Domain knowledge Interoperability to build the Semantic Web of Things

Amelie Gyrard

- Christian Bonnet (Eurecom, Mobile Communication)
- Karima Boudaoud (I3S, Security)
Motivation

- How to help developers to design IoT applications?
  - How to combine domains?
  - How to reuse domain knowledge?
  - How to reason on sensor data?

How to get the meaning of the data?

Application 1: Smart Kitchen
- milk → 1 litter: milk
- orange → 1 kilo: orange
- 110°C

Application 2: Health
- 40°C
- 5g/L: cholesterol

Application 3: Weather Forecasting
- 22°C

- Milk contains lactose?
- Allergic to lactose?
- Orange: Color, Fruit?
- If it is a fruit it contains vitamin C
- Cholesterol-free food

Suggest a recipe according to the external temperature and the health?
The M3 ontology (Machine to Machine Measurement)

- Extension of the W3C Semantic Sensor Networks (SSN) ontology (Observation Value concept)
  - Do not provide a basis for reasoning that can ease the development of advanced applications

- Classify all the concepts in the Machine-to-Machine (M3) ontology
  - **Domain** (health, smart building, weather, room, city, etc.)
  - **Measurement type** \((t = \text{temp} = \text{temperature})\)
  - **Sensor type** (rainfall sensor = precipitation sensor)

- Standardize sensors, measurements and domain terms?
How to deduce new knowledge?

- **Rules example:**
  - If `Domain` == Health & `MeasurementType` == Temperature then `NewType` = `BodyTemperature`
  - If `BodyTemperature` > 39°C then “Fever”
  - `BodyTemperature` and `Fever` are already described in domain ontologies or datasets!

- **More than 200 ontology-based IoT applications are referenced:**
  - Difficulties to automate knowledge extraction
    - Lack of semantic web best practices [OneM2M, Gyrard 2014]
    - Heterogeneous terms used (e.g., etymology, synonyms)
  - Standardize sensor-based domain ontologies?
    - As it has been done for W3C SSN, W3C Time or Schema.org
## Intelligent Transport Systems

<table>
<thead>
<tr>
<th>Authors</th>
<th>Year</th>
<th>Paper</th>
<th>Url onto</th>
<th>Technologies</th>
<th>Sensors</th>
<th>Rules</th>
</tr>
</thead>
<tbody>
<tr>
<td>Feld, Muller</td>
<td>2011</td>
<td>Paper: The Automotive ontology: Managing knowledge inside the vehicle and sharing it between cars.</td>
<td>Work in progress (Response) Concepts: Road, Parking, Traffic Events, Emotional State, Driving Preferences, Mental State, Abilities, Characteristics, Personality</td>
<td></td>
<td>Speed, voice (microphone), ice sensor, heart beat, blood pressure, arousal, alcohol level</td>
<td></td>
</tr>
<tr>
<td>Ruta et al.</td>
<td>2010</td>
<td>Paper: A mobile knowledge-based system for on-board diagnostics and car driving assistance.</td>
<td>Ontology and Rules URL Concepts: Weather conditions (fog, windy, cloud, rain, snow, clear), road surface (uneven, even), road condition (high/low speed), traffic (high/low density), driving style (even pace, imprudent) vehicle model (high, low).</td>
<td>GPS, accelerometer, speed, wind, esp, abs, fog lamp</td>
<td>fog -&gt; low speed, fog lamp, abs (OWL restrictions)</td>
<td></td>
</tr>
</tbody>
</table>

We propose the **Linked Open Rules**

- Heterogeneous formats (ontology editor tool, inference engine, etc.)

---

**Sensors used in your application?**

*Choose a sensor*

**Rules using this sensor**

- Rule: TropicalStormRain, IF Rain GREATER THAN 0.1
  Project: Paul Staroch, 2013
- Rule: **HeavyRain**, IF Rain GREATER THAN 0.5
  Project: **Paul Staroch**, 2013
- Rule: MediumRain, IF Rain GREATER THAN 0.3
  Project: Paul Staroch, 2013
- Rule: RainySpeedSafetyDevice, IF Rainy and Rainy Speed
  Project: Ruta et al. 2010
- Rule: ModeratePrecipitation, IF Precipitation
  Project: Kofler et al., ThinkHome, 2011
- Rule: NoPrecipitation, NoRain, IF Precipitation
  Project: Paul Staroch, 2013
- Rule: **NoPrecipitation, NoRain, IF Precipitation = 0 mm THEN NoPrecipitation**
  Project: Kofler et al., ThinkHome, 2011
- Rule: **HeavyPrecipitation**, IF precipitation GREATER_THAN 4mm THEN HeavyPrecipitation
  Project: Kofler et al., ThinkHome, 2011

---

Scenario 1: Body Temperature

Reason on M2M Data

Find food recommended when you are sick

1. SenML API (Simulate M2M measurements): Simulate temperature measurements
2. M2M Aggregation Gateway (Convert Health Measurements into Semantic Data):
3. We deduce that the temperature corresponds to the body temperature.
4. We deduce that the person is sick.
5. We propose all fruits/vegetables according to this disease.
6. M2M Application: Temperature => Cold => Food: (Wait 10 seconds!)

```xml
<rdf:Description rdf:about="http://sensormeasurement.appspot.com/m3#Measurement5">
  <m3:hasUnit rdf:datatype="http://www.w3.org/2001/XMLSchema#string">Cel</m3:hasUnit>
  <m3:hasDateTimeValue rdf:datatype="http://www.w3.org/2001/XMLSchema#dateTime">0.0</m3:hasDateTimeValue>
  <m3:hasValue rdf:datatype="http://www.w3.org/2001/XMLSchema#decimal">39.0</m3:hasValue>
  <m3:hasName rdf:datatype="http://www.w3.org/2001/XMLSchema#string">temperature</m3:hasName>
  <rdf:type rdf:resource="http://sensormeasurement.appspot.com/m3#Measurement"/>
  <rdf:type rdf:resource="http://sensormeasurement.appspot.com/m3#BodyTemperature"/>
</rdf:Description>
```

6. M2M Application: Temperature => Cold => Food: (Wait 10 seconds!)

- Value = 39.0, Unit = Cel, Type = Body Temperature, Disease = Cold, Food = Kiwi
- Value = 39.0, Unit = Cel, Type = Body Temperature, Disease = Cold, Food = Lemon
- Value = 39.0, Unit = Cel, Type = Body Temperature, Disease = Cold, Food = Honey
- Value = 39.0, Unit = Cel, Type = Body Temperature, Disease = Cold, Food = Ginger

Scenario 2: Weather Temperature & Luminosity

Weather & Activity

1. SenML API (Simulate M2M measurements): Simulate Weather measurements
2. M2M Aggregation Gateway (Convert weather Measurements into Semantic Data): Convert weather measurements
3. We deduce the weather outside.
4. We propose activities according to the weather.
5. M2M Application (Temperature => weather => Activity): Activity & Temperature
6. M2M Application (Luminosity => weather => Activity): Activity & Luminosity
7. M2M Application (Precipitation => weather => Activity): Activity & Precipitation
8. M2M Application (Wind speed => weather => Activity): Activity & Wind Speed

• Value = 39.0, Type = Weather Temperature, Unit = Cel, Weather = Sunny, Activity = BeachSunbathing
• Value = 39.0, Type = Weather Temperature, Unit = Cel, Weather = Sunny, Activity = BeachVolley

Weather & Emotion

• Value = 50000.0, Type = Weather Luminosity, Unit = lx, Emotion = Joy, Color = Yellow
• Value = 50000.0, Type = Weather Luminosity, Unit = lx, Emotion = Happiness, Color = Yellow
• Value = 50000.0, Type = Weather Luminosity, Unit = lx, Emotion = Fear, Color = Yellow
• Value = 50000.0, Type = Weather Luminosity, Unit = lx, Emotion = Sadness, Color = Gray
• Value = 5000.0, Type = Weather Luminosity, Unit = lx, Emotion = Confusion, Color = Gray
• Value = 5000.0, Type = Weather Luminosity, Unit = lx, Emotion = Boredom, Color = Gray
• Value = 5000.0, Type = Weather Luminosity, Unit = lx, Emotion = Depressed, Color = Gray

Paper: Mapping emotion to color [Nijdam 2009]
“Seasonal affective disorder”
SWoT framework (Semantic Web of Things)

- To help developers to build IoT applications:
  - Reason on sensor data
  - Build interoperable IoT applications
  - Easily combine domains
  - Reuse domain knowledge
Conclusion & Future works

- **Standardization suggestions:**
  - OneM2M, ETSI M2M, W3C Web of Things, W3C SSN
  - Semantic web best practices
  - Sensor measurements in a unified way
  - Linked Open Rules
  - Sensor-based domain ontologies
Thank you!

- We are looking for new real-use case scenarios
- gyrard@eurecom.fr
- http://sensormeasurement.appspot.com/