Table B.1 — SensorManagement required Dataltem(s)**

| Name                    | Type                | Encoding      | Description  |
|-------------------------|---------------------|---------------|--|
| <u>ClientID</u>         | <u>xsd:string</u>   | <u>string</u> | <p>The required <u>ClientID</u> Dataltem shall match the value of the <u>SensorClientID</u> argument of the SensorTransportGeneric service <u>ConnectSensor()</u> or <u>ReadSensor()</u> actions.</p> <p>This Dataltem can be used by sensor transport clients to label the transport connection (or SOAP action) which is requesting the sensor data.</p> |
| <u>ReceiveTimestamp</u> | <u>xsd:dateTime</u> | <u>string</u> | <p>The required <u>ReceiveTimestamp</u> Dataltem shall provide a date-time stamp indicating when the sensor data was received by the SensorManagement device</p> <p>For example both of these timestamps represent the same time:</p> <p>2013-02-01T20:00:01Z<br/>2013-02-01T12:00:00-08:00</p>  |

## Annex C Common Device Identifiers (normative)

### C.1 Introduction

The following table lists common device identifiers. If one of these device identifiers apply to the SensorManagement application, then this value should be used in SensorManagement URN [device-identifier] fields.

Table C.1-1 — Common Device Identifiers

| <u>id</u> | <u>Category and Device</u>     | <u>Comment</u>  | <u>subid</u> | <u>Subclassification</u><br><u>(incl. UPnP classification)</u> |
|-----------|--------------------------------|---|--------------|--|
|           | <b>Space Conditioning (11)</b> |   |              |  |
| 1         | Unitary System                 | window and wall units without external components; self-contained | 1S1          | <a href="#"><u>Climate</u></a>                                 |
| 2         | Boiler                         | device that heats water (or other liquid).                        | 2S1          | <a href="#"><u>HotWater Heater</u></a>                         |
| 3         | Furnace                        | device that heats air (or other gas)                              |              |  |
| 4         | Pump                           | device that moves water (or other liquid)                         |              |  |
| 5         | Fan                            | device that moves air (or other gas)                              |              |  |
| 6         | Condensing Unit                | always includes a compressor                                      |              |  |
| 7         | Condensor                      | no compressor; just fan   |              |  |
| 8         | Humidifier                     | adds moisture to air  |              |  |
| 9         | Dehumidifier                   | removes moisture from air   |              |  |
| 10        | HVAC - control                 | directs operation of other HVAC devices; includes HVAC sensors    | 10S1         | <a href="#"><u>Controller</u></a>                              |
|           |                                |   | 10S2         | <a href="#"><u>HVAC</u></a>                                    |
|           |                                |   | 10S3         | <a href="#"><u>Thermostat</u></a>                              |
| 11        | HVAC - other                   | not readily classified into any of the above                      | 11S1         | <a href="#"><u>HVAC</u></a>                                    |
|           |                                |   | 11S2         | <a href="#"><u>RoomAirConditioner</u></a>                      |
|           |                                |   |              |  |
|           | <b>Lighting (5)</b>            |   |              |  |
| 12        | Lighting - outdoor             |   |              |  |
| 13        | Lighting - fixed               | hardwired   | 13S1         | <a href="#"><u>Light</u></a>                                   |
| 14        | Lighting - portable            | has normal plug   |              |  |
| 15        | Lighting - controls            | incl. sensors for lighting  |              |  |
| 16        | Lighting - other               | not readily classified into any of the above                      |              |  |
|           |                                |   |              |  |
|           | <b>Appliance (10)</b>          | large devices; small in misc.                                     |              |  |

SensorDataModel:1

|    |                         |   |      |                        |
|----|-------------------------|---|------|------------------------|
| 17 | Clothes dryer           |   | 17S1 | <u>Dryer</u>           |
| 18 | Clothes washer          | incl. combined washer/dryer                                 | 18S1 | <u>Washing machine</u> |
| 19 | Dishwasher              |   |      |                        |
| 20 | Freezer                 |   |      |                        |
| 21 | Ice machine             |   |      |                        |
| 22 | Oven                    | incl. warming drawer  |      |                        |
| 23 | Range                   | incl. cooktop and combined cooktop/oven                     |      |                        |
| 24 | Refrigerator            | including wine coolers, fridge/freezer combo                |      |                        |
| 25 | Water heater            |   |      |                        |
| 26 | Appliance - other       | not readily classified into any of the above                | 26S1 | <u>Microwave</u>       |
|    |                         |   |      |                        |
|    | <b>Electronics (21)</b> |   |      |                        |
| 27 | Audio system            | integrated source and speaker; incl. radio, boombox         |      |                        |
| 28 | Audio/video player      | e.g. CD, DVD, VCR, cassette, turntable                      | 28S1 | <u>MediaRenderer</u>   |
| 29 | Camera                  |   |      |                        |
| 30 | Computer, desktop       | incl. integrated  |      |                        |
| 31 | Computer, notebook      |   |      |                        |
| 32 | Computer, server        |   |      |                        |
| 33 | Computer, other         |   |      |                        |
| 34 | Data storage            |   |      |                        |
| 35 | Display                 | incl. monitor, projectors, TVs, digital picture frame       |      |                        |
| 36 | Electronics - portable  | can be operated by battery and not otherwise classified     |      |                        |
| 37 | Game console            |   |      |                        |
| 38 | Imaging equipment       | fax, multi-function device, scanner, printer, label printer |      |                        |
| 39 | Musical instrument      | also incl. recording devices, mixers, amplifiers            |      |                        |
| 40 | Network equipment       | modems, switches, routers, access points, etc.              |      |                        |
| 41 | Phone handset           | incl. tablet  |      |                        |
| 42 | Receiver                | incl. amplifier, home theatre system                        |      |                        |
| 43 | Set-top box             | cable, satellite, Internet                                  |      |                        |

SensorDataModel:1

|    |                           |  |      |                                      |
|----|---------------------------|--|------|--------------------------------------|
| 44 | Telephony                 | base stations, answering machines, corded phones           |      |                                      |
| 45 | Television                |  |      |                                      |
| 46 | Audio/video - other       |  |      |                                      |
| 47 | Electronics - other       | not readily classified into any of the above, incl. clocks | 47S1 | <a href="#"><u>MediaServer</u></a>   |
|    |                           |  |      |                                      |
|    | <b>Miscellaneous (29)</b> |  |      |                                      |
| 48 | Agriculture               | incl. irrigation timers                                    | 48S1 | <a href="#"><u>LawnSprinkler</u></a> |
| 49 | Air compressors           |  |      |                                      |
| 50 | Bathroom device           | toilets, faucets, hand dryers, etc.                        |      |                                      |
| 51 | Battery charger           |  |      |                                      |
| 52 | Business equipment        | money or office paper related                              |      |                                      |
| 53 | Cleaning equipment        | incl. vacuum   |      |                                      |
| 54 | Cooking - portable        |  |      |                                      |
| 55 | Decorations/hobby/leisure | fountains, aquaria, kilns, baby rockers, massage chairs    |      |                                      |
| 56 | Entertainment             | gambling, arcade, etc.                                     |      |                                      |
| 57 | Exercise machine          |  |      |                                      |
| 58 | Food service/prep         | ice cream machine, fryer, steam cooker                     |      |                                      |
| 59 | Household                 | non-food; incld iron, sewing machine, ...                  |      |                                      |
| 60 | HVAC - portable           | humidifiers, portable fans, heaters                        |      |                                      |
| 61 | Industrial                | process equipment and related                              |      |                                      |
| 62 | Kitchen                   | small devices  |      |                                      |
| 63 | Laboratory equipment      |  |      |                                      |
| 64 | Lighting - decorative     |  | 64S1 | <a href="#"><u>Light</u></a>         |
| 65 | Lighting - emergency      | incl. exit lights  |      |                                      |
| 66 | Medical equipment         | for professional settings                                  |      |                                      |
| 67 | Microwave oven            |  |      |                                      |
| 68 | Outdoor appliance         | mowers, trimmers, snow melting coils, grill                |      |                                      |
| 69 | Personal - health         | hair dryers, curlers, shavers, trimmers, toothbrush        |      |                                      |
| 70 | Pool - spa                |  |      |                                      |
| 71 | Sanitation                |  | 70S1 | <a href="#"><u>Disposal</u></a>      |

SensorDataModel:1

|    |                            |  |       |  |
|----|----------------------------|--|-------|--|
|    |                            | garbage disposal, trash compactor, wastewater pump, water filter             | 71S2  | <a href="#"><u>TrashCompactor</u></a>        |
| 72 | Sign                       | not incl. electronic displays  |       |  |
| 73 | Tool - construction        |  |       |  |
| 74 | Tool - non-construction    | e.g. auto, beauty  |       |  |
| 75 | Vending machine            |  |       |  |
| 76 | Water dispenser            |  |       |  |
|    |                            |  |       |  |
|    | <b>Infrastructure (11)</b> | Devices not used directly by people  |       |  |
| 77 | Breakers                   | incl. AFI/GFCI   |       |  |
| 78 | Doors / Windows            | incl. garage doors, gates, window shades                                     | 78S1  | <a href="#"><u>Door Lock</u></a>             |
|    |                            |  | 78S2  | <a href="#"><u>Door Sensor</u></a>           |
|    |                            |  | 78S3  | <a href="#"><u>Window Sensor</u></a>         |
|    |                            |  | 78S4  | <a href="#"><u>Keypad</u></a>                |
|    |                            |  | 78S5  | <a href="#"><u>Keyfob</u></a>                |
|    |                            |  | 78S6  | <a href="#"><u>GarageDoorOpener</u></a>      |
|    |                            |  | 78S7  | <a href="#"><u>WindowCovering</u></a>        |
| 79 | Fireplace                  |  |       |  |
| 80 | Motor                      | actuators  | 80S1  | <a href="#"><u>Actuator</u></a>              |
| 81 | Power - portable           | power strips, surge protectors, UPS, timer                                   |       |  |
| 82 | Power - fixed              | transformers, switchgears, inverters, voltage regulators, power conditioners |       |  |
| 83 | Pump                       |  | 83S1  | <a href="#"><u>HeatPump</u></a>              |
|    |                            |  | 83S2  | <a href="#"><u>PoolPump</u></a>              |
| 84 | Security                   | security cameras, systems  | 84S1  | <a href="#"><u>Surveillance</u></a>          |
|    |                            |  | 84S2  | <a href="#"><u>Alarm Panel</u></a>           |
|    |                            |  | 84S3  | <a href="#"><u>Panic Pendant</u></a>         |
| 85 | Sensors                    | incl. detectors: fire, smoke, gas, fluids                                    | 85S1  | <a href="#"><u>Sensor</u></a>                |
|    |                            |  | 85S2  | <a href="#"><u>Motion Sensor</u></a>         |
|    |                            |  | 85S3  | <a href="#"><u>Occupancy Sensor</u></a>      |
|    |                            |  | 85S4  | <a href="#"><u>Temperature Sensor</u></a>    |
|    |                            |  | 85S5  | <a href="#"><u>Glass break Sensor</u></a>    |
|    |                            |  | 85S6  | <a href="#"><u>CO Sensor</u></a>             |
|    |                            |  | 85S7  | <a href="#"><u>Heat Sensor</u></a>           |
|    |                            |  | 85S8  | <a href="#"><u>Smoke Sensor</u></a>          |
|    |                            |  | 85S9  | <a href="#"><u>Flammable Gas Sensor</u></a>  |
|    |                            |  | 85S10 | <a href="#"><u>Humidity Sensor</u></a>       |
|    |                            |  | 85S11 | <a href="#"><u>Wind Speed Sensor</u></a>     |
|    |                            |  | 85S12 | <a href="#"><u>Wind Direction Sensor</u></a> |

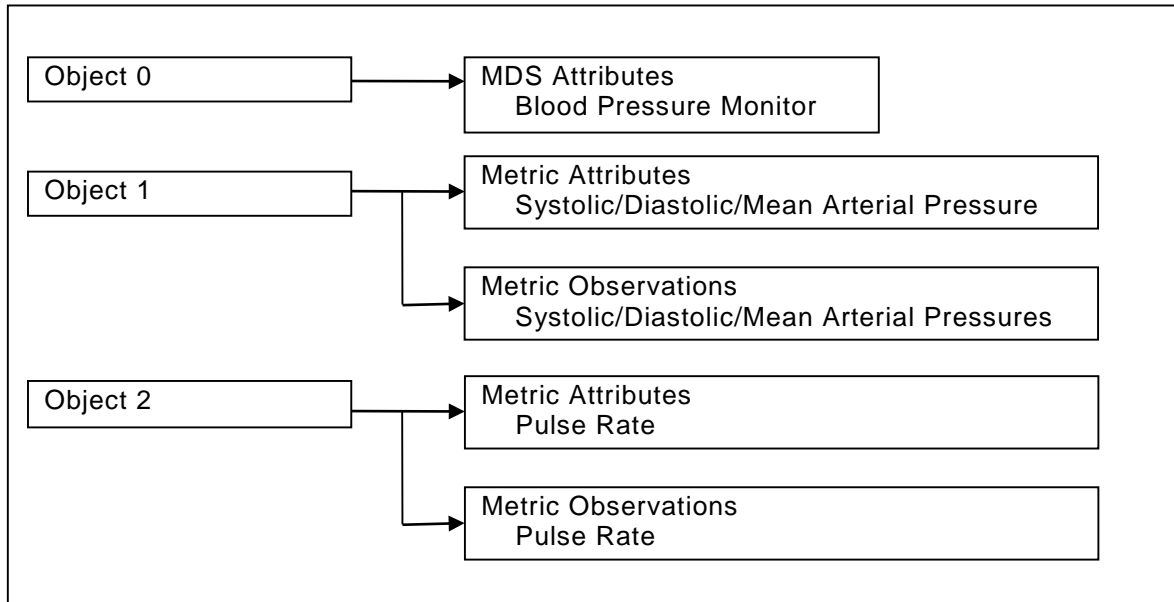
SensorDataModel:1

|    |                           |   |       |                                   |
|----|---------------------------|---|-------|-----------------------------------|
|    |                           |   | 85S13 | <u>Rain Sensor</u>                |
|    |                           |   | 85S14 | <u>Dewpoint Sensor</u>            |
|    |                           |   | 85S15 | <u>Barometric Pressure Sensor</u> |
|    |                           |   | 85S16 | <u>Stress Sensor</u>              |
|    |                           |   | 85S17 | <u>Flux Sensor</u>                |
|    |                           |   | 85S18 | <u>Weight Sensor</u>              |
| 86 | Signage                   |   |       |                                   |
| 87 | Meter                     | utility and other   |       |                                   |
|    |                           |   |       |                                   |
|    | <b>Transportation (4)</b> |   |       |                                   |
| 88 | Transport, fixed          | elevator, escalator, lifts, etc.  |       |                                   |
| 89 | Vehicle-large             |   |       |                                   |
| 90 | Vehicle-small             | incl. wheelchair, golf cart   |       |                                   |
| 91 | Transport, other          | incl. auto engine heater, vehicle charger, parking meter, parking equipment |       |                                   |
|    |                           |   |       |                                   |
|    | <b>Other (1)</b>          |   |       |                                   |
| 92 | Other                     | truly unclassifiable  | 92S1  | <u>Safety</u>                     |

"Table C.1- was prepared at Lawrence Berkeley National Laboratory under Contract No. DE-AC02-05CH11231 with the U.S. Department of Energy."

## Annex D SensorManagement IEEE-11073 Personal Health Devices

### D.1 SensorManagement for IEEE-11073 Personal Health Medical Device Systems



**Figure D.1** — Blood Pressure Monitor – Medical Device System (Informative)

This annex describes UPnP [SensorManagement](#) device requirements for support of IEEE-11073 Personal Health Medical Device Systems (agents). A UPnP SensorManagement device supporting IEEE-11073 Personal Health Devices encapsulates the functionality provided by an IEEE-11073 Manager which in turn implements IEEE-11073 communication protocols per IEEE-11073 Part 20601: Application Profile - Optimized Exchange [8]. IEEE-11073 Medical Device Systems are described as a set of objects. At a top level, Object 0 represents the properties of the medical device itself including identifying information for the device. Objects 1-n describe various measurement functions the medical device can perform. IEEE-11073 protocols separate data description and data transmission, therefore received measurement data (observations) only contain the minimal data relating to the quantities being measured. The attributes of the corresponding measurement object provide context such as measurement type, units of measure and observation data formats. Depending on the medical device being described measurement data may contain both time information as well as multiple measured quantities. Therefore, it is important that both measurement data and measurement object attributes are provided to UPnP clients processing data from IEEE-11073 agent devices. See the IEEE-11073-104zz Device specialization [9] series for detailed information on specific measurement devices. Attribute information from medical devices is conveyed in a tag-length-value format based on IEEE-11073 Medical Device Encoding Rules (MDER) as described in IEEE-11073 Part 20601: Application Profile - Optimized Exchange.

UPnP SensorManagement exposes IEEE-11073 Medical Devices to UPnP clients as a sensor collection and sensor. The sensor collection exposes identifying information extracted from the Object 0 of the Medical Device System. The Sensor supports a set of Dataltem(s) representing the attributes and observation data for each medical device object reported by the IEEE-11073 Manager component. Additional Dataltem(s) supply timestamps for the time of observation (if available) as well as the time of reception by the UPnP SensorManagement device.

UPnP SensorManagement support of Persistent Metric Storage objects (PM-store) is described in Annex D.3.

### D.1.1 SensorManagement IEEE-11073 Personal Health Device Parameters

This section describes extensions to the SensorManagement General Data Model for IEEE-11073 Personal Health Devices.

**Table D.1 — IEEE-11073 specific Medical Device System Parameters**

| Name   | Type           | Acc | Req | Reference | EOC | Ver |
|--|----------------|-----|-----|-----------|-----|-----|
| /UPnP/SensorMgt/SensorCollections/#!/CollectionSpecific/IEEE-11073/                  | SingleInstance |     |     | D.1.1.2   |     |     |
| SystemType   | string         |     |     | D.1.1.3   |     |     |
| SystemModel  | string         |     |     | D.1.1.4   |     |     |
| SystemID   | string         |     |     | D.1.1.5   |     |     |
| ConfigurationsNumberOfEntries  | int            |     |     |           |     |     |
|  |                |     |     |           |     |     |
| /UPnP/SensorMgt/SensorCollections/#!/CollectionSpecific/IEEE11073/Association/       | SingleInstance |     |     | D.1.1.6   |     |     |
| RequestConfig  | int            |     |     | D.1.1.7   |     |     |
| CurrentConfig  | int            |     |     | D.1.1.8   |     |     |
| LastConfig   | int            |     |     | D.1.1.9   |     |     |
| ConfigurationsNumberOfEntries  | int            |     |     |           |     |     |
|  |                |     |     |           |     |     |
| /UPnP/SensorMgt/SensorCollections/#!/CollectionSpecific/IEEE11073/Configurations/#!/ | MultiInstance  |     |     |           |     |     |
| ConfigID   | int            |     |     |           |     |     |
| ConfigData   | string         |     |     |           |     |     |
|  |                |     |     |           |     |     |

#### D.1.1.1 /UPnP/SensorMgt/SensorCollections/#!/CollectionType

**Type:** string

**Description:** The following CollectionType value shall be used for IEEE-11073 Personal Health Devices.

```
MDS-Collection-Type ::= urn:upnp-org:smgt-sct:ieee-11073_mds:ieee-org:11073_104zzz
                        :<MDS-vendor>:<MDS-model>
```

Where: 11073\_104zzz indicates the IEEE document number defining the Medical Device Specialization.

#### D.1.1.2 /UPnP/SensorMgt/SensorCollections/#!/CollectionSpecific/IEEE-11073/

**Type:** SingleInstance

**Description:** The conditionally required IEEE-11073 *SingleInstance* node shall be included for IEEE-11073 Personal Health Devices.

#### D.1.1.3 /UPnP/SensorMgt/SensorCollections/#!/CollectionSpecific/IEEE-11073/SystemType

**Type:** CSV(string)

**Description:** The required SystemType parameter indicates the type of IEEE-11073 Medical Device System. The contents of the SystemType parameter correspond to the *System-Type* or *System-Type-Spec-List* attribute(s) as described in IEEE-11073 Part 20601: Application Profile - Optimized Exchange [8] - MDS Class. Values for the SystemType parameter shall be presented in text format as defined by the individual IEEE-11073-104zz Device specialization [9] documents.



### SensorDataModel:1

For example a IEEE-11073 Blood Pressure measurement device is encoded as the CSV(string) value:

```
MDC_DEV_SPEC_PROFILE_BP, 1
```

#### **D.1.1.4     /UPnP/SensorMgt/SensorCollections/#!/CollectionSpecific/IEEE-11073/SystemModel**

**Type:** CSV(string)

**Description:** The required *SystemModel* parameter indicates the manufacturer and model of a IEEE-11073 Medical Device System. The contents of the *SystemModel* parameter correspond to the *System-Model* attribute as described in IEEE-11073 Part 20601: Application Profile - Optimized Exchange [8] - MDS Class Values for the *SystemModel* parameter shall be presented in text format as defined by individual IEEE-11074-104zz Device specialization [9] documents.

For example, a IEEE-11073 Blood Pressure measurement device with Device-Vendor: The Company and with Device-Model: TheBPM ABC, is encoded as the CSV(**string**) value consisting of the Device-Vendor identification followed by the Device-Model identification:

```
TheComany,TheBPM ABC
```

#### **D.1.1.5     /UPnP/SensorMgt/SensorCollections/#!/CollectionSpecific/IEEE-11073/SystemID**

**Type:** string

**Description:** The required *SystemID* parameter provides a 64-bit extended unique identifier value for the IEEE-11073 Medical Device System. The contents of the *SystemID* parameter correspond to the *System-Id* attribute as described in IEEE-11073 Part 20601: Application Profile - Optimized Exchange [8] - MDS Class. The values for this *parameter* shall be presented as a 16-digit character hexadecimal string.

For example a IEEE-11073 Blood Pressure measurement device would be encoded as:

```
1122334455667707
```

#### **D.1.1.6     /UPnP/SensorMgt/SensorCollections/#!/CollectionSpecific/IEEE-11073/Association**

**Type:** SingleInstance

**Description:** The required *Association* *SingleInstance* node includes parameters which control the association process for IEEE-11073 Medical Device Systems.

#### **D.1.1.7     /UPnP/SensorMgt/SensorCollections/#!/CollectionSpecific/IEEE-11073/Association/RequestConfig**

**Type:** int

**Description:** The required *RequestConfig* parameter shall request the SensorManagement device begin a new association process with the corresponding Medical Device System. This value corresponds to the *Dev-Config-Id* value as described in IEEE-11073 Part 20601: Application Profile - Optimized Exchange [8]. If this parameter is written to ("0"), the SensorManagement device shall terminate any current association and shall disable further associations with this SensorManagement device. If this parameter is set to ("-1"), then the SensorManagement device shall automatically associate with any acceptable configuration presented by the corresponding IEEE-11073 Medical Device System.

#### **D.1.1.8     /UPnP/SensorMgt/SensorCollections/#!/CollectionSpecific/IEEE-11073/Association/CurrentConfig**

**Type:** int

### SensorDataModel:1

**Description:** The required *CurrentConfig* parameter shall indicate the current association for the corresponding Medical Device System. If there is no current association, then this parameter shall be ("0").

#### **D.1.1.9     /UPnP/SensorMgt/SensorCollections/#/CollectionSpecific/IEEE-11073/Association/LastConfig**

**Type:**           int

**Description:** The required *LastConfig* parameter shall indicate the last (or currently active) associated configuration for the corresponding Medical Device System. This parameter shall retain its value regardless of whether or not the Medical Device System is currently associated. If there was no known prior associated configuration this parameter shall be ("0").

## D.2 SensorManagement for IEEE-11073 Medical Device Sensors

### D.2.1 SensorManagement Parameters for IEEE-11073 Personal Health Sensors

The UPnP SensorManagement device represents IEEE-11073 Personal Health Devices as a SensorCollection and an included Sensor. The Sensor shall include the following parameters:

**Table D.2 — IEEE-11073 Medical Device Sensor Parameters**

| Name  | Type           | Acc | Req | Reference           | EOC | Ver |
|---|----------------|-----|-----|---------------------|-----|-----|
| /UPnP/SensorMgt/SensorCollections/#!/Sensors/#!/SensorType                      | MultilInstance |     |     |                     |     |     |
| SensorType  | string         |     |     | D.2.1.1             |     |     |
|   |                |     |     |                     |     |     |
| /UPnP/SensorMgt/SensorCollections/#!/Sensors/#!/SensorURNs/#!/SensorURN         | MultilInstance |     |     |                     |     |     |
| SensorURN   | string         |     |     | D.2.1.3             |     |     |
|   |                |     |     |                     |     |     |
| /UPnP/SensorMgt/SensorCollections/#!/Sensors/#!/SensorURNs/#!/DataItems/#!/Name | MultilInstance |     |     |                     |     |     |
| Name  | string         |     |     | D.2.1.5,<br>D.2.1.6 |     |     |
| Type  | string         |     |     | D.2.1.5,<br>D.2.1.6 |     |     |
| Encoding  | string         |     |     | D.2.1.5,<br>D.2.1.6 |     |     |

#### D.2.1.1 /UPnP/SensorMgt/SensorCollections/#!/Sensors/#!/SensorType

**Type:** string

**Description:** The following SensorType value shall be used for IEEE-11073 Medical Device Systems:

```
MOC-Sensor-Type ::= urn:upnp-org:smgt-st:ieee-11073_mds:ieee-org:11073_104zzz
                    :<MDS-vendor>:<MDS-model>:telemetry
```

Where: 11073\_104zzz indicates the IEEE document number defining the Medical Device Specialization.

#### D.2.1.2 /UPnP/SensorMgt/SensorCollections/#!/Sensors/#!/SensorSpecific/

**Type:** SingleInstance

**Description:** There are no defined child nodes for IEEE-11073 Medical Object Class sensors.

#### D.2.1.3 /UPnP/SensorMgt/SensorCollections/#!/Sensors/#!/SensorURNs/#!/SensorURN

**Type:** string

**Description:** The SensorURN parameter for IEEE-11073 Medical Object Class Sensors shall use the following value:

```
MOC-SensorURN ::= urn:upnp-org:smgt-surn:ieee-11073_mds:ieee-org:11073_104zzz
                  :<MDS-vendor>:<MDS-model>:telemetry
```

Where: 11073\_104zzz indicates the IEEE document number defining the Medical Device Specialization.

#### D.2.1.4 SensorManagement DataItems for IEEE-11073 Medical Object Class Sensors

**Type:** MultilInstance

### SensorDataModel:1

**Description:** The IEEE-11073 SensorURN shall support the following Dataltem(s) as listed in the table below. Each Dataltem shall be described by a set of three parameter nodes:

/UPnP/SensorMgt/SensorCollections/#!/Sensors/#!/SensorURNs/#!/SensorURN/Dataltems/#!/

Name

Type

Encoding

The following table lists the settings for the three parameters above to generate the named Dataltem described.

Note: These Sensor parameters do not directly report sensor data, instead they describe named Dataltems which are returned by the sensor using either SOAP actions or transport connections to the sensor.

**Table D.3 — Default Dataltem(s) for IEEE-11073 Medical Object Class Sensors**

| Name                        | Type                | Encoding      | Description  |
|-----------------------------|---------------------|---------------|--|
| <u>ClientID</u>             | <u>xsd:string</u>   | <u>string</u> | <u>ClientID</u> argument value provided from the SensorTransportGeneric ConnectSensor() action.  |
| <u>ReceiveTimeStamp</u>     | <u>xsd:dateTime</u> | <u>string</u> | The required <u>ReceiveTimeStamp</u> Dataltem for an IEEE-11073 Sensor shall provide a date-time stamp indicating when the Medical Device observation was received by the SensorManagement device.<br>For example:<br>2013-02-01T20:00:01Z<br>2013-02-01T12:00:00-08:00  |
| <u>ObservationTimeStamp</u> | <u>xsd:dateTime</u> | <u>string</u> | The required <u>ObservationTimeStamp</u> Dataltem for an IEEE-11073 Sensor shall provide a date-time stamp indicating when the observation was recorded by the Medical Device. If the Medical Device observation does not provide timestamp information, the value of this Dataltem shall be empty.<br>For example:<br>2013-02-01T20:00:01Z<br>2013-02-01T12:00:00-08:00 |
| <u>PersonID</u>             | <u>int</u>          | <u>string</u> | The allowed PersonID corresponds to the IEEE-11073 <u>person-id</u> attribute.   |

#### **D.2.1.5 Dataltem(s) for IEEE-11073 Medical Object Class Sensor Attributes**

**Type:** MultiInstance

**Description:** The IEEE-11073 SensorURN shall support the following Dataltem(s) to report the attributes of each Medical Object Class Handle within a IEEE-11073 Medical Device System. Each Dataltem shall be described by a set of three parameter nodes:

/UPnP/SensorMgt/SensorCollections/#!/Sensors/#!/SensorURNs/#!/SensorURN/Dataltems/#!/

Name

Type

Encoding

The following table summarizes the settings for the three parameter values listed above which define named Dataltem(s) that return attributes for IEEE-11073 Medical Objects.

## SensorDataModel:1

Note: These Sensor parameters do not directly report sensor data, instead they describe named Dataltem(s) which are returned by the sensor using either SOAP actions or transport connections to the sensor. See Sensor Transport Generic specification [12] for further details.

**Table D.4 — Dataltem(s) for Medical Object Class Attributes**

| Name                                       | Type   | Encoding                      |
|--|--|-------------------------------|
| <u><a href="#">\$[MDS-Handle]-Attr</a></u> | <u><a href="#">mds:MDSAttributes</a></u>       | <u><a href="#">Base64</a></u> |
| <u><a href="#">\$[MDS-Handle]-Attr</a></u> | <u><a href="#">mds:MOCMetricAttributes</a></u> | <u><a href="#">Base64</a></u> |

Where: [\[MDS-Handle\]](#) is substituted with the Handle value of the corresponding Medical Device System Object.

For MDS Object 0, the Dataltem [Type](#) parameter value `IEEE-11073-MDSAttributes` shall be used. The value of this Dataltem shall be MDS Object 0 attributes ([MDER ASN.1 AttributeList](#) structure) as defined in IEEE-11073 Part 20601: Application Profile - Optimized Exchange [x] - MDS Attributes. This Dataltem shall be returned as a Base64 encoded string.

For MDS Objects 1-n, the Dataltem [Type](#) parameter value `IEEE-11073-MetricAttributes` shall be used. The value of this Dataltem shall be MDS Object 1-n attributes ([MDER ASN.1 AttributeList](#) structure) as defined in IEEE-11073 Part 20601: Application Profile - Optimized Exchange [8] - Metric Attribute. This Dataltem shall be returned as a Base64 encoded string.

For example:

The Dataltem named

`$0-Attr`

provides a Base-64 encoded string containing the MDS Object Attributes ([MDER ASN.1 AttributeList](#) structure) for the entire Personal Health Device.

The Dataltem named

`$1-Attr`

provides a Base-64 encoded string containing the Medical Class Object Attributes ([MDER ASN.1 AttributeList](#) structure) for Handle 1 within the Medical Device System.

### **D.2.1.6 Dataltem(s) for IEEE-11073 Medical Object Class Sensors Observations**

**Type:** MultiInstance

**Description:** IEEE-11073 SensorURN shall support the following Dataltem(s) to report The observations for each Medical Object Class Handle within a IEEE-11073 Medical Device System. The Dataltem shall be described by a set of three parameter nodes:

[/UPnP/SensorMgt/SensorCollections/#/Sensors/#/SensorURNs/#/SensorURN/Dataltems/#/](#)

[Name](#)

[Type](#)

[Encoding](#)

The following table summarizes the settings for the three parameter values listed above which define named Dataltem(s) that return observations from IEEE-11073 Medical Objects.

Note: These Sensor parameters do not directly report sensor data, instead they describe named Dataltems which are returned by the sensor using either SOAP actions or transport connections to the sensor.

**Table D.5 — Dataltem(s) for Medical Object Class Observations**

| Name                              | Type                             | Encoding             |
|-----------------------------------|----------------------------------|----------------------|
| <u><i>[\$[MDS-Handle]-Obs</i></u> | <u><i>mds:MDSObservation</i></u> | <u><i>Base64</i></u> |

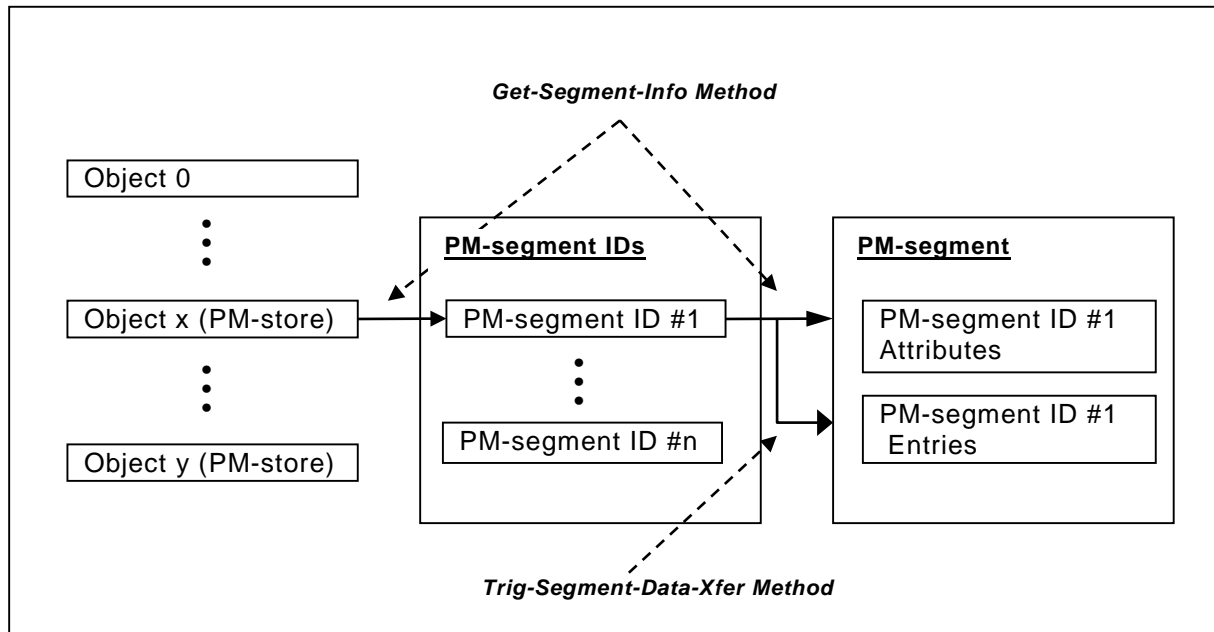
Where: *[MDS-Handle]* is substituted with the Handle value of the corresponding Medical Device System Metric Object. The Dataltem *Type* parameter value `IEEE-11073-MDSObservation` shall be used. The value of this Dataltem shall be MDS Object 1-n attributes (*MDER ASN.1 ObservationScanList* structure) as defined in IEEE-11073 Part 20601: Application Profile - Optimized Exchange [8]. This Dataltem shall be returned as a Base64 encoded string.

For example the Dataltem named

\$1-Obs

provides a Base-64 encoded string containing the MDS Object observation (*MDER ASN.1 ObservationScanList* structure) for Handle 1 of the IEEE-11073 Personal Health Device.

### D.3 SensorManagement for IEEE-11073 Persistent Metric Stores (PM-Store)



**Figure D.2 — Medical Device System with PM-store Object (informative)**

The UPnP SensorManagement device supports IEEE-11073 Persistent Metric Store (PM-store) objects. Each PM-store object supports zero or more PM-segments. Each PM-Segment contains zero or more Entries with each Entry containing an allowed timestamp header and a series of zero or more Elements. Each Element consists of a defined set of observations from one or more Metric Objects within the medical device. PM-segment's attributes define the data format which applies to all Elements within a given PM-segment.

The UPnP SensorManagement device allocates a separate Sensor node within the UPnP SensorManagement Sensor DataModel for each PM-store object with a SensorType parameter value of "upnp-org:sensors:IEEE-11073\_PM-store". This Sensor node supports parameters which control operations for the PM-store object such as requesting the transfer of the current contents of PM-store segments to the UPnP SensorManagement device. In addition, each detected PM-segment is allocated a separate Sensor node with a SensorType parameter value of "upnp-org:sensors:IEEE-11073\_PM-segment". Each Sensor node corresponding PM-segment(s) supports Dataltem(s) to read the attributes of the PM-segment and to identify and read the contents (observations) of each Element in the PM-segment.

The UPnP SensorManagement device treats each element within a PM-segment as a separate occurrence providing Dataltems (SegEntryIndex and SegElementIndex) which indicate the position of the Element within the PM-segment. When reading the PM-segment sensor all available entries within the PM-segment are sent to the UPnP Client.

#### D.3.1 SensorManagement Parameters for IEEE-11073 Persistent Metric Stores (PM-Store)

The UPnP SensorManagement device represents IEEE-11073 Personal Health Devices as a SensorCollection and an included Sensor. The Sensor shall include the following parameters:

**Table D.6 — IEEE-11073 Persistent Metric Store Parameter Nodes**

| Name | Type | Acc | Req | Reference | EOC | Ver |
|------|------|-----|-----|-----------|-----|-----|
|------|------|-----|-----|-----------|-----|-----|

### SensorDataModel:1

|   |              |  |  |         |  |  |
|---|--------------|--|--|---------|--|--|
| UPnP/SensorMgt/SensorCollections/#!/Sensors/#!/SensorsRelated/#!/ | MultInstance |  |  |         |  |  |
| SensorPath  | string       |  |  | D.3.1.2 |  |  |
|   |              |  |  |         |  |  |
| /UPnP/SensorMgt/SensorCollections/#!/Sensors/#!/                  | MultInstance |  |  |         |  |  |
| SensorType  | string       |  |  | D.3.1.3 |  |  |
| SensorUpdateRequest   |              |  |  | D.3.1.4 |  |  |
|   |              |  |  |         |  |  |

#### **D.3.1.1 /UPnP/SensorMgt/SensorCollection/#!/Sensors/#!/SensorRelated/**

**Type:** MultInstance

**Description:** The conditionally required SensorRelated MultInstance node shall be included for IEEE-11073 Medical Device Persistent Metric Store sensors. An instance of this node shall be generated for each PM-segment sensor this PM-store sensor currently supports.

#### **D.3.1.2 /UPnP/SensorMgt/SensorCollection/#!/Sensors/#!/SensorRelated/#!/SensorPath**

**Type:** string

**Description:** The required SensorPath parameter shall provide a partial path to a PM-segment sensor supported by this PM-store sensor.

#### **D.3.1.3 /UPnP/SensorMgt/SensorCollection/#!/Sensors/#!/SensorType**

**Type:** string

**Description:** The following SensorType value shall be used for IEEE-11073 Medical Device Persistent Metric Store sensors.

```
PM-store-SensorType ::= urn:upnp-org:smgt-st:ieee-11073_pm-store
                        :ieee-org:11073_104zzz:<MDS-vendor>:<MDS-model>:telemetry
```

Where: 11073\_104zzz indicates the IEEE document number defining the Medical Device Specialization.

#### **D.3.1.4 /UPnP/SensorMgt/SensorCollection/#!/Sensors/#!/SensorUpdateRequest**

**Type:** boolean

**Description:** The conditionally required SensorUpdateRequest parameter shall be implemented for IEEE-11073 PM-store Segment Sensors. When written to ("1") this parameter shall cause SensorManagement to perform the following updates:

- Update the PM-store attributes (as reported by corresponding the \$[PM-store-Handle]-Attr DataItem).
- Create new PM-Segment Sensor instances for any PM-store segments not already instantiated.



## D.4 SensorManagement Parameters for IEEE-11073 Persistent Metric Segments (PM-Segment)

### D.4.1 SensorManagement Parameters for IEEE-11073 Persistent Metric Segments (PM-Segment)

The UPnP SensorManagement device represents IEEE-11073 Personal Health Devices as a SensorCollection and an included Sensor. The Sensor shall include the following parameters:

**Table D.7 — IEEE-11073 Persistent Metric Segment Parameter Nodes**

| Name  | Type         | Acc | Req | Reference | EOC | Ver |
|---|--------------|-----|-----|-----------|-----|-----|
| UPnP/SensorMgt/SensorCollections/#!/Sensors/#!/SensorsRelated/#!/           | MultInstance |     |     |           |     |     |
| SensorPath  | string       |     |     | D.4.1.3   |     |     |
|   |              |     |     |           |     |     |
| /UPnP/SensorMgt/SensorCollections/#!/Sensors/#!/                            | MultInstance |     |     |           |     |     |
| SensorType  | string       |     |     | D.4.1.1   |     |     |
| SensorUpdateRequest   |              |     |     |           |     |     |
|   |              |     |     |           |     |     |
| /UPnP/SensorMgt/SensorCollections/#!/Sensors/#!/SensorURNs/#!/              | MultInstance |     |     |           |     |     |
| SensorURN   | string       |     |     | D.4.1.5   |     |     |
|   |              |     |     |           |     |     |
| /UPnP/SensorMgt/SensorCollections/#!/Sensors/#!/SensorURNs/#!/DataItems/#!/ | MultInstance |     |     |           |     |     |
| Name  | string       |     |     | D.4.1.6   |     |     |
| Type  | string       |     |     | D.4.1.6   |     |     |
| Encoding  | string       |     |     | D.4.1.6   |     |     |
|   |              |     |     |           |     |     |

#### D.4.1.1 /UPnP/SensorMgt/SensorCollections/#!/Sensors/#!/SensorType

**Type:** string

**Description:** The following SensorType value shall be used for IEEE-11073 Medical Device Persistent Metric Segment sensors.

```
PM-segment-SensorType ::= urn:upnp-org:smgt-st:ieee-11073_pm-segment
                           :ieee-org:11073_104zzz
                           :<MDS-vendor>:<MDS-model>:telemetry
```

Where: 11073\_104zzz indicates the IEEE document number defining the Medical Device Specialization.

#### D.4.1.2 /UPnP/SensorMgt/SensorCollections/#!/Sensors/#!/SensorRelated/

**Type:** MultInstance

**Description:** The conditionally required SensorRelated MultInstance node shall be included for IEEE-11073 PM-segment sensors. An instance of this node shall be generated for the PM-store sensor that manages this PM-segment sensor.

#### D.4.1.3 /UPnP/SensorMgt/SensorCollections/#!/Sensors/#!/SensorRelated/#!/SensorPath

**Type:** string

### SensorDataModel:1

**Description:** The required SensorPath parameter shall provide a partial path to a PM-store sensor which includes this PM-segment.

#### **D.4.1.4 /UPnP/SensorMgt/SensorCollections/#!/Sensors/#!/SensorUpdateRequest**

**Type:** boolean

**Description:** The conditionally required SensorUpdateRequest parameter shall be implemented for IEEE-11073 PM-store segment sensors. When written to “1” shall cause SensorManagement service transfer and retain a copy of the PM-store segment associated with this Sensor. When the transfer is complete this parameter shall be set to “0”.

#### **D.4.1.5 /UPnP/SensorMgt/SensorCollections/#!/Sensors/#!/SensorURNs/#!/SensorURN**

**Type:** string

**Description:** The SensorURN parameter for IEEE-11073 PM-segment Sensors shall use the following value:

```
PM-segment-SensorURN ::= urn:upnp-org:smgt-surn:ieee-11073_pm-segment
                           :ieee-org:11073_104zzz:<MDS-vendor>:<MDS-model>:telemetry
```

Where: 11073\_104zzz indicates the IEEE document number defining the Medical Device Specialization [9].

#### **D.4.1.6 Dataltems for IEEE-11073 Persistent Metric Segment Objects**

**Type:** MultiInstance

**Description:** The IEEE-11073 SensorURN shall support the following *Dataltem*(s) as listed in the table below. Each Dataltem shall be described by a set of three parameter nodes:

/UPnP/SensorMgt/SensorCollections/#!/Sensors/#!/SensorURNs/#!/SensorURN/Dataltems/#!/

Name

Type

Encoding

The following table lists the settings for the three parameters above to generate the named Dataltem described.

Note: These Sensor parameters do not directly report sensor data, instead they describe named Dataltem(s) which are returned by the sensor using either SOAP actions or transport connections to the sensor.

**Table D.8 — IEEE-11073 Persistent Metric Segment Dataltem(s)**

| Name                        | Type   | Encoding      | Description   |
|-----------------------------|--|---------------|---|
| <u>ReceiveTimeStamp</u>     | <u>xsd:dateTime</u>                                | <u>string</u> | See Table D.3   |
| <u>ObservationTimeStamp</u> | <u>xsd:dateTime</u>                                | <u>string</u> | See Table D.3   |
| <u>PersonID</u>             | <u>int</u>   | <u>string</u> | See Table D.3   |
|                             |  |               |   |
| <u>SegInstanceNumber</u>    | <u>int</u>   | <u>string</u> | The required <u>SegInstanceNumber</u> Dataltem returns the instance number of the PM-Store segment.   |
| <u>SegEntryIndex</u>        | <u>int</u>   | <u>string</u> | The required <u>SegEntryIndex</u> Dataltem provides a zero-based index corresponding to this entry within the PM-Store segment..                            |
| <u>SegElementIndex</u>      | <u>int</u>   | <u>string</u> | The required <u>SegElementIndex</u> Dataltem provides a zero-based index corresponding to this element within the PM-Store segment entry.                   |
| <u>SegEntryHeader</u>       | <u>mds:AbsoluteTime</u><br><u>mds:RelativeTime</u> | <u>Base64</u> | The required <u>SegEntryHeader</u> Dataltem provides the segment entry header contents which prefix the corresponding PM-Store segment entry. If no segment |

### SensorDataModel:1

|  |  |                               |  |
|--|--|-------------------------------|--|
|  | <u><a href="#">mds:HighResRelativeTime</a></u> |                               | entry header is defined for this PM-Store segment this entry shall return an empty string.   |
| <u><a href="#">SegEntryMap</a></u>     | <u><a href="#">mds:PmSegmentEntryMap</a></u>   | <u><a href="#">Base64</a></u> | The required <u><a href="#">SegEntryMap</a></u> Dataltem provides the contents of the PMSegmentEntryMap IEEE-11073 data type which describes the PM-Store entry segment header and element(s).   |
| <u><a href="#">SegEntryElement</a></u> | <u><a href="#">mds:SegmentElement</a></u>      | <u><a href="#">Base64</a></u> | The required <u><a href="#">SegEntryElement</a></u> Dataltem provides the contents of a single PM-Store segment element. The <u><a href="#">SegmentEntryIndex</a></u> and <u><a href="#">SegmentElementIndex</a></u> Dataltem(s) shall reflect to indices of this element within the PM-store segment. |

## Annex E Common Dataltem Names (Normative)

### E.1 Introduction

This annex contains a list of predefined Dataltem names. The Dataltem naming convention is specified in (see SensorManagement UPnP Sensor and DataStore Architecture Overview [10], subclause 4.3, "Dataltem Semantics" for additional information).

The described Dataltems are specified in such way that they will have a default unit. The used unit can be made explicit by using the units of measurement xml section in the Dataltem description (see SensorManagement UPnP Sensor and DataStore Architecture Overview [10], subclause 4.5 for additional information).

DataModels can have the vector indication “\_xyz” as suffix. This means that this value is a vector defined for [X,Y,Z]. The contents will then be then formatted as CSV with the values in order of X,Y,Z.

DataModel names can have the set point indication “\_sp” or current value indication “\_cur” as suffixes. This is done to when a sensor has latency to achieve a set point value. The value suffixed with “\_sp” is the set point value and the value suffixed with “\_cur” the current value. When the “\_sp” and “\_cur” value have the same value then the set point is achieved.

Note that both extensions are possible but should have the order of vector indication followed by setpoint or current value indication.

### E.2 Alphabetical list of the Dataltem Names

**Table E-9 Alphabetical list of common Dataltem names**

| Dataltem Name        | Section number |
|----------------------|----------------|
| <u>Acceleration</u>  | E.3.1          |
| <u>Angle</u>         | E.3.2          |
| <u>Area</u>          | E.3.3          |
| <u>BatteryState</u>  | E.3.37         |
| <u>Brightness</u>    | E.3.4          |
| <u>CleanerState</u>  | E.3.38         |
| <u>CloseToDevice</u> | E.3.39         |
| <u>ColorRGB</u>      | E.3.5          |
| <u>ControlValve</u>  | E.3.6          |
| <u>Count</u>         | E.3.7          |

SensorDataModel:1

|                      |        |
|----------------------|--------|
| <u>Current</u>       | E.3.8  |
| <u>Direction</u>     | E.3.9  |
| <u>DryingState</u>   | E.3.40 |
| <u>Flow</u>          | E.3.10 |
| <u>FlowDirection</u> | E.3.11 |
| <u>Frequency</u>     | E.3.12 |
| <u>GPS</u>           | E.3.13 |
| <u>Humidity</u>      | E.3.14 |
| <u>Location</u>      | E.3.41 |
| <u>Lock</u>          | E.3.42 |
| <u>MagnetoMeter</u>  | E.3.43 |
| <u>Mass</u>          | E.3.15 |
| <u>Moisture</u>      | E.3.16 |
| <u>Percentage</u>    | E.3.17 |
| <u>Position</u>      | E.3.18 |
| <u>Power</u>         | E.3.19 |
| <u>PowerSwitch</u>   | E.3.20 |
| <u>Presence</u>      | E.3.21 |
| <u>Pressure</u>      | E.3.22 |
| <u>Proximity</u>     | E.3.23 |
| <u>RelPosition</u>   | E.3.24 |
| <u>RelProximity</u>  | E.3.25 |
| <u>RelSpeed</u>      | E.3.26 |
| <u>Saturation</u>    | E.3.27 |
| <u>Speed</u>         | E.3.28 |

## SensorDataModel:1

|                       |        |
|-----------------------|--------|
| <u>SpeedDirection</u> | E.3.29 |
| <u>Switch</u>         | E.3.30 |
| <u>Temperature</u>    | E.3.31 |
| <u>Vibration</u>      | E.3.32 |
| <u>Voltage</u>        | E.3.33 |
| <u>Voltage_dc</u>     | E.3.34 |
| <u>Volume</u>         | E.3.35 |
| <u>WashProgram</u>    | E.3.44 |
| <u>WashState</u>      | E.3.45 |
| <u>WRDirection</u>    | E.3.36 |

### E.3 Dataltem Definitions

#### E.3.1 Dataltem Name Acceleration

This Annex describes the dataltem model of the Acceleration actuator/sensor.

##### E.3.1.1 Dataltem Acceleration definition

The Dataltem is defined with the fields defined in Table E.3-10.

Acceleration\_xyz defines the acceleration with a specific direction.

**Table E.3-10 — Dataltem Acceleration definition**

| Field  | Value               | R/A      |
|--|---------------------|----------|
| <u>Name</u>  | <u>Acceleration</u> | <u>R</u> |
| <u>Type</u>  | uda:float           | <u>R</u> |
| <u>Encoding</u>  | <u>ascii</u>        | <u>R</u> |
| <u>access</u>  | <u>rw</u>           | <u>R</u> |
| <sup>a</sup> For a device this column indicates whether the action shall be implemented or not, where <u>R</u> = required, <u>A</u> = allowed, <u>CR</u> = conditionally required, <u>CA</u> = conditionally allowed, <u>X</u> = Non-standard, add <u>-D</u> when deprecated (e.g., <u>R-D</u> , <u>A-D</u> ). |                     |          |

The Acceleration is default expressed in [m/s<sup>2</sup>].

##### E.3.1.2 Dataltem description xml

```
<DataltemDescription
xmlns="urn:schemas-upnp-org:smgt:sdmdid"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="
```

## SensorDataModel:1

```
urn:schemas-upnp-org:smgt:sdmddid.xsd
http://www.upnp.org/schemas/smgt/sdmddid-v1.xsd"
itemname="Acceleration"
access="rw">
<description>The Acceleration</description>
</DataItemDescription>
```

### E.3.2 DataItem Name Angle

This Annex describes the dataitem model of the Angle actuator/sensor.

#### E.3.2.1 DataItem Angle definition

The DataItem is defined with the fields defined in Table E.3-11.

**Table E.3-11 — DataItem Angle definition**

| Field  | Value        | R/A      |
|--|--------------|----------|
| <u>Name</u>  | <u>Angle</u> | <u>R</u> |
| <u>Type</u>  | uda:float    | <u>R</u> |
| <u>Encoding</u>  | <u>ascii</u> | <u>R</u> |
| <u>access</u>  | <u>rw</u>    | <u>R</u> |
| <sup>a</sup> For a device this column indicates whether the action shall be implemented or not, where <u>R</u> = required, <u>A</u> = allowed, <u>CR</u> = conditionally required, <u>CA</u> = conditionally allowed, <u>X</u> = Non-standard, add <u>-D</u> when deprecated (e.g., <u>R-D</u> , <u>A-D</u> ). |              |          |

The Angle is default expressed in degrees between [0,360].

#### E.3.2.2 DataItem description xml

```
<DataItemDescription
xmlns="urn:schemas-upnp-org:smgt:sdmddid"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="
urn:schemas-upnp-org:smgt:sdmddid.xsd
http://www.upnp.org/schemas/smgt/sdmddid-v1.xsd"
itemname="Angle"
access="rw">
<description>The Angle</description>
</DataItemDescription>
```

### E.3.3 DataItem Name Area

This Annex describes the dataitem model of the Area sensor.

#### E.3.3.1 DataItem Area definition

The sensor is defined with the fields defined in Table E.3-12.

**Table E.3-12 — DataItem Area definition**

| Field           | Value        | R/A      |
|-----------------|--------------|----------|
| <u>Name</u>     | <u>Area</u>  | <u>R</u> |
| <u>Type</u>     | uda:float    | <u>R</u> |
| <u>Encoding</u> | <u>ascii</u> | <u>R</u> |
| <u>access</u>   | <u>rw</u>    | <u>R</u> |

## SensorDataModel:1

| Field  | Value | R/A |
|--|-------|-----|
| <sup>a</sup> For a device this column indicates whether the action shall be implemented or not, where <u>R</u> = required, <u>A</u> = allowed, <u>CR</u> = conditionally required, <u>CA</u> = conditionally allowed, <u>X</u> = Non-standard, add <u>-D</u> when deprecated (e.g., <u>R-D</u> , <u>A-D</u> ). |       |     |

The Area is default expressed in square meters [m<sup>2</sup>].

### E.3.3.2 Dataltem description xml

```
<DataltemDescription
xmlns="urn:schemas-upnp-org:smgt:sdmddid"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="
urn:schemas-upnp-org:smgt:sdmddid.xsd
http://www.upnp.org/schemas/smgmt/sdmddid-v1.xsd"
itemname="Area"
access="rw">
<description>The Area measurement</description>
</DataltemDescription>
```

### E.3.4 Dataltem Name Brightness

This Annex describes the dataltem model of the Brightness actuator/sensor.

#### E.3.4.1 Dataltem Brightness definition

The sensor is defined with the fields defined in Table E.3-13.

Table E.3-13 — Dataltem Brightness definition

| Field  | Value             | R/A      |
|--|-------------------|----------|
| <u>Name</u>  | <u>Brightness</u> | <u>R</u> |
| <u>Type</u>  | uda:int32         | <u>R</u> |
| <u>Encoding</u>  | <u>ascii</u>      | <u>R</u> |
| <u>access</u>  | <u>rw</u>         | <u>R</u> |
| <sup>a</sup> For a device this column indicates whether the action shall be implemented or not, where <u>R</u> = required, <u>A</u> = allowed, <u>CR</u> = conditionally required, <u>CA</u> = conditionally allowed, <u>X</u> = Non-standard, add <u>-D</u> when deprecated (e.g., <u>R-D</u> , <u>A-D</u> ). |                   |          |

The Brightness is used for specifying the brightness of a light/display. This model specifies the intensity on a scale of 0 to 100, with 0 (zero) indicating the minimum intensity.

### E.3.4.2 Dataltem description xml

```
<DataltemDescription
xmlns="urn:schemas-upnp-org:smgt:sdmddid"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="
urn:schemas-upnp-org:smgt:sdmddid.xsd
http://www.upnp.org/schemas/smgmt/sdmddid-v1.xsd"
itemname="Brightness"
access="rw">
<description>The Brightness of a light</description>
</DataltemDescription>
```

### E.3.5 Dataltem Name ColorRGB

This Annex describes the dataltem model of the ColorRGB actuator/sensor.



**E.3.5.1 Dataltem ColorRGB definition**

The sensor is defined with the fields defined in Table E.3-14.

**Table E.3-14 — Dataltem ColorRGB definition**

| Field  | Value           | R/A      |
|--|-----------------|----------|
| <u>Name</u>  | <u>ColorRGB</u> | <u>R</u> |
| <u>Type</u>  | uda:int32       | <u>R</u> |
| <u>Encoding</u>  | <u>ascii</u>    | <u>R</u> |
| <u>access</u>  | <u>rw</u>       | <u>R</u> |
| <sup>a</sup> For a device this column indicates whether the action shall be implemented or not, where <u>R</u> = required, <u>A</u> = allowed, <u>CR</u> = conditionally required, <u>CA</u> = conditionally allowed, <u>X</u> = Non-standard, add <u>-D</u> when deprecated (e.g., <u>R-D</u> , <u>A-D</u> ). |                 |          |

The RGBColor model is used for specifying colors. This model specifies the intensity of red, green, and blue on a scale of 0 to 255, with 0 (zero) indicating the minimum intensity. The settings of the three colors are converted to a single integer value by using this formula:  

$$\text{RGB value} = \text{Red} + (\text{Green} * 256) + (\text{Blue} * 256 * 256)$$

**E.3.5.2 Dataltem description xml**

```
<DataltemDescription
xmlns="urn:schemas-upnp-org:smgt:sdmddid"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="
urn:schemas-upnp-org:smgt:sdmddid.xsd
http://www.upnp.org/schemas/smgt/sdmddid-v1.xsd"
itemname="ColorRGB"
access="rw">
<description>RGB value</description>
</DataltemDescription>
```

**E.3.6 Dataltem Name ControlValve**

This Annex describes the dataltem model of the ControlValve actuator/sensor.

**E.3.6.1 Dataltem ControlValve definition**

The sensor is defined with the fields defined in Table E.3-15.

**Table E.3-15 — Dataltem ControlValve definition**

| Field  | Value               | R/A      |
|--|---------------------|----------|
| <u>Name</u>  | <u>ControlValve</u> | <u>R</u> |
| <u>Type</u>  | uda:string          | <u>R</u> |
| <u>Encoding</u>  | <u>ascii</u>        | <u>R</u> |
| <u>access</u>  | <u>rw</u>           | <u>R</u> |
| <sup>a</sup> For a device this column indicates whether the action shall be implemented or not, where <u>R</u> = required, <u>A</u> = allowed, <u>CR</u> = conditionally required, <u>CA</u> = conditionally allowed, <u>X</u> = Non-standard, add <u>-D</u> when deprecated (e.g., <u>R-D</u> , <u>A-D</u> ). |                     |          |

The ControlValve can be expressed as: "open", "close" and "auto".

**E.3.6.2 Dataltem description xml**

```
<DataltemDescription
xmlns="urn:schemas-upnp-org:smgt:sdmddid"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
```

### SensorDataModel:1

```
xsi:schemaLocation="
urn:schemas-upnp-org:smgt:sdmddid.xsd
http://www.upnp.org/schemas/smgt/sdmddid-v1.xsd"
itemname="ControlValve"
access="rw">
<description>The controlvalve value</description>
</DataItemDescription>
```

#### E.3.7 DataItem Name Count

This Annex describes the dataitem model of an Count actuator/sensor.

This is an abstracted value that can be used count or set occurrences.

##### E.3.7.1 DataItem Count definition

The sensor is defined with the fields defined in Table E.3-16.

**Table E.3-16 — DataItem Count definition**

| Field  | Value        | R/A      |
|--|--------------|----------|
| <u>Name</u>  | <u>Count</u> | <u>R</u> |
| <u>Type</u>  | uda:int32    | <u>R</u> |
| <u>Encoding</u>  | <u>ascii</u> | <u>R</u> |
| <u>access</u>  | <u>rw</u>    | <u>R</u> |
| <sup>a</sup> For a device this column indicates whether the action shall be implemented or not, where <u>R</u> = required, <u>A</u> = allowed, <u>CR</u> = conditionally required, <u>CA</u> = conditionally allowed, <u>X</u> = Non-standard, add <u>-D</u> when deprecated (e.g., <u>R-D</u> , <u>A-D</u> ). |              |          |

##### E.3.7.2 DataItem description xml

```
<DataItemDescription
xmlns="urn:schemas-upnp-org:smgt:sdmddid"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="
urn:schemas-upnp-org:smgt:sdmddid.xsd
http://www.upnp.org/schemas/smgt/sdmddid-v1.xsd"
itemname="Count"
access="rw">
<description>counted value</description>
</DataItemDescription>
```

#### E.3.8 DataItem Name Current

This Annex describes the dataitem model of an Current sensor.

##### E.3.8.1 DataItem Current definition

The sensor is defined with the fields defined in Table E.3-17.

**Table E.3-17 — DataItem Current definition**

| Field           | Value          | R/A      |
|-----------------|----------------|----------|
| <u>Name</u>     | <u>Current</u> | <u>R</u> |
| <u>Type</u>     | uda:float      | <u>R</u> |
| <u>Encoding</u> | <u>ascii</u>   | <u>R</u> |

## SensorDataModel:1

| Field   | Value     | R/A      |
|---|-----------|----------|
| <u>access</u>   | <i>ro</i> | <u>R</u> |
| a For a device this column indicates whether the action shall be implemented or not, where <u>R</u> = required, <u>A</u> = allowed, <u>CR</u> = conditionally required, <u>CA</u> = conditionally allowed, <u>X</u> = Non-standard, add <u>-D</u> when deprecated (e.g., <u>R-D</u> , <u>A-D</u> ). |           |          |

The Current is default expressed in Ampere [A].

### E.3.8.2 Dataltem description xml

```
<DataltemDescription
xmlns="urn:schemas-upnp-org:smgt:sdmddid"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="
urn:schemas-upnp-org:smgt:sdmddid.xsd
http://www.upnp.org/schemas/smgt/sdmddid-v1.xsd"
itemname="Current"
access="ro">
<description>Current value</description>
</DataltemDescription>
```

### E.3.9 Dataltem Name Direction

This Annex describes the dataltem model of an Direction actuator/sensor.

#### E.3.9.1 Dataltem Direction definition

The sensor is defined with the fields defined in Table E.3-18.

Table E.3-18 — Dataltem Direction definition

| Field   | Value            | R/A      |
|---|------------------|----------|
| <u>Name</u>   | <u>Direction</u> | <u>R</u> |
| <u>Type</u>   | uda:string       | <u>R</u> |
| <u>Encoding</u>   | <u>ascii</u>     | <u>R</u> |
| <u>access</u>   | <i>rw</i>        | <u>R</u> |
| a For a device this column indicates whether the action shall be implemented or not, where <u>R</u> = required, <u>A</u> = allowed, <u>CR</u> = conditionally required, <u>CA</u> = conditionally allowed, <u>X</u> = Non-standard, add <u>-D</u> when deprecated (e.g., <u>R-D</u> , <u>A-D</u> ). |                  |          |

The Direction is expressed with values: "up", "down", "left", "right", "forward", "reverse" and the values specified for WRDirection (see section E.3.36).

#### E.3.9.2 Dataltem description xml

```
<DataltemDescription
xmlns="urn:schemas-upnp-org:smgt:sdmddid"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="
urn:schemas-upnp-org:smgt:sdmddid.xsd
http://www.upnp.org/schemas/smgt/sdmddid-v1.xsd"
itemname="Direction"
access="rw">
<description>Direction measurement</description>
</DataltemDescription>
```

## SensorDataModel:1

### E.3.10 Dataltem Name Flow

This Annex describes the dataltem model of an Flow actuator/sensor.

#### E.3.10.1 Dataltem Flow definition

The sensor is defined with the fields defined in Table E.3-19.

**Table E.3-19 — Dataltem Flow definition**

| Field  | Value        | R/A      |
|--|--------------|----------|
| <u>Name</u>  | <u>Flow</u>  | <u>R</u> |
| <u>Type</u>  | uda:float    | <u>R</u> |
| <u>Encoding</u>  | <u>ascii</u> | <u>R</u> |
| <u>access</u>  | <u>rw</u>    | <u>R</u> |
| <sup>a</sup> For a device this column indicates whether the action shall be implemented or not, where <u>R</u> = required, <u>A</u> = allowed, <u>CR</u> = conditionally required, <u>CA</u> = conditionally allowed, <u>X</u> = Non-standard, add <u>-D</u> when deprecated (e.g., <u>R-D</u> , <u>A-D</u> ). |              |          |

The Flow is default expressed in [m<sup>3</sup>/s].

#### E.3.10.2 Dataltem description xml

```
<DataltemDescription
xmlns="urn:schemas-upnp-org:smgt:sdmddid"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="
urn:schemas-upnp-org:smgt:sdmddid.xsd
http://www.upnp.org/schemas/smgt/sdmddid-v1.xsd"
itemname="Flow"
access="rw">
<description>Flow sensor</description>
</DataltemDescription>
```

### E.3.11 Dataltem Name FlowDirection

This Annex describes the dataltem model of an FlowDirection actuator/sensor.

#### E.3.11.1 Dataltem FlowDirection definition

The sensor is defined with the fields defined in Table E.3-20.

**Table E.3-20 — Dataltem FlowDirection definition**

| Field  | Value                | R/A      |
|--|----------------------|----------|
| <u>Name</u>  | <u>FlowDirection</u> | <u>R</u> |
| <u>Type</u>  | uda:string           | <u>R</u> |
| <u>Encoding</u>  | <u>ascii</u>         | <u>R</u> |
| <u>access</u>  | <u>rw</u>            | <u>R</u> |
| <sup>a</sup> For a device this column indicates whether the action shall be implemented or not, where <u>R</u> = required, <u>A</u> = allowed, <u>CR</u> = conditionally required, <u>CA</u> = conditionally allowed, <u>X</u> = Non-standard, add <u>-D</u> when deprecated (e.g., <u>R-D</u> , <u>A-D</u> ). |                      |          |

The FlowDirection is expressed as: "in" and "out".

#### E.3.11.2 Dataltem description xml

```
<DataltemDescription
xmlns="urn:schemas-upnp-org:smgt:sdmddid"
```

### SensorDataModel:1

```
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation=".
urn:schemas-upnp-org:smgt:sdmddid.xsd
http://www.upnp.org/schemas/smgt/sdmddid-v1.xsd"
itemname="FlowDirection"
access="rw">
<description>FlowDirection sensor</description>
</DataltemDescription>
```

#### E.3.12 Dataltem Name Frequency

This Annex describes the dataltem model of an (voltage) Frequency sensor/actuator.

##### E.3.12.1 Dataltem Frequency definition

The sensor is defined with the fields defined in Table E.3-21.

**Table E.3-21 — Dataltem Frequency definition**

| Field   | Value            | R/A      |
|---|------------------|----------|
| <u>Name</u>   | <u>Frequency</u> | <u>R</u> |
| <u>Type</u>   | uda:int32        | <u>R</u> |
| <u>Encoding</u>   | <u>ascii</u>     | <u>R</u> |
| <u>access</u>   | <u>ro</u>        | <u>R</u> |
| a For a device this column indicates whether the action shall be implemented or not, where <u>R</u> = required, <u>A</u> = allowed, <u>CR</u> = conditionally required, <u>CA</u> = conditionally allowed, <u>X</u> = Non-standard, add <u>-D</u> when deprecated (e.g., <u>R-D</u> , <u>A-D</u> ). |                  |          |

The Frequency is default expressed in [Hz].

##### E.3.12.2 Dataltem description xml

```
<DataltemDescription
xmlns="urn:schemas-upnp-org:smgt:sdmddid"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation=".
urn:schemas-upnp-org:smgt:sdmddid.xsd
http://www.upnp.org/schemas/smgt/sdmddid-v1.xsd"
itemname="Frequency"
access="ro">
<description>Frequency</description>
</DataltemDescription>
```

#### E.3.13 Dataltem Name GPS

This Annex describes the dataltem model of an GPS sensor.

##### E.3.13.1 Dataltem GPS definition

The sensor is defined with the fields defined in Table E.3-22.

**Table E.3-22 — Dataltem GPS definition**

| Field           | Value        | R/A      |
|-----------------|--------------|----------|
| <u>Name</u>     | <u>GPS</u>   | <u>R</u> |
| <u>Type</u>     | uda:string   | <u>R</u> |
| <u>Encoding</u> | <u>ascii</u> | <u>R</u> |

## SensorDataModel:1

| Field  | Value     | R/A      |
|--|-----------|----------|
| <u>access</u>  | <i>ro</i> | <u>R</u> |
| <sup>a</sup> For a device this column indicates whether the action shall be implemented or not, where <u>R</u> = required, <u>A</u> = allowed, <u>CR</u> = conditionally required, <u>CA</u> = conditionally allowed, <u>X</u> = Non-standard, add <u>-D</u> when deprecated (e.g., <u>R-D</u> , <u>A-D</u> ). |           |          |

The GPS coordinate is defined in unit coord-gps or as unit coord-dms section 4.5.6 of [10].

### E.3.13.2 Dataltem description xml

```
<DataltemDescription
xmlns="urn:schemas-upnp-org:smgt:sdmddid"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="
urn:schemas-upnp-org:smgt:sdmddid.xsd
http://www.upnp.org/schemas/smgt/sdmddid-v1.xsd"
itemname="GPS"
access="ro">
<description>GPS position</description>
</DataltemDescription>
```

### E.3.14 Dataltem Name Humidity

This Annex describes the dataltem model of an Humidity sensor.

#### E.3.14.1 Dataltem Humidity definition

The sensor is defined with the fields defined in Table E.3-23.

Table E.3-23 — Dataltem Humidity definition

| Field  | Value           | R/A      |
|--|-----------------|----------|
| <u>Name</u>  | <u>Humidity</u> | <u>R</u> |
| <u>Type</u>  | uda:float       | <u>R</u> |
| <u>Encoding</u>  | <u>ascii</u>    | <u>R</u> |
| <u>access</u>  | <i>ro</i>       | <u>R</u> |
| <sup>a</sup> For a device this column indicates whether the action shall be implemented or not, where <u>R</u> = required, <u>A</u> = allowed, <u>CR</u> = conditionally required, <u>CA</u> = conditionally allowed, <u>X</u> = Non-standard, add <u>-D</u> when deprecated (e.g., <u>R-D</u> , <u>A-D</u> ). |                 |          |

The Humidity is expressed in percentage of water in the atmosphere.

### E.3.14.2 Dataltem description xml

```
<DataltemDescription
xmlns="urn:schemas-upnp-org:smgt:sdmddid"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="
urn:schemas-upnp-org:smgt:sdmddid.xsd
http://www.upnp.org/schemas/smgt/sdmddid-v1.xsd"
itemname="Humidity"
access="ro">
<description>Humidity</description>
</DataltemDescription>
```

### E.3.15 Dataltem Name Mass

This Annex describes the dataltem model of an Mass sensor.

### E.3.15.1 Dataltem Mass definition

The sensor is defined with the fields defined in Table E.3-24.

**Table E.3-24 — Dataltem Mass definition**

| Field  | Value        | R/A      |
|--|--------------|----------|
| <u>Name</u>  | <u>Mass</u>  | <u>R</u> |
| <u>Type</u>  | uda:float    | <u>R</u> |
| <u>Encoding</u>  | <u>ascii</u> | <u>R</u> |
| <u>access</u>  | <u>ro</u>    | <u>R</u> |
| <sup>a</sup> For a device this column indicates whether the action shall be implemented or not, where <u>R</u> = required, <u>A</u> = allowed, <u>CR</u> = conditionally required, <u>CA</u> = conditionally allowed, <u>X</u> = Non-standard, add <u>-D</u> when deprecated (e.g., <u>R-D</u> , <u>A-D</u> ). |              |          |

The Mass is default expressed in [kg].

### E.3.15.2 Dataltem description xml

```
<DataltemDescription
xmlns="urn:schemas-upnp-org:smgt:sdmddid"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="
urn:schemas-upnp-org:smgt:sdmddid.xsd
http://www.upnp.org/schemas/smgt/sdmddid-v1.xsd"
itemname="Mass"
access="ro">
<description>Mass</description>
</DataltemDescription>
```

### E.3.16 Dataltem Name Moisture

This Annex describes the dataltem model of an Moisture sensor.

#### E.3.16.1 Dataltem Moisture definition

The sensor is defined with the fields defined in Table E.3-25.

**Table E.3-25 — Dataltem Moisture definition**

| Field  | Value           | R/A      |
|--|-----------------|----------|
| <u>Name</u>  | <u>Moisture</u> | <u>R</u> |
| <u>Type</u>  | uda:float       | <u>R</u> |
| <u>Encoding</u>  | <u>ascii</u>    | <u>R</u> |
| <u>access</u>  | <u>ro</u>       | <u>R</u> |
| <sup>a</sup> For a device this column indicates whether the action shall be implemented or not, where <u>R</u> = required, <u>A</u> = allowed, <u>CR</u> = conditionally required, <u>CA</u> = conditionally allowed, <u>X</u> = Non-standard, add <u>-D</u> when deprecated (e.g., <u>R-D</u> , <u>A-D</u> ). |                 |          |

The Moisture is expressed in percentage of water in the measured substance.

### E.3.16.2 Dataltem description xml

```
<DataltemDescription
xmlns="urn:schemas-upnp-org:smgt:sdmddid"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="
```

### SensorDataModel:1

```
urn:schemas-upnp-org:smgt:sdmdid.xsd
http://www.upnp.org/schemas/smgt/sdmdid-v1.xsd"
itemname="Moisture"
access="ro">
<description>Moisture</description>
</DataItemDescription>
```

#### E.3.17 DataItem Name Percentage

This Annex describes the dataitem model of an Percentage actuator/sensor.

##### E.3.17.1 DataItem Percentage definition

The sensor is defined with the fields defined in Table E.3-26.

**Table E.3-26 — DataItem Percentage definition**

| Field  | Value             | R/A      |
|--|-------------------|----------|
| <u>Name</u>  | <u>Percentage</u> | <u>R</u> |
| <u>Type</u>  | uda:int16         | <u>R</u> |
| <u>Encoding</u>  | <u>ascii</u>      | <u>R</u> |
| <u>access</u>  | <u>rw</u>         | <u>R</u> |
| <sup>a</sup> For a device this column indicates whether the action shall be implemented or not, where <u>R</u> = required, <u>A</u> = allowed, <u>CR</u> = conditionally required, <u>CA</u> = conditionally allowed, <u>X</u> = Non-standard, add <u>-D</u> when deprecated (e.g., <u>R-D</u> , <u>A-D</u> ). |                   |          |

The Percentage is used for specifying an value between [0,100] where 0 is the minimum value and 100 is the maximum value.

##### E.3.17.2 DataItem description xml

```
<DataItemDescription
xmlns="urn:schemas-upnp-org:smgt:sdmdid"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="
urn:schemas-upnp-org:smgt:sdmdid.xsd
http://www.upnp.org/schemas/smgt/sdmdid-v1.xsd"
itemname="Percentage"
access="rw">
<description>Percentage</description>
</DataItemDescription>
```

#### E.3.18 DataItem Name Position

This Annex describes the dataitem model of an position actuator/sensor.

The generic DataItem Name is defined as Position.

The current position is defined as Position\_cur.

The set point (wanted) is defined as Position\_sp.

##### E.3.18.1 DataItem Position definition

The sensor is defined with the fields defined in Table E.3-27.



**Table E.3-27 — Dataltem Position definition**

| Field  | Value           | R/A      |
|--|-----------------|----------|
| <u>Name</u>  | <u>Position</u> | <u>R</u> |
| <u>Type</u>  | uda:float       | <u>R</u> |
| <u>Encoding</u>  | <u>ascii</u>    | <u>R</u> |
| <u>access</u>  | <u>rw</u>       | <u>R</u> |
| <sup>a</sup> For a device this column indicates whether the action shall be implemented or not, where <u>R</u> = required, <u>A</u> = allowed, <u>CR</u> = conditionally required, <u>CA</u> = conditionally allowed, <u>X</u> = Non-standard, add <u>-D</u> when deprecated (e.g., <u>R-D</u> , <u>A-D</u> ). |                 |          |

The Position is default expressed in meters [m].

### E.3.18.2 Dataltem description xml

```
<DataltemDescription
xmlns="urn:schemas-upnp-org:smgt:sdmddid"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="
urn:schemas-upnp-org:smgt:sdmddid.xsd
http://www.upnp.org/schemas/smgt/sdmddid-v1.xsd"
itemname="Position"
access="rw">
<description>The absolute position</description>
</DataltemDescription>
```

### E.3.19 Dataltem Name Power

This Annex describes the dataltem model of the Power actuator/sensor.

#### E.3.19.1 Dataltem Power definition

The sensor is defined with the fields defined in Table E.3-28.

**Table E.3-28 — Dataltem Power definition**

| Field  | Value        | R/A      |
|--|--------------|----------|
| <u>Name</u>  | <u>Power</u> | <u>R</u> |
| <u>Type</u>  | uda:float    | <u>R</u> |
| <u>Encoding</u>  | <u>ascii</u> | <u>R</u> |
| <u>access</u>  | <u>rw</u>    | <u>R</u> |
| <sup>a</sup> For a device this column indicates whether the action shall be implemented or not, where <u>R</u> = required, <u>A</u> = allowed, <u>CR</u> = conditionally required, <u>CA</u> = conditionally allowed, <u>X</u> = Non-standard, add <u>-D</u> when deprecated (e.g., <u>R-D</u> , <u>A-D</u> ). |              |          |

The Power is default expressed in watts [W].

### E.3.19.2 Dataltem description xml

```
<DataltemDescription
xmlns="urn:schemas-upnp-org:smgt:sdmddid"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="
urn:schemas-upnp-org:smgt:sdmddid.xsd
http://www.upnp.org/schemas/smgt/sdmddid-v1.xsd"
itemname="Power"
access="rw">
```

## SensorDataModel:1

<description>The power</description>  
</DataltemDescription>

### E.3.20 Dataltem Name PowerSwitch

This Annex describes the dataltem model of an PowerSwitch actuator/sensor.

#### E.3.20.1 Dataltem PowerSwitch definition

The sensor is defined with the fields defined in Table E.3-29.

**Table E.3-29 — Dataltem PowerSwitch definition**

| Field  | Value              | R/A      |
|--|--------------------|----------|
| <u>Name</u>  | <u>PowerSwitch</u> | <u>R</u> |
| <u>Type</u>  | uda:string         | <u>R</u> |
| <u>Encoding</u>  | <u>ascii</u>       | <u>R</u> |
| <u>access</u>  | <u>rw</u>          | <u>R</u> |
| <sup>a</sup> For a device this column indicates whether the action shall be implemented or not, where <u>R</u> = required, <u>A</u> = allowed, <u>CR</u> = conditionally required, <u>CA</u> = conditionally allowed, <u>X</u> = Non-standard, add <u>-D</u> when deprecated (e.g., <u>R-D</u> , <u>A-D</u> ). |                    |          |

The PowerSwitch is expressed as; “on”, “off” or “sleep”.

#### E.3.20.2 Dataltem description xml

```
<DataltemDescription
xmlns="urn:schemas-upnp-org:smgt:sdmddid"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="
urn:schemas-upnp-org:smgt:sdmddid.xsd
http://www.upnp.org/schemas/smgtsdmdid-v1.xsd"
itemname="PowerSwitch"
access="rw">
<description>The power switch</description>
</DataltemDescription>
```

### E.3.21 Dataltem Name Presence

This Annex describes the dataltem model of an Presence actuator/sensor.

#### E.3.21.1 Dataltem Presence definition

The sensor is defined with the fields defined in Table E.3-30.

**Table E.3-30 — Dataltem Presence definition**

| Field  | Value           | R/A      |
|--|-----------------|----------|
| <u>Name</u>  | <u>Presence</u> | <u>R</u> |
| <u>Type</u>  | uda:string      | <u>R</u> |
| <u>Encoding</u>  | <u>ascii</u>    | <u>R</u> |
| <u>access</u>  | <u>rw</u>       | <u>R</u> |
| <sup>a</sup> For a device this column indicates whether the action shall be implemented or not, where <u>R</u> = required, <u>A</u> = allowed, <u>CR</u> = conditionally required, <u>CA</u> = conditionally allowed, <u>X</u> = Non-standard, add <u>-D</u> when deprecated (e.g., <u>R-D</u> , <u>A-D</u> ). |                 |          |

The Presence is used for specifying presence, and is expressed as “present” and “notpresent”.

### SensorDataModel:1

Note that the value "present" can also be interpreted as available and the value "notpresent" as unavailable.

#### **E.3.21.2 Dataltem description xml**

```
<DataltemDescription
xmlns="urn:schemas-upnp-org:smgt:sdmddid"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="
urn:schemas-upnp-org:smgt:sdmddid.xsd
http://www.upnp.org/schemas/smgt/sdmddid-v1.xsd"
itemname="Presence"
access="rw">
<description>The presence indication</description>
</DataltemDescription>
```

#### **E.3.22 Dataltem Name Pressure**

This Annex describes the dataltem model of the Presure actuator/sensor.

The generic Dataltem name is defined as Pressure.

The current pressure is defined as Pressure\_cur.

The set point (wanted) presure is defined as Pressure\_sp.

##### **E.3.22.1 Dataltem Pressure definition**

The sensor is defined with the fields defined in Table E.3-31.

**Table E.3-31 — Dataltem Pressure definition**

| Field  | Value           | R/A      |
|--|-----------------|----------|
| <u>Name</u>  | <u>Pressure</u> | <u>R</u> |
| <u>Type</u>  | uda:int32       | <u>R</u> |
| <u>Encoding</u>  | <u>ascii</u>    | <u>R</u> |
| <u>access</u>  | <u>rw</u>       | <u>R</u> |
| <sup>a</sup> For a device this column indicates whether the action shall be implemented or not, where <u>R</u> = required, <u>A</u> = allowed, <u>CR</u> = conditionally required, <u>CA</u> = conditionally allowed, <u>X</u> = Non-standard, add <u>-D</u> when deprecated (e.g., <u>R-D</u> , <u>A-D</u> ). |                 |          |

The Pressure is default expressed in pascal [Pa].

#### **E.3.22.2 Dataltem description xml**

```
<DataltemDescription
xmlns="urn:schemas-upnp-org:smgt:sdmddid"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="
urn:schemas-upnp-org:smgt:sdmddid.xsd
http://www.upnp.org/schemas/smgt/sdmddid-v1.xsd"
itemname="Pressure"
access="rw">
<description>The pressure of a system</description>
</DataltemDescription>
```

## SensorDataModel:1

### E.3.23 Dataltem Name Proximity

This Annex describes the dataltem model of an Proximity actuator/sensor.

#### E.3.23.1 Dataltem Proximity definition

The sensor is defined with the fields defined inTable E.3-32.

**Table E.3-32 — Dataltem Proximity definition**

| Field  | Value            | R/A      |
|--|------------------|----------|
| <u>Name</u>  | <u>Proximity</u> | <u>R</u> |
| <u>Type</u>  | uda:float        | <u>R</u> |
| <u>Encoding</u>  | <u>ascii</u>     | <u>R</u> |
| <u>access</u>  | <u>rw</u>        | <u>R</u> |
| <sup>a</sup> For a device this column indicates whether the action shall be implemented or not, where <u>R</u> = required, <u>A</u> = allowed, <u>CR</u> = conditionally required, <u>CA</u> = conditionally allowed, <u>X</u> = Non-standard, add <u>-D</u> when deprecated (e.g., <u>R-D</u> , <u>A-D</u> ). |                  |          |

The Proximity is default expressed in meters [m].

#### E.3.23.2 Dataltem description xml

```
<DataltemDescription
xmlns="urn:schemas-upnp-org:smgt:sdmdid"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="
urn:schemas-upnp-org:smgt:sdmdid.xsd
http://www.upnp.org/schemas/smgt/sdmdid-v1.xsd"
itemname="Proximity"
access="rw">
<description>Distance measurement</description>
</DataltemDescription>
```

### E.3.24 Dataltem Name RelPosition

This Annex describes the dataltem model of an relative position actuator/sensor.

The generic Dataltem Name is defined as RelPosition.

The current position is defined as RelPosition\_cur.

The set point (wanted) position is defined as RelPosition\_sp.

#### E.3.24.1 Dataltem RelPosition definition

The sensor is defined with the fields defined in Table E.3-33.

**Table E.3-33 — Dataltem RelPosition definition**

| Field           | Value              | R/A      |
|-----------------|--------------------|----------|
| <u>Name</u>     | <u>RelPosition</u> | <u>R</u> |
| <u>Type</u>     | uda:float          | <u>R</u> |
| <u>Encoding</u> | <u>ascii</u>       | <u>R</u> |
| <u>access</u>   | <u>rw</u>          | <u>R</u> |

### SensorDataModel:1

| Field   | Value | R/A |
|---|-------|-----|
| a For a device this column indicates whether the action shall be implemented or not, where <u>R</u> = required, <u>A</u> = allowed, <u>CR</u> = conditionally required, <u>CA</u> = conditionally allowed, <u>X</u> = Non-standard, add <u>-D</u> when deprecated (e.g., <u>R-D</u> , <u>A-D</u> ). |       |     |

The RelPosition is relative and is expressed in a value between [0,100] where 0 means closed and 100 means fully open.

#### E.3.24.2 Dataltem description xml

```
<DataltemDescription
xmlns="urn:schemas-upnp-org:smgt:sdmdid"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="
urn:schemas-upnp-org:smgt:sdmdid.xsd
http://www.upnp.org/schemas/smg/sdmdid-v1.xsd"
itemname="RelPosition"
access="rw">
<description>The relative position</description>
</DataltemDescription>
```

#### E.3.25 Dataltem Name RelProximity

This Annex describes the dataltem model of an RelProximity actuator/sensor.

##### E.3.25.1 Dataltem RelProximity definition

The sensor is defined with the fields defined in Table E.3-34.

Table E.3-34 — Dataltem RelProximity definition

| Field   | Value               | R/A      |
|---|---------------------|----------|
| <u>Name</u>   | <u>RelProximity</u> | <u>R</u> |
| <u>Type</u>   | uda:float           | <u>R</u> |
| <u>Encoding</u>   | <u>ascii</u>        | <u>R</u> |
| <u>access</u>   | <u>rw</u>           | <u>R</u> |
| a For a device this column indicates whether the action shall be implemented or not, where <u>R</u> = required, <u>A</u> = allowed, <u>CR</u> = conditionally required, <u>CA</u> = conditionally allowed, <u>X</u> = Non-standard, add <u>-D</u> when deprecated (e.g., <u>R-D</u> , <u>A-D</u> ). |                     |          |

The RelProximity (distance) is relative and is expressed in a value between [0,100] where 0 means no distance, and 100 means max distance.

#### E.3.25.2 Dataltem description xml

```
<DataltemDescription
xmlns="urn:schemas-upnp-org:smgt:sdmdid"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="
urn:schemas-upnp-org:smgt:sdmdid.xsd
http://www.upnp.org/schemas/smg/sdmdid-v1.xsd"
itemname="RelProximity"
access="rw">
<description>The Relative Distance</description>
</DataltemDescription>
```

## SensorDataModel:1

### E.3.26 Dataltem Name RelSpeed

This Annex describes the dataltem model of an relative speed actuator/sensor.

The generic Dataltem name is defined as RelSpeed.

The current speed is defined as RelSpeed\_cur.

The set point (wanted) speed is defined as RelSpeed\_sp.

#### E.3.26.1 Dataltem RelSpeed definition

The sensor is defined with the fields defined in Table E.3-35.

**Table E.3-35 — Dataltem RelSpeed definition**

| Field  | Value           | R/A      |
|--|-----------------|----------|
| <u>Name</u>  | <u>RelSpeed</u> | <u>R</u> |
| <u>Type</u>  | uda:float       | <u>R</u> |
| <u>Encoding</u>  | <u>ascii</u>    | <u>R</u> |
| <u>access</u>  | <u>rw</u>       | <u>R</u> |
| <sup>a</sup> For a device this column indicates whether the action shall be implemented or not, where <u>R</u> = required, <u>A</u> = allowed, <u>CR</u> = conditionally required, <u>CA</u> = conditionally allowed, <u>X</u> = Non-standard, add <u>-D</u> when deprecated (e.g., <u>R-D</u> , <u>A-D</u> ). |                 |          |

The RelSpeed is relative and is expressed in a value between [0,100] where 0 means stationary and 100 means maximum speed.

#### E.3.26.2 Dataltem description xml

```
<DataltemDescription
xmlns="urn:schemas-upnp-org:smgt:sdmddid"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="
urn:schemas-upnp-org:smgt:sdmddid.xsd
http://www.upnp.org/schemas/smgt/sdmddid-v1.xsd"
itemname="RelSpeed"
access="rw">
<description>The relative speed</description>
</DataltemDescription>
```

### E.3.27 Dataltem Name Saturation

This Annex describes the dataltem model of the Saturation actuator/sensor.

#### E.3.27.1 Dataltem Saturation definition

The sensor is defined with the fields defined in Table E.3-36.

**Table E.3-36 — Dataltem Saturation definition**

| Field           | Value             | R/A      |
|-----------------|-------------------|----------|
| <u>Name</u>     | <u>Saturation</u> | <u>R</u> |
| <u>Type</u>     | uda:int32         | <u>R</u> |
| <u>Encoding</u> | <u>ascii</u>      | <u>R</u> |
| <u>access</u>   | <u>rw</u>         | <u>R</u> |

## SensorDataModel:1

| Field  | Value | R/A |
|--|-------|-----|
| <sup>a</sup> For a device this column indicates whether the action shall be implemented or not, where <u>R</u> = required, <u>A</u> = allowed, <u>CR</u> = conditionally required, <u>CA</u> = conditionally allowed, <u>X</u> = Non-standard, add <u>-D</u> when deprecated (e.g., <u>R-D</u> , <u>A-D</u> ). |       |     |

The Saturation defines a range from gray (0%) to pure color (100%) at constant lightness level. A pure color is fully saturated.

### E.3.27.2 Dataltem description xml

```
<DataltemDescription
xmlns="urn:schemas-upnp-org:smgt:sdmddid"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="
urn:schemas-upnp-org:smgt:sdmddid.xsd
http://www.upnp.org/schemas/smgt/sdmddid-v1.xsd"
itemname="Saturation"
access="rw">
<description>The color saturation value</description>
</DataltemDescription>
```

### E.3.28 Dataltem Name Speed

This Annex describes the dataltem model of the Speed actuator/sensor.

#### E.3.28.1 Dataltem Speed definition

The sensor is defined with the fields defined in Table E.3-37.

Table E.3-37 — Dataltem Speed definition

| Field  | Value        | R/A      |
|--|--------------|----------|
| <u>Name</u>  | <u>Speed</u> | <u>R</u> |
| <u>Type</u>  | uda:int32    | <u>R</u> |
| <u>Encoding</u>  | <u>ascii</u> | <u>R</u> |
| <u>Access</u>  | <u>rw</u>    | <u>R</u> |
| <sup>a</sup> For a device this column indicates whether the action shall be implemented or not, where <u>R</u> = required, <u>A</u> = allowed, <u>CR</u> = conditionally required, <u>CA</u> = conditionally allowed, <u>X</u> = Non-standard, add <u>-D</u> when deprecated (e.g., <u>R-D</u> , <u>A-D</u> ). |              |          |

The Speed is default expressed in meters per second [m/s].

### E.3.28.2 Dataltem description xml

```
<DataltemDescription
xmlns="urn:schemas-upnp-org:smgt:sdmddid"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="
urn:schemas-upnp-org:smgt:sdmddid.xsd
http://www.upnp.org/schemas/smgt/sdmddid-v1.xsd"
itemname="Speed"
access="rw">
<description>The speed value</description>
</DataltemDescription>
```

### E.3.29 Dataltem Name SpeedDirection

This Annex describes the dataltem model of the speed direction actuator/sensor.

The Generic Dataltem name is defined as SpeedDirection.

## SensorDataModel:1

The current speed direction is defined as SpeedDirection\_cur.

The set point (wanted) speed direction is defined as SpeedDirection\_sp.

### E.3.29.1 Dataltem SpeedDirection definition

The sensor is defined with the fields defined in Table E.3-38.

**Table E.3-38 — Dataltem SpeedDirection definition**

| Field  | Value                 | R/A      |
|--|-----------------------|----------|
| <u>Name</u>  | <u>SpeedDirection</u> | <u>R</u> |
| <u>Type</u>  | uda:string            | <u>R</u> |
| <u>Encoding</u>  | <u>ascii</u>          | <u>R</u> |
| <u>Access</u>  | <u>rw</u>             | <u>R</u> |
| <sup>a</sup> For a device this column indicates whether the action shall be implemented or not, where <u>R</u> = required, <u>A</u> = allowed, <u>CR</u> = conditionally required, <u>CA</u> = conditionally allowed, <u>X</u> = Non-standard, add <u>-D</u> when deprecated (e.g., <u>R-D</u> , <u>A-D</u> ). |                       |          |

The SpeedDirection (for revolving speeds) is defined as "left" and "right".

### E.3.29.2 Dataltem description xml

```
<DataltemDescription
xmlns="urn:schemas-upnp-org:smgt:sdmdid"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="
urn:schemas-upnp-org:smgt:sdmdid.xsd
http://www.upnp.org/schemas/smgt/sdmdid-v1.xsd"
itemname="SpeedDirection"
access="rw">
<description>The Direction of the speed</description>
</DataltemDescription>
```

### E.3.30 Dataltem Name Switch

This Annex describes the dataltem model of the Switch sensor.

#### E.3.30.1 Dataltem Switch definition

The sensor is defined with the fields defined in Table E.3-39.

**Table E.3-39 — Dataltem Switch definition**

| Field  | Value         | R/A      |
|--|---------------|----------|
| <u>Name</u>  | <u>Switch</u> | <u>R</u> |
| <u>Type</u>  | uda:boolean   | <u>R</u> |
| <u>Encoding</u>  | <u>ascii</u>  | <u>R</u> |
| <u>access</u>  | <u>rw</u>     | <u>R</u> |
| <sup>a</sup> For a device this column indicates whether the action shall be implemented or not, where <u>R</u> = required, <u>A</u> = allowed, <u>CR</u> = conditionally required, <u>CA</u> = conditionally allowed, <u>X</u> = Non-standard, add <u>-D</u> when deprecated (e.g., <u>R-D</u> , <u>A-D</u> ). |               |          |

The Switch value "1" means that the device is active, value "0" means that the device is inactive.

### E.3.30.2 Dataltem description xml

```
<DataltemDescription
xmlns="urn:schemas-upnp-org:smgt:sdmdid"
```



### SensorDataModel:1

```
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation=".
urn:schemas-upnp-org:smgt:sdmdid.xsd
http://www.upnp.org/schemas/smgt/sdmdid-v1.xsd"
itemname="Switch"
access="rw">
<description>Generic on/off switch</description>
</DataItemDescription>
```

#### E.3.31 Dataltem Name Temperature

This Annex describes the dataltem model of the Temperature sensor.

The Temperature sensor definition can occur as Temperature, current Temperature and set point Temperatures

The generic Dataltem name is defined as Temperature.

The current temperature is defined as Temperature\_cur.

The set point (wanted) temperature is defined as Temperature\_sp.

##### E.3.31.1 Dataltem Temperature definition

The sensor is defined with the fields defined in Table E.3-40.

**Table E.3-40 — Dataltem Temperature definition**

| Field  | Value              | R/A      |
|--|--------------------|----------|
| <u>Name</u>  | <u>Temperature</u> | <u>R</u> |
| <u>Type</u>  | uda:int32          | <u>R</u> |
| <u>Encoding</u>  | <u>ascii</u>       | <u>R</u> |
| <u>access</u>  | <u>rw</u>          | <u>R</u> |
| <sup>a</sup> For a device this column indicates whether the action shall be implemented or not, where <u>R</u> = required, <u>A</u> = allowed, <u>CR</u> = conditionally required, <u>CA</u> = conditionally allowed, <u>X</u> = Non-standard, add <u>-D</u> when deprecated (e.g., <u>R-D</u> , <u>A-D</u> ). |                    |          |

The Temperature is default expressed in Celcius [C].

##### E.3.31.2 Dataltem description xml

```
<DataItemDescription
xmlns="urn:schemas-upnp-org:smgt:sdmdid"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation=".
urn:schemas-upnp-org:smgt:sdmdid.xsd
http://www.upnp.org/schemas/smgt/sdmdid-v1.xsd"
itemname="Temperature"
access="rw">
<description>The generic, current or set point temperature</description>
</DataItemDescription>
```

#### E.3.32 Dataltem Name Vibration

This Annex describes the dataltem model of the vibration actuator/sensor.

##### E.3.32.1 Dataltem Vibration definition

The sensor is defined with the fields defined in Table E.3-41.

**Table E.3-41 — Dataltem Vibration definition**

| Field  | Value            | R/A      |
|--|------------------|----------|
| <u>Name</u>  | <u>Vibration</u> | <u>R</u> |
| <u>Type</u>  | uda:float        | <u>R</u> |
| <u>Encoding</u>  | <u>ascii</u>     | <u>R</u> |
| <u>Access</u>  | <u>rw</u>        | <u>R</u> |
| <sup>a</sup> For a device this column indicates whether the action shall be implemented or not, where <u>R</u> = required, <u>A</u> = allowed, <u>CR</u> = conditionally required, <u>CA</u> = conditionally allowed, <u>X</u> = Non-standard, add <u>-D</u> when deprecated (e.g., <u>R-D</u> , <u>A-D</u> ). |                  |          |

The Vibration is default expressed in [Hz].

### E.3.32.2 Dataltem description xml

```
<DataltemDescription
xmlns="urn:schemas-upnp-org:smgt:sdmddid"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="
urn:schemas-upnp-org:smgt:sdmddid.xsd
http://www.upnp.org/schemas/smgt/sdmddid-v1.xsd"
itemname="Vibration"
access="rw">
<description>The vibration of a system</description>
</DataltemDescription>
```

### E.3.33 Dataltem Name Voltage

This Annex describes the dataltem model of the alternating voltage actuator/sensor.

The generic Dataltem name is defined as Voltage.

The voltage of the different phases are defined as Voltage\_phX, where X is defined as 1,2 and 3.

#### E.3.33.1 Dataltem Voltage definition

The sensor is defined with the fields defined in Table E.3-42.

**Table E.3-42 — Dataltem Voltage definition**

| Field  | Value          | R/A      |
|--|----------------|----------|
| <u>Name</u>  | <u>Voltage</u> | <u>R</u> |
| <u>Type</u>  | uda:float      | <u>R</u> |
| <u>Encoding</u>  | <u>ascii</u>   | <u>R</u> |
| <u>Access</u>  | <u>rw</u>      | <u>R</u> |
| <sup>a</sup> For a device this column indicates whether the action shall be implemented or not, where <u>R</u> = required, <u>A</u> = allowed, <u>CR</u> = conditionally required, <u>CA</u> = conditionally allowed, <u>X</u> = Non-standard, add <u>-D</u> when deprecated (e.g., <u>R-D</u> , <u>A-D</u> ). |                |          |

The Voltage is default expressed in Volts [v].

### E.3.33.2 Dataltem description xml

```
<DataltemDescription
xmlns="urn:schemas-upnp-org:smgt:sdmddid"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
```

### SensorDataModel:1

```
xsi:schemaLocation="
urn:schemas-upnp-org:smgt:sdmdid.xsd
http://www.upnp.org/schemas/smgt/sdmdid-v1.xsd"
itemname="Voltage"
access="rw">
<description>The voltage of a system</description>
</DataltemDescription>
```

#### E.3.34 Dataltem Name Voltage\_dc

This Annex describes the dataltem model of the direct current voltage actuator/sensor.

##### E.3.34.1 Dataltem Voltage\_dc definition

The sensor is defined with the fields defined in Table E.3-43.

**Table E.3-43 — Dataltem Voltage\_dc definition**

| Field  | Value             | R/A      |
|--|-------------------|----------|
| <u>Name</u>  | <u>Voltage_dc</u> | <u>R</u> |
| <u>Type</u>  | uda:float         | <u>R</u> |
| <u>Encoding</u>  | <u>ascii</u>      | <u>R</u> |
| <u>access</u>  | <u>rw</u>         | <u>R</u> |
| <sup>a</sup> For a device this column indicates whether the action shall be implemented or not, where <u>R</u> = required, <u>A</u> = allowed, <u>CR</u> = conditionally required, <u>CA</u> = conditionally allowed, <u>X</u> = Non-standard, add <u>-D</u> when deprecated (e.g., <u>R-D</u> , <u>A-D</u> ). |                   |          |

The Voltage\_dc is default expressed in Volts [v].

##### E.3.34.2 Dataltem description xml

```
<DataltemDescription
xmlns="urn:schemas-upnp-org:smgt:sdmdid"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="
urn:schemas-upnp-org:smgt:sdmdid.xsd
http://www.upnp.org/schemas/smgt/sdmdid-v1.xsd"
itemname="Voltage_dc"
access="rw">
<description>The dc voltage of a system</description>
</DataltemDescription>
```

#### E.3.35 Dataltem Name Volume

This Annex describes the dataltem model of the volume sensor.

##### E.3.35.1 Dataltem Volume definition

The sensor is defined with the fields defined in Table E.3-44.

**Table E.3-44 — Dataltem Volume definition**

| Field           | Value         | R/A      |
|-----------------|---------------|----------|
| <u>Name</u>     | <u>Volume</u> | <u>R</u> |
| <u>Type</u>     | uda:float     | <u>R</u> |
| <u>Encoding</u> | <u>ascii</u>  | <u>R</u> |

### SensorDataModel:1

| Field   | Value     | R/A      |
|---|-----------|----------|
| <u>access</u>   | <i>ro</i> | <u>R</u> |
| a For a device this column indicates whether the action shall be implemented or not, where <u>R</u> = required, <u>A</u> = allowed, <u>CR</u> = conditionally required, <u>CA</u> = conditionally allowed, <u>X</u> = Non-standard, add <u>-D</u> when deprecated (e.g., <u>R-D</u> , <u>A-D</u> ). |           |          |

The Volume is default expressed in [m<sup>3</sup>].

#### E.3.35.2 Dataltem description xml

```
<DataltemDescription
xmlns="urn:schemas-upnp-org:smgt:sdmdid"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="
urn:schemas-upnp-org:smgt:sdmdid.xsd
http://www.upnp.org/schemas/smgt/sdmdid-v1.xsd"
itemname="volume"
access="ro">
<description>The volume</description>
</DataltemDescription>
```

#### E.3.36 Dataltem Name WRDirection

This Annex describes the dataltem model of an Windrose Direction sensor.

##### E.3.36.1 Dataltem WRDirection definition

The sensor is defined with the fields defined in Table E.3-45.

Table E.3-45 — Dataltem WRDirection definition

| Field   | Value              | R/A      |
|---|--------------------|----------|
| <u>Name</u>   | <u>WRDirection</u> | <u>R</u> |
| <u>Type</u>   | uda:string         | <u>R</u> |
| <u>Encoding</u>   | <u>ascii</u>       | <u>R</u> |
| <u>access</u>   | <i>rw</i>          | <u>R</u> |
| a For a device this column indicates whether the action shall be implemented or not, where <u>R</u> = required, <u>A</u> = allowed, <u>CR</u> = conditionally required, <u>CA</u> = conditionally allowed, <u>X</u> = Non-standard, add <u>-D</u> when deprecated (e.g., <u>R-D</u> , <u>A-D</u> ). |                    |          |

The WRDirection (wind rose direction) is expressed in (starting from North clock wise direction):

"N"

"NNE"

"NE"

"ENE"

"E"

"ESE"

"SE"

"SSE"

"S"

## SensorDataModel:1

"SSW"

"SW"

"WSW"

"W"

"WNW"

"NW"

"NNW"

### E.3.36.2 Dataltem description xml

```
<DataltemDescription
xmlns="urn:schemas-upnp-org:smgt:sdmdid"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation=".
urn:schemas-upnp-org:smgt:sdmdid.xsd
http://www.upnp.org/schemas/smgmt/sdmdid-v1.xsd"
itemname="WRDirection"
access="rw">
<description>Wind Rose direction</description>
</DataltemDescription>
```

### E.3.37 Dataltem Name BatteryState

This Annex describes the dataltem model of the BatteryState sensor.

#### E.3.37.1 Dataltem BatteryState definition

The Dataltem is defined with the fields defined in Table E.3-10.

Table E.3-46 Dataltem BatteryState definition

| Field   | Value               | R/A      |
|---|---------------------|----------|
| <u>Name</u>   | <u>BatteryState</u> | <u>R</u> |
| <u>Type</u>   | uda:int             | <u>R</u> |
| <u>Encoding</u>   | <u>ascii</u>        | <u>R</u> |
| <u>access</u>   | rw                  | <u>R</u> |
| a For a device this column indicates whether the action shall be implemented or not, where <u>R</u> = required, <u>A</u> = allowed, <u>CR</u> = conditionally required, <u>CA</u> = conditionally allowed, <u>X</u> = Non-standard, add <u>-D</u> when deprecated (e.g., <u>R-D</u> , <u>A-D</u> ). |                     |          |

The Batterystate is expressed in percentage, where 0 means empty battery and 100 means full battery.

### E.3.37.2 Dataltem description xml

```
<DataItemDescription
xmlns="urn:schemas-upnp-org:smgt:sdmdid"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation=".
urn:schemas-upnp-org:smgt:sdmdid.xsd
http://www.upnp.org/schemas/smgmt/sdmdid-v1.xsd"
itemname="BatteryState"
access="rw">
```

## SensorDataModel:1

```
<description>The Battery State</description>
</DataItemDescription>
```

### E.3.38 Dataltem Name CleanerState

This Annex describes the dataltem model of the Cleaner actuator/sensor.

#### E.3.38.1 Dataltem CleanerState definition

The Dataltem is defined with the fields defined in Table E.3-10.

**Table E.3-47 Dataltem CleanerState definition**

| Field  | Value               | R/A      |
|--|---------------------|----------|
| <u>Name</u>  | <u>CleanerState</u> | <u>R</u> |
| <u>Type</u>  | uda:string          | <u>R</u> |
| <u>Encoding</u>  | <u>ascii</u>        | <u>R</u> |
| <u>access</u>  | <u>rw</u>           | <u>R</u> |
| <sup>a</sup> For a device this column indicates whether the action shall be implemented or not, where <u>R</u> = required, <u>A</u> = allowed, <u>CR</u> = conditionally required, <u>CA</u> = conditionally allowed, <u>X</u> = Non-standard, add <u>-D</u> when deprecated (e.g., <u>R-D</u> , <u>A-D</u> ). |                     |          |

The CleanerState is expressed as “Start”, ”Homing”, “Cleaning”, “Charging”, “Completed”.

#### E.3.38.2 Dataltem description xml

```
<DataItemDescription
xmlns="urn:schemas-upnp-org:smgt:sdmddid"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="
urn:schemas-upnp-org:smgt:sdmddid.xsd
http://www.upnp.org/schemas/smgt/sdmddid-v1.xsd"
itemname="CleanerState"
access="rw">
<description>The Cleaner State</description>
</DataItemDescription>
```

### E.3.39 Dataltem Name CloseToDevice

This Annex describes the dataltem model of the close to another device location indication.

This value describes if this device is close to another device. This other device might have a known location.

#### E.3.39.1 Dataltem CloseToDevice definition

The Dataltem is defined with the fields defined in Table E.3-10.

**Table E.3-48 Dataltem CloseToDevice definition**

| Field  | Value                | R/A      |
|--|----------------------|----------|
| <u>Name</u>  | <u>CloseToDevice</u> | <u>R</u> |
| <u>Type</u>  | uda:string           | <u>R</u> |
| <u>Encoding</u>  | <u>ascii</u>         | <u>R</u> |
| <u>access</u>  | <u>rw</u>            | <u>R</u> |
| <sup>a</sup> For a device this column indicates whether the action shall be implemented or not, where <u>R</u> = required, <u>A</u> = allowed, <u>CR</u> = conditionally required, <u>CA</u> = conditionally allowed, <u>X</u> = Non-standard, add <u>-D</u> when deprecated (e.g., <u>R-D</u> , <u>A-D</u> ). |                      |          |

The CloseToDevice indicates another device id.

### E.3.39.2 Dataltem description xml

```

<DataItemDescription
xmlns="urn:schemas-upnp-org:smgt:sdmddid"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="
urn:schemas-upnp-org:smgt:sdmddid.xsd
http://www.upnp.org/schemas/smgt/sdmddid-v1.xsd"
itemname="CloseToDevice"
access="rw">
<description>The ID of another device that is close to this one
</description>
</DataItemDescription>

```

### E.3.40 Dataltem Name DryingState

This Annex describes the dataltem model of the Lock actuator/sensor.

#### E.3.40.1 Dataltem DryingState definition

The Dataltem is defined with the fields defined in Table E.3-10.

**Table E.3-49 Dataltem DryingState definition**

| Field  | Value              | R/A      |
|--|--------------------|----------|
| <u>Name</u>  | <u>DryingState</u> | <u>R</u> |
| <u>Type</u>  | uda:string         | <u>R</u> |
| <u>Encoding</u>  | <u>ascii</u>       | <u>R</u> |
| <u>access</u>  | <u>rw</u>          | <u>R</u> |
| <sup>a</sup> For a device this column indicates whether the action shall be implemented or not, where <u>R</u> = required, <u>A</u> = allowed, <u>CR</u> = conditionally required, <u>CA</u> = conditionally allowed, <u>X</u> = Non-standard, add <u>-D</u> when deprecated (e.g., <u>R-D</u> , <u>A-D</u> ). |                    |          |

The DryingState is default expressed as “Idle”, “Drying”, “Completed”.

#### E.3.40.2 Dataltem description xml

```

<DataItemDescription
xmlns="urn:schemas-upnp-org:smgt:sdmddid"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="
urn:schemas-upnp-org:smgt:sdmddid.xsd
http://www.upnp.org/schemas/smgt/sdmddid-v1.xsd"

```

## SensorDataModel:1

```
itemname=" DryingState"
access="rw">
<description>The Drying state</description>
</DataItemDescription>
```

### E.3.41 Dataltem Name Location

This Annex describes the dataltem model of the Location indication.

This value describes the location of an device as dataltem. The syntax is the same as described in Annex F.

#### E.3.41.1 Dataltem Location definition

The Dataltem is defined with the fields defined in Table E.3-10.

Table E.3-50 Dataltem Location definition

| Field  | Value           | R/A      |
|--|-----------------|----------|
| <u>Name</u>  | <u>Location</u> | <u>R</u> |
| <u>Type</u>  | uda:string      | <u>R</u> |
| <u>Encoding</u>  | <u>ascii</u>    | <u>R</u> |
| <u>access</u>  | rw              | <u>R</u> |
| <sup>a</sup> For a device this column indicates whether the action shall be implemented or not, where <u>R</u> = required, <u>A</u> = allowed, <u>CR</u> = conditionally required, <u>CA</u> = conditionally allowed, <u>X</u> = Non-standard, add <u>-D</u> when deprecated (e.g., <u>R-D</u> , <u>A-D</u> ). |                 |          |

The Location indicates the location of the device.

#### E.3.41.2 Dataltem description xml

```
<DataItemDescription
xmlns="urn:schemas-upnp-org:smgt:sdmdid"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="
urn:schemas-upnp-org:smgt:sdmdid.xsd
http://www.upnp.org/schemas/smgt/sdmdid-v1.xsd"
itemname=" Location"
access="rw">
<description>The Location of the device</description>
</DataItemDescription>
```

### E.3.42 Dataltem Name Lock

This Annex describes the dataltem model of the Lock actuator/sensor.

#### E.3.42.1 Dataltem Lock definition

The Dataltem is defined with the fields defined in Table E.3-10.

The Dataltem name can be prefixed with:

- "Door"
- "Window"
- Device types that can be locked, full list is defined in Annex C. Examples:
  - Dishwasher



### SensorDataModel:1

- Washing machine
- Dryer
- Microwave

**Table E.3-51 Dataltem Lock definition**

| Field  | Value        | R/A      |
|--|--------------|----------|
| <u>Name</u>  | <u>Lock</u>  | <u>R</u> |
| <u>Type</u>  | uda:string   | <u>R</u> |
| <u>Encoding</u>  | <u>ascii</u> | <u>R</u> |
| <u>access</u>  | <u>rw</u>    | <u>R</u> |
| <sup>a</sup> For a device this column indicates whether the action shall be implemented or not, where <u>R</u> = required, <u>A</u> = allowed, <u>CR</u> = conditionally required, <u>CA</u> = conditionally allowed, <u>X</u> = Non-standard, add <u>-D</u> when deprecated (e.g., <u>R-D</u> , <u>A-D</u> ). |              |          |

The Lock is default expressed as “locked”, “unlocked”.

#### **E.3.42.2 Dataltem description xml**

```
<DataItemDescription
xmlns="urn:schemas-upnp-org:smgt:sdmdid"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="
urn:schemas-upnp-org:smgt:sdmdid.xsd
http://www.upnp.org/schemas/smgmt/sdmdid-v1.xsd"
itemname="Lock"
access="rw">
<description>The Lock</description>
</DataItemDescription>
```

#### **E.3.43 Dataltem Name MagnetoMeter**

This Annex describes the dataltem model of the Lock actuator/sensor.

##### **E.3.43.1 Dataltem MagnetoMeter definition**

The Dataltem is defined with the fields defined in Table E.3-10.

**Table E.3-52 Dataltem MagnetoMeter definition**

| Field  | Value               | R/A      |
|--|---------------------|----------|
| <u>Name</u>  | <u>MagnetoMeter</u> | <u>R</u> |
| <u>Type</u>  | uda:float           | <u>R</u> |
| <u>Encoding</u>  | <u>ascii</u>        | <u>R</u> |
| <u>access</u>  | <u>rw</u>           | <u>R</u> |
| <sup>a</sup> For a device this column indicates whether the action shall be implemented or not, where <u>R</u> = required, <u>A</u> = allowed, <u>CR</u> = conditionally required, <u>CA</u> = conditionally allowed, <u>X</u> = Non-standard, add <u>-D</u> when deprecated (e.g., <u>R-D</u> , <u>A-D</u> ). |                     |          |

The MagnetoMeter is default expressed in Tesla.

### E.3.43.2 Dataltem description xml

```
<DataItemDescription
xmlns="urn:schemas-upnp-org:smgt:sdmdid"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="
urn:schemas-upnp-org:smgt:sdmdid.xsd
http://www.upnp.org/schemas/smgmt/sdmdid-v1.xsd"
itemname=" MagnetoMeter"
access="rw">
<description>The MagnetoMeter</description>
</DataItemDescription>
```

### E.3.44 Dataltem Name WashProgram

This Annex describes the dataltem model of the WashProgram actuator/sensor.

#### E.3.44.1 Dataltem WashProgram definition

The Dataltem is defined with the fields defined in Table E.3-10.

The WashProgram can be used in a WashingMachine or Dishwasher.

Note that not all states are applicable for a Dishwasher.

**Table E.3-53 Dataltem WashProgram definition**

| Field   | Value              | R/A      |
|---|--------------------|----------|
| <u>Name</u>   | <u>WashProgram</u> | <u>R</u> |
| <u>Type</u>   | uda:float          | <u>R</u> |
| <u>Encoding</u>   | <u>ascii</u>       | <u>R</u> |
| <u>access</u>   | <u>rw</u>          | <u>R</u> |
| a For a device this column indicates whether the action shall be implemented or not, where <u>R</u> = required, <u>A</u> = allowed, <u>CR</u> = conditionally required, <u>CA</u> = conditionally allowed, <u>X</u> = Non-standard, add <u>-D</u> when deprecated (e.g., <u>R-D</u> , <u>A-D</u> ). |                    |          |

The WashProgram is expressed as “PreWash”, “FastWash”, “NormalWash”, “IntensiveWash”, “Rinse”, “Spin”, “Rinse&Spin”, “White”, “Color”, “Fine”, “Cotton”, “Wool”, “Synthetic” “PumpOut”. The WashProgram definitions can be extended by vendors. The vendor defined states shall start with “X\_”.

### E.3.44.2 Dataltem description xml

```
<DataItemDescription
xmlns="urn:schemas-upnp-org:smgt:sdmdid"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="
urn:schemas-upnp-org:smgt:sdmdid.xsd
http://www.upnp.org/schemas/smgmt/sdmdid-v1.xsd"
itemname=" WashProgram"
access="rw">
<description>The Wash program</description>
</DataItemDescription>
```

### E.3.45 Dataltem Name WashState

This Annex describes the dataltem model of the WashState actuator/sensor.

#### E.3.45.1 Dataltem WashState definition

The Dataltem is defined with the fields defined in Table E.3-10.

The WashState can be used in a WashingMachine or Dishwasher.

Note that the “Spinning” state in a Dishwasher can be omitted.

**Table E.3-54 Dataltem WashState definition**

| Field  | Value            | R/A      |
|--|------------------|----------|
| <u>Name</u>  | <u>WashState</u> | <u>R</u> |
| <u>Type</u>  | uda:float        | <u>R</u> |
| <u>Encoding</u>  | <u>ascii</u>     | <u>R</u> |
| <u>access</u>  | <u>rw</u>        | <u>R</u> |
| <sup>a</sup> For a device this column indicates whether the action shall be implemented or not, where <u>R</u> = required, <u>A</u> = allowed, <u>CR</u> = conditionally required, <u>CA</u> = conditionally allowed, <u>X</u> = Non-standard, add <u>-D</u> when deprecated (e.g., <u>R-D</u> , <u>A-D</u> ). |                  |          |

The WashState is expressed as “Idle”, ”Washing”, “Rinsing”, “Spinning”, “Drying”, “Completed”.

#### E.3.45.2 Dataltem description xml

```
<DataItemDescription
xmlns="urn:schemas-upnp-org:smgt:sdmdid"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="
urn:schemas-upnp-org:smgt:sdmdid.xsd
http://www.upnp.org/schemas/smgt/sdmdid-v1.xsd"
itemname="WashState"
access="rw">
<description>The WashState</description>
</DataItemDescription>
```

## Annex F Location

This Annex describes the EBNF of the location.

The location can be used to identify where a sensor/actuator is located

Currently there are 2 mechanisms defining the location of a Device; the prefix of the Dataltem and the Location as a Dataltem.

The mechanism to prefix the data items is described in [10]– section 4.3.2]. The EBNF of the location contains characters to distinguish between free text and the EBNF location. When using the EBNF notation the restricted characters specified in [10]– section 4.3.2] are allowed.

The EBNF location syntax can also be used in a Location Dataltem. This Dataltem is specified to indicate the location of a Device by means of the syntax of F.1.

### F.1 Location syntax

|                  |     |  |                                    |
|------------------|-----|--|------------------------------------|
| Location         | ::= | buildinglocation<br>vehicle<br>gpslocation<br>freetext   | <br> <br> <br>                     |
| buildinglocation | ::= | '#' [site_][buildinglist '_' ] roomlist<br>[('_')absinteger][('_')relpos]  |                                    |
| vehicle          | ::= | '?'vehiclelist   |                                    |
| gpslocation      | ::= | %dms' dmscoord   '%gps' gpscoord   |                                    |
| freetext         | ::= | (a-z,A-Z)+<br>(* anyfree text *)   |                                    |
| dmscoord         | ::= | 'lat:'deg','min','sec',lon:'deg','min','sec  |                                    |
| gpscoord         | ::= | 'lat:'float', lon:'float[' ,alt:'float]<br>(* altitude in meters above sealevel*)  |                                    |
| relpos           | ::= | '{'float','float','float'}'<br>(* relative postion in the room, defined in abstract<br>box of [-1,-1,-1],[1,1,1], [width,length,height] where<br>[0,0,0] is the center of the room, the shortest wall of<br>the room is the width of the room *) |                                    |
| deg              | ::= | integer  |                                    |
| min              | ::= | integer  |                                    |
| sec              | ::= | integer  |                                    |
| float            | ::= | [-](1-9)[(0-9)+][.(0-9)+]  |                                    |
| integer          | ::= | [-](1-9)[(0-9)+]   |                                    |
| absinteger       | ::= | (1-9)[(0-9)+]  |                                    |
| site             | ::= | 'campus'<br>'city'<br>'technologypark'<br>'town'   | <br> <br> <br>                     |
| buildinglist     | ::= | 'airport'<br>'arena'<br>'bar'<br>'bar'<br>'barn'<br>'brewery'<br>'business'<br>'busstation'<br>'cafe'  | <br> <br> <br> <br> <br> <br> <br> |

## SensorDataModel:1

```
'chalet'  
'church'  
'club'  
'conferencecenter'  
'court'  
'gasstation'  
'gym'  
'factory'  
'ferryterminal'  
'firestation'  
'home'  
'holidayhome'  
'hospital'  
'hotel'  
'mall'  
'monastery'  
'mosque'  
'motel'  
'museum'  
'office'  
'operahouse'  
'parkinggarage'  
'policestation'  
'port'  
'prison'  
'railwaystation'  
'restaurant'  
'school'  
'shop'  
'stadium'  
'supermarket'  
'synagogue'  
'teahouse'  
'temple'  
'terminal'  
'theater'  
'townhall'  
'treehouse'  
'winery'
```

```
roomlist ::= 'attic'  
'balcony'  
'ballroom'  
'bathroom'  
'bedroom'  
'border'  
'boxroom'  
'cellar'  
'cloakroom'  
'conservatory'  
'corridor'  
'deck'  
'den'  
'diningroom'  
'drawingroom'  
'driveway'  
'dungeon'  
'ensuite'  
'entrance'  
'familyroom'  
'garage'  
'garden'  
'guestroom'  
'hall'  
'kitchen'
```

## SensorDataModel:1

```
'larder'  
'lawn'  
'library'  
'livingroom'  
'lounge'  
'mancafe'  
'masterbedroom'  
'musicroom'  
'office'  
'pantry'  
'parkinglot'  
'parlour'  
'patio'  
'receptionroom'  
'roof'  
'roofterrace'  
'sauna'  
'shed'  
'sittingroom'  
'snug'  
'spa'  
'studio'  
'suite'  
'swimmingpool'  
'toilet'  
'utilityroom'  
'ward'  
'vegetableplot'  
'terrace'  
'utilityroom'  
  
vehiclelist ::= 'airplane'  
                'bicycle'  
                'boat'  
                'bus'  
                'car'  
                'cruiseship'  
                'ferry'  
                'hoverboard'  
                'motorcycle'  
                'motorhome'  
                'ship'  
                'spaceship'  
                'train'  
                'yacht'
```

## F.2 Example of standard names

#bathroom

#bedroom\_2

#livingroom\_{0,0.5,1}

%gpslat:-1.3535400 ,lon:50.9684900

%dmslat:37,15 ,47,lon:115,47,34