Mapping LWM2M model to CoMI YANG

draft-vanderstok-core-yang-LWM2M-00

Peter van der Stok
Jaime Jiménez

(Work in Progress)

Purpose

- Standard organisations use hierarchical models that can be specified in XML and describe classes with attributes and operations that can be instantiated to objects.

- OMA LWM2M and IPSO standardise numbered object types.

- CoMI at IETF (draft-vanderstok-core-comi-09) describes a network management interface based on CoAP and YANG.

- Goal: convert a LWM2M xml-based device specification to a YANG MODULE for CoMI consumption.

Humidity Object

Object definition

Name	Object ID	Instances	Mandatory	Object URN
Humidity	0	Multiple	Mandatory	urn:oma:lwm2m:ipso:3304

Resource definitions

ID	Name	Operations	Instances	Mandatory	Type	Units	Description
5700	Sensor Value	R	Single	Mandatory	Float	…	…
5601	Min Measured Value	R	Single	Optional	Float	…	…
5602	Max Measured Value	R	Single	Optional	Float	…	…
5603	Min Range Value	R	Single	Optional	Float	…	…
5604	Max Range Value	R	Single	Optional	Float	…	…
5701	Sensor Units	R	Single	Optional	String	…	…
5605	Reset Min and Max	E	Single	Optional	Opaque	…	…

Conversion Rules
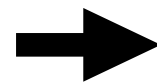
LWM2M	➡	YANG (RFC6020)
optional /mandatory attribute		false / true statement
R, W attributes		Config statement (False=R, True=W)
E attribute		YANG RPC
range attribute		range statement
units		units statement

Resources		Leafs on a YANG list
		`+--ro ID3301* [instance_number]`
		`+--ro 5700 uint16`

Object Instance		"instance" key attribute

URI Conversion
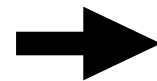
LWM2M	➡	YANG
		RESTCONF URI (example 3): http://example.com/type/instance/resource
URI: coap+lwm2m://example.com/object/instance/resource		CoMI URI (example 3): coap://example.com/type/resource**?key=0** if only one instance then coap://example.com/type/resource

- Keys as query parameter for instance number.

Generated YANG modules

[ ]   list keys

rw   configuration data (read and write)

ro   state data (read only)

?   optional node

*   list and leaf list

( )   choice

:   case nodes

...   subtrees not shown

```
1.  module: ietf-yang-humidityID
         +--ro ID3301* [instance_number]
            +--ro instance_number    uint16
            +--ro ID5700              decimal64
            +--ro ID5701?             string
            +--ro ID5601?             decimal64
            +--ro ID5602?             decimal64
            +--ro ID5603?             decimal64
            +--ro ID5604?             decimal64
            +---x ID5605
```

```
2.  module: ietf-yang-humidityNM
         +--ro IPSO-humidity* [instance_number]
            +--ro instance_number            uint16
            +--ro Sensor_Value               decimal64
            +--ro Units?                     string
            +--ro Min_Measured_Value?        decimal64
            +--ro Max_Measured_Value?        decimal64
            +--ro Min_Range_Value?           decimal64
            +--ro Max_Range_Value?           decimal64
            +---x Reset_Min_and_Max_measured_values
```

Generated YANG modules

[ ]   list keys

rw   configuration data (read and write)

ro   state data (read only)

?   optional node

*****   list and leaf list

( )   choice

:   case nodes

...   subtrees not shown

```
3. module: ietf-yang-humidityLF
        +--rw IPSO-humidity
           +--ro identifier     uint16
           +--ro resources* [instance_number]
              +--ro instance_number  uint16
              +--ro Sensor_Value
              |  +--ro identifier?    uint16
              |  +--ro content        decimal64
              +--ro Units
              |  +--ro identifier?    uint16
              |  +--ro content?       string
              +--ro Min_Measured_Value
              |  +--ro identifier?    uint16
              |  +--ro content?       decimal64
              +--ro Max_Measured_Value
              |  +--ro identifier?    uint16
              |  +--ro content?       decimal64
              +--ro Min_Range_Value
              |  +--ro identifier?    uint16
              |  +--ro content?       decimal64
              +--ro Max_Range_Value
              |  +--ro identifier?    uint16
              |  +--ro content?       decimal64
              +--ro Reset_Min_and_Max_measured_values
                 +--ro identifier?    uint16
                 +---x reset
```

Takeaways

- Example 1 (module: ietf-yang-humidityID) is a bit forced and lacks the Resource Name.

- Example 2 (module: ietf-yang-humidityNM) seems to be the best fit.

- Example 3 (ietf-yang-humidityLF) seems too complex.

- YANG is much more expressive than LWM2M, there are many ways to express the same thing.

- Both .XML (3482 characters) and .YANG (4570 characters on example 1) have a lot of "noise" in them.

- Key leafs are just one possible way to represent instances.

- Access Control mapping might be done better.

- YANG has no Float, we use 64 bit precision (float is 32).

- Need to script automatic conversion.

- Where would a converter run? GWs, devices, server?

Links

- https://tools.ietf.org/html/rfc6020

- http://technical.openmobilealliance.org/Technical/technical-information/release-program/current-releases/oma-lightweightm2m-v1-0

- http://ipso-alliance.github.io/pub/

- (Preliminary work) http://jaimejim.github.io/drafts/draft-vanderstok-core-yang-lwm2m-00.txt

- jaimejim.github.io/drafts/3304.xml

- jaimejim.github.io/drafts/3304.yang

3304.yang

```
module ietf-humidityID{

  yang-version 1.1;   // needed for action

  namespace
      "urn:ietf:params:xml:ns:yang:ietf-humidityID";

  prefix humid;

  organization
    "IPSO";

  contact
    "WG Web:   http://tools.ietf.org/wg/core/
    WG List:  mailto:core@ietf.org

    WG Chair: Carsten Bormann
              mailto:cabo@tzi.org

    WG Chair: Jaime Jimenez
              mailto:jaime.jimenez@ericsson.com

    Editor:   Peter van der Stok
              mailto:consultancy@vanderstok.org

    Editor:   Jaime Jimenez
              mailto:jaime.jimenez@ericsson.com";

  description
    "This module contains information about the operation of the IPSO LWM2M humidity sensor with ID 3301.

    Copyright (c) 2016 IETF Trust and the persons identified as
    authors of the code.  All rights reserved.

    Redistribution and use in source and binary forms, with or

    without modification, is permitted pursuant to, and subject
    to the license terms contained in, the Simplified BSD License
    set forth in Section 4.c of the IETF Trust's Legal Provisions
    Relating to IETF Documents
    (http://trustee.ietf.org/license-info).

    This version of this YANG module is part of RFC XXXX; see
    the RFC itself for full legal notices.";
```

```
revision "2016-07-25" {
    description "Initial revision.";
    reference
      "I-D:draft-vanderstok-core-yang-LWM2M: YANG language applied to the LWM2M IPSO humidity sensor specification";
}


list ID3301 {
  key instance_number;
  config  false;     // should be same for key leaf
  description
    "IPSO humidity: The humidity sensor is composed of a set of instances";
      leaf instance_number {
          type uint16{
              range "0..1";   // only one instance zero (0)
          }
          config false;     // R access
          mandatory "true";
          description
           "the number of the humidity sensor instance";
      }
      leaf ID5700 {
          type decimal64{     // YANG has no float
                fraction-digits 2;
                range "10.0 .. 66.6";}
          config false;     // R access
          mandatory "true";
          description
           "Sensor Value: Last or Current Measured Value from the Sensor";
      }
      leaf ID5701 {
          type string;
          units "Defined by 'Units' resource";
          config false;     // R access
          description
           "Units: Measurement unit definition e.g. 'Cel' for temperature in Celsius";
      }
```

```
leaf ID5602 {
    type decimal64{     // YANG has no float
            fraction-digits 2;
            range "10.0 .. 66.6";}
    units "Defined by 'Units' resource";
    config false;     // R access
    description
      "Max Measured Value: The maximum value measured by the sensor since power ON or reset";
}
leaf ID5603 {
    type decimal64{     // YANG has no float
            fraction-digits 2;
            range "10.0 .. 66.6";}
    units "Defined by 'Units' resource";
    config false;     // R access
    description
      "Min Range Value: The minimum value that can be measured by the sensor";
}
leaf ID5604 {
    type decimal64{     // YANG has no float
            fraction-digits 2;
            range "10.0 .. 66.6";}
    units "Defined by 'Units' resource";
    config false;     // R access
    description
      "Max Range Value: The maximum value that can be measured by the sensor";
}
action ID5605 {
    //E access: this is an RPC without input and output parameters
    description
      "Reset Min and Max measured values: Reset the Min and Max measured values to current value";
}
    }  // list ID3301
} // module ietf-yang-humidity
```