4 - Towards Semantic IoT, oneM2M Base Ontology

Dr. Mahdi Ben Alaya
Founder & CEO, Sensinov
benalaya@sensinov.com
www.sensinov.com

November 14, 2016
Do we really need semantic?

- oneM2M Release-1 ensure interoperability at the level of communications.
- Data is treated as black boxes. The content is opaque and applications have to a-priori know how to interpret the data.
- The consumer is programmed or configured for certain consumers. No data interoperability.
Beforehand agreement required

• It is required to learn information model of each device before using it.
• Beforehand agreement on the data representation is needed between applications and devices.
• Hard to integrate and deal with existing legacy devices.
• Can work in small and closed environment. But does not scale!
Can XML/JSON do the job?

• Human can understand XML-Documents.
  • Intuitively clear for human.
  • Tag names provide semantic meaning since they are domain-terms.

• Machines do not have intuition.
  • Tag names do not provide semantics for machines.
  • XML defines the structure and lacks of semantic model.

```
<measurement>
  <device>TEMP-AZ1299B</device>
  <value>17</value>
  <unit>C</unit>
  <time>20160116T192030</time>
</measurement>
```
Semantic gap between machines

- Which words shall we use to describe a given set of concepts?
- A common vocabulary is required for IoT to bridge the semantic gap between machines.
- Semantic technologies must be used to solve these issues.
From data to decision

- Collecting data is not sufficient, only your ability to convert data into decisions that gives you the edge.

Source: Curt Swindoll, Persuant 2011
Levels of meaningfulness

• There is not just one single level of semantics that could be attached to a raw data element.
• Different levels of meaningfulness can be identified to describe data and device descriptions.
The cost of semantic clarity

- Ontologies provide the highest level of semantic clarity however they are costly in terms of time and money.
- Is it reasonable to use ontologies?

Source: Mike Pergman
The cost of data integration

- Ontology-driven approaches provides a lower costs when dealing with high number of data sources.
- It ensure interoperability for open and big environments.

Source: Price Waterhouse Cooper
Semantic web and ontologies

• “The Semantic Web is an extension of the current web in which information is given well-defined meaning, better enabling computers and people to work in co-operation.”
  - Tim Berners-Lee et al, 2001

• The term ontology is originated from philosophy. It is a formal specification of a domain including concepts and their relationships, attributes and some logical restrictions.

• Example:
Semantic web building blocs

- **URI/IRI**: Almost everything is a URI.
- **RDFS/OWL**: Describe taxonomies and classification networks.
- **SPARQL**: Ontology querying: Select, Update, Construct, etc.)
Semantic IoT vs Semantic Web

• Semantic Web:
  – Relatively static content.
  – E.g. Semantic Wikipedia (dbpedia), annotated pages, etc.

• Seamantic IoT
  – Highly dynamic environment.
  – the meaning of data and the annotations can change frequently over time/space.
  – E.g. fleet tracking, patient monitoring, etc.

• The semantic IoT is more complex to manage than semantic web. It requires continuous monitoring, pre-processing, filtering, aggregation, annotation and integration.
Semantic IoT goals

- Effective data interoperability between devices and applications. Communication without any prior agreement.
- Generic interworking and automated management of resources.
- Semantic discovery and data querying.
- Semantic matching and binding of devices and applications.
- Semantic reasoning to infer new knowledge from a set of asserted facts.
- Better monitoring and understanding of the surrounding environment.
- Make smart decisions and dynamically adapt to environments changes.
Towards semantic IoT model

• We have good models and description frameworks. RDF, OWL, SPARQL
• Having good models and developing ontologies is not enough.
• Think of the applications and use-cases before starting to annotate the data.
• Semantic descriptions are intermediary solutions, not the end product.
• We should provide machine-interpretable but not machine-untreatable. Think of constrained devices in IoT.
• We should accept the fact that sometimes we do not need full semantic descriptions.
Semantic in oneM2M

- oneM2M offered minor semantic enhancement in release-1 and aims to provide full semantic support in the next releases.
Evolution of semantic in oneM2M

Attributes

- App-ID
- appName
- label
- ontologyRef

Child Resources

- <container>
- <semanticDescriptor>

Release-1: Resource Names (Text) can be defined by the client and so have some meaning. E.g. « Temperature-AE »

Release-1: A set of labels (keywords) can be defined for each resource. E.g. « temperature », « luminosity »

Release-1: Each resource can be linked to an ontology reference (description). E.g. « http://ontology.tno.nl/saref#Device »

Release-2: A complete semantic description can be added to a resource (Ontology). E.g.<RDF>…</RDF>
oneM2M base ontology model

- AreaNetwork
- Thing
- Device
- Value
- Functionality
- Aspect
- Controlling
- Measuring
- Input
- Output
- Target
- Method

Relationships:
- AreaNetwork isPartOf Thing
- Device consistsOf hasThingProperty
- Device hasService
- Device hasFunctionality
- Service hasOperation
- Service hasMethod
- Operation hasInput
- Operation hasOutput
- Functionality hasTarget
- Target refersTo
- Functionality subclassOf Controlling
- Functionality subclassOf Measuring
oneM2M base ontology instance

Device #1234AB

AreaNetwork #WIFI-1

Service #TempServ

Operation #RetrieveTempOp

Method #Retrieve

Output #Temp

hasService

hasFunctionality

hasOperation

hasTarget

hasThingProperty

Value #17

Aspect #Temperature

Measuring #TempFunction

Target #/MN-CSE/AE-123/CNT-TEMP

hasMethod

isExposedBy

isPartOf

concerns

refersTo

describes
Mapping to vertical ontologies

Service Layer
oneM2M base ontology

Vertical domain
SarefOntology

oneM2M:Device
onem2m:Service
onem2m:Functionality
Aspect

saref:Device
saref:Service
saref:Fonction
saref:Temperature
saref:Fonction
Semantic oneM2M architecture

- AE-METER
- AE-TEMP
- AE-LAMP
- MN-CSE-1
- MN-CSE-2
- IN-CSE
- AE-ANALYTICS

Triple Store
Resource Repository

CSF: Data Mgmt & Repo
CSF: Semantics
Generic data modeling for interworking

- Common abstract data model for non oneM2M devices.
Generic interworking using semantic

- Non oneM2M devices are described using the oneM2M base ontology + domain specific extensions.
- The Interworking Proxy Entity translates the ontology instance to resources on the CSE based on pre-defined instantiation rules.
oneM2M semantic challenges

- Access Rights management of semantic data
  - How to protect non open data in oneM2M?
  - Include Access Control Policy in the oneM2M base ontology?
- Semantic querying and discovery
  - SPARQL through « mca » interface?
- Semantic reasoning
  - Infer new knowledge for dynamic reconfiguration.
- Distributed triple store
  - How to connect remote triple store together. Via « mcc » oneM2M interface?
- Performance and support of constrained environments
Thank you for your Attention

benalaya@sensinov.com
www.sensinov.com