The Schema Editor of OpenIoT for Semantic Sensor Networks

Prem P. Jayaraman, Jean-Paul Calbimonte and Hoan Nguyen Mau Quoc

RMIT University, LSIR EPFL, Insight – NUI Galway

Bethlehem, PA, October 2015
Established Open-source platform for IoT
https://github.com/OpenIoTOrg/openiot

Use cases and validation scenarios

- Integrate sensors & things with cloud computing
- Configure, deploy and use IoT services
- Auditing/assessing privacy of IoT apps in the cloud
- Semantic annotations of internet-connected objects
- Energy-efficient data harvesting
- Publish/subscribe for continuous processing and sensor data filtering
- Mobility of sensors and QoS aspects in IoT
The OpenIoT Architecture

Applications

Semantic data management

Sensor data management
OpenIoT Services and Components

0. Setup and Management
1. End User Request
2. Discovery Services
3. Query Content
4. Sensor Configuration
5. Collect Content / Mobile
6. Content Adaptation
7. Utility Service Feedback
8. Service Delivery
9. Service Visualisation
10. Get Visualisation
11. Data Presentation
Request Definition & Presentation

Everything nice if your data is also nice
Request Definition & Presentation

Data at this level is already RDF-ized

Generated SPARQL queries
OpenIoT and the SSN Ontology

Where to look for vocabs?

When do we set up the onto?

Who sets it up?


Generated URIs? Vocabulary? Literals?
Register metadata

Do I have control over my sensor metadata?

If ontologies change...

If we add new types of sensors?
Virtual Sensor configuration

```xml
<virtual-sensor name="room-monitor" >
  <addressing>
    <predicate key="geographical">BC143</predicate>
    <predicate key="usage">room monitoring</predicate>
  </addressing>
  <life-cycle pool size="10" />
  <output-structure>
    <field name="image" type="binary:image/jpeg" />
    <field name="temp" type="int" />
  </output-structure>
  <storage permanent="true" history-size="10h" />
  <input-streams>
    <input-stream name="cam">
      <stream-source alias="cam" storage-size="1m" sampling-rate="1">
        <address wrapper="tinyos2.x">
          <predicate key="host">tinybox.epfl.ch</predicate>
        </address>
        select * from WRAPPER
      </stream-source>
    </input-stream>
    <input-stream name="temperature1">
      <stream-source alias="temperature1" storage-size="1m" sampling-rate="1">
        <address wrapper="remote">
          <predicate key="type">temperature</predicate>
          <predicate key="geographical">BC143-N</predicate>
        </address>
        select AVG(temp1) as T1 from WRAPPER
      </stream-source>
    </input-stream>
    <input-stream name="temperature2">
      <stream-source alias="temperature2" storage-size="1m" sampling-rate="1">
        <address wrapper="remote">
          <predicate key="type">temperature</predicate>
          <predicate key="geographical">BC143-S</predicate>
        </address>
        select AVG(temp2) as T2 from WRAPPER
      </stream-source>
    </input-stream>
  </input-streams>
  <query>
    select cam.picture as image, temperature1.T1 as temp from cam, temperature1
    where temperature1.T1 > 30 AND temperature1.T1 = temperature2.T2
  </query>
</virtual-sensor>
```
Metadata properties

sensorID="http://lsm.deri.ie/resource/1099207032411018"
sensorName=closedsense
source="Some source"
sourceType=lausanne
sensorType=lausanne
information=Air Quality Sensors from Lausanne station 1
author=opensense

feature="http://lsm.deri.ie/OpenIoT/opensensefeature"
fields="humidity,temperature"
field.temperature.propertyName="http://lsm.deri.ie/OpenIoT/Temperture"
field.temperature.unit=C
field.humidity.propertyName="http://lsm.deri.ie/OpenIoT/Humidity"
field.humidity.unit=Percent
field.co.propertyName="http://lsm.deri.ie/OpenIoT/CO"
field.co.unit=PPM
latitude=46.529838
longitude=6.596818
Turtle RDF registration

```
<sensor/5010> rdf:type aws:CapacitiveBead, ssn:Sensor;
   rdfs:label "Sensor 5010";
   ssn:observes aws:air_temperature;
   phenonet:hasSerialNumber
<sensor/5010/serial/serial2>;
   ssn:onPlatform <site/narrabri/Pweather>;
   ssn:ofFeature <site/narrabri/sf/sf_narrabri>;
   ssn:hasMeasurementProperty
<sensor/5010/accuracy/acc_1>;
   prov:wasGeneratedBy "AuthorName";
   DUL:hasLocation <place/location1>;
   lsm:hasSensorType <sensorType1>;
   lsm:hasSourceType "SourceType".

<sensorType1> rdfs:label "TypeName".

<sensor/5010/serial/serial2> rdf:type phenonet:SerialNumber;
   phenonet:hasId "5010".
```
Editing Ontologies?

Standard Tools
Better Suited for ontologists
Complex for small tasks
No integration with IoT platforms
Generate RDF instances?
OpenIoT Schema Editor

Existing Sensor Types

Observed properties

Users exposed to URIs as identifiers
Sensor Types

- ssn:Sensor
  - rdf:type openiot:WeatherStation
    - ssn:observes cf:Humidity
    - ssn:observes cf:AirTemperature
  - rdfs:subClassOf ssn:MeasurementCapability
    - ssn:hasMeasurementProperty
      - openiot:resource /Capability1
  - rdf:type ssn:Frequency
    - openiot:resource /Frequency1
  - ssn:forProperty
    - openiot:resource /Accuracy1
    - ssn:Accuracy
Schema Editor: Sensor Types

New Type

Observed properties

Measurement Capabilities

Generated RDF
Sensor Instances

foaf:David
geo:_Area1
geo:lon
geo:lat
"-6.23"^^xsd:double
"53.427"^^xsd:double
dul:hasLocation
geo:_DublinAirport
openiot:Resource/sensor/demo_weatherstation
openiot:WeatherStation
openiot:hasName
openiot:Resource/observation/AirTemperature_1_23_2015_06_14_13_30
openiot:Resource/observation
"demo-weatherstation"
ssn:featureOfInterest
ssn:observedBy
ssn:Observation
ssn:ObservationResult
"23.5"^^xsd:double
su:numericValue
qu:unit
unit:F
rdfs:subClassOf
DUL:UnitOfMeasure
rdfs:type
DUL:UnitOfMeasure
"2015-06-14T13:30"^^xsd:dateTime
"2015-06-14T13:30"^^xsd:dateTime
"2015-06-14T13:30"^^xsd:dateTime
"2015-06-14T13:30"^^xsd:dateTime
Schema Editor: New Instances

Sensor Instance Creator

- Sensor Name: WeatherStationCSIRO
- Author/Owner: CSIRO
- Description/Information: Weather Station in Canberra
- Source URL: http://localhost:22001
- Source Type: GSN
- Latitude: -35.280266659925426
- Longitude: 149.12370085628936
- Feature of Interest: field

Observed Properties

- http://services.openiot.eu/AirTemperature
  - Unit of Measurement: °C
  - XGSN Field Name: temp

- http://services.openiot.eu/Humidity
  - Unit of Measurement: percent
  - XGSN Field Name: humidity

Register Sensor Instance
OpenIoT Schema Editor

Based on Standards
Facilitates Extensions
Integrated with OpenIoT
Sensor Types and Instances
Web User Interface

Extensible Editor?
URI generation?
Link to Vocab Libraries?
Validation of content?
Beyond OpenIoT-only?

Schema Editor based on SSN Ontology
OpenIoT resources

OpenIoT Github
https://github.com/OpenIoTOrg/openiot

OpenIoT VDK virtual Machine
https://github.com/OpenIoTOrg/openiot/wiki/VDKv2---OpenIoT-Release

Viedo Demos:
http://www.youtube.com/user/OpenIoT
Muchas gracias!

@jpcik

Jean-Paul Calbimonte
LSIR EPFL