



Making smart discussions between things

M3 Framework: User's guide & tutorial


Creator	Amelie Gyrard (Eurecom - Insight - NUIG/DERI) Designed and implemented by Amélie Gyrard, she was a PhD student at Eurecom under the supervision of Prof. Christian Bonnet and Dr. Karima Boudaoud. Currently, LOVIoT is maintained since she is a post-doc researcher at Insight within the IoT unit led by Dr. Martin Serrano and involved in the FIESTA-IoT (Federated Interoperable Semantic IoT/Cloud Testbeds and Applications) H2020 project.
Send Feedback	Do not hesitate to ask for help or give us feedback, advices to improve our tools or documentations, fix bugs and make them more user-friendly and convenient: amelie.gyrard@insight-centre.org
Google Group	https://groups.google.com/d/forum/m3-semantic-web-of-things
Last updated	June 2016 <ul style="list-style-type: none"> • Adding the end to end scenario • Update the documentation
Created	March 2015
Status	 Work in progress
Goal	This documentation enables a first approach with M3: <ul style="list-style-type: none"> • Understand what we can do with M3 • Understanding M3 and its components
URL	http://www.sensormeasurement.appspot.com/documentation/UserGuide.pdf

Table of contents

I.	Citations	5
II.	Introduction	5
III.	Understanding what you can do with M3	7
IV.	Generating Semantic Web of Things templates with the SWoT generator.....	7
V.	Interpreting IoT data with SLOR.....	9
VI.	Reusing domain knowledge with LOv4IoT	9
VII.	Simulating SenML sensor measurements.....	11
VIII.	Converting senML sensor data with the M3 language and the Semantic Annotator	13
IX.	Testing our scenarios	15

Table of figures

FIGURE 1. HOME PAGE	5
FIGURE 2. M3 FRAMEWORK AND ITS COMPONENTS.....	6
FIGURE 3. BUILDING DEDUCING MEANINGFUL INFORMATION WITH M3 AND ITS COMPONENTS.....	7
FIGURE 4. GENERATING SEMANTIC WEB OF THINGS TEMPLATES	8
FIGURE 5. ZIP FILE GENERATED WITH DOMAIN KNOWLEDGE FOR INTERPRETING SENSOR DATA	8
FIGURE 6. FINDING RULES TO INTERPRET SENSOR DATA WITH S-LOR	9
FIGURE 7. ONTOLOGIES CLASSIFIED IN VARIOUS DOMAINS	10
FIGURE 8. CLASSIFICATION OF PROJECTS ACCORDING TO THE REUSABILITY	10
FIGURE 9. SCREENSHOT OF LOV4IoT	11
FIGURE 10. SIMULATE M2M AREA NETWORKS	12
FIGURE 11. SIMULATING SENSOR MEASUREMENTS.....	13
FIGURE 12. SEMANTICALLY ANNOTATING IoT DATA WITH THE M3 CONVERTER USER INTERFACE	14
FIGURE 13. SEMANTICALLY ANNOTATING SENSOR DATA	15
FIGURE 14. TESTING OUR SCENARIOS	15

Terms and acronyms

IoT	Internet of Things (IoT)
LOV	Linked Open Vocabularies
LOV4IoT	Linked Open Vocabularies for Internet of Things
M3 framework	Machine-to-Machine Measurement (M3) framework
S-LOR	Sensor-based Linked Open Rules
SWoT	Semantic Web of Things
WoT	Web of Things
STAC	Security Toolbox: Attacks & Countermeasures

I. Citations

Please do not forget to cite our M3 framework:


- Cross-Domain Internet of Things Application Development: M3 Framework and Evaluation. 3rd International Conference on Future Internet of Things and Cloud (FiCloud 2015), 24-26 August 2015, Rome, Italy. Amelie Gyrard, Soumya Kanti Datta, Christian Bonnet, Karima Boudaoud
- Enrich Machine-to-Machine Data with Semantic Web Technologies for Cross-Domain Applications IEEE World Forum on Internet of Things (WF-IoT), Seoul, Korea, March 6-8, 2014. Amelie Gyrard, Christian Bonnet and Karima Boudaoud
- All publications:
 - <http://localhost:57708/?p=publication>

II. Introduction

The M3 framework enables assisting to (see Figure 1):

- Develop Semantic Web of Things (SWoT) applications with the SWoT generator.
- Interpret IoT data with S-LOR
- Find & reuse domain knowledge already designed by domain experts with LOV4IoT
- Find attacks & security mechanisms related to specific technologies employed in IoT with STAC

Semantic Web of Things
 M3 framework
 Scenarios
 Publications
 Security
 Contributing to M3
 About us
 M3 Memento



Making smart discussions between things

Machine-to-Machine Measurement (M3) is a framework to semantically annotate and easily interpret [Internet of Things \(IoT\)](#) data. M3 enables to design interoperable domain-specific or cross-domain [Semantic Web of Things \(SWoT\)](#) applications. M3 is composed of the following components:






				
Generating SWoT applications	Reusing domain knowledge	Interpreting IoT data	M3 Interoperable IoT Data & Domain Knowledge	Securing IoT applications

Figure 1. Home page

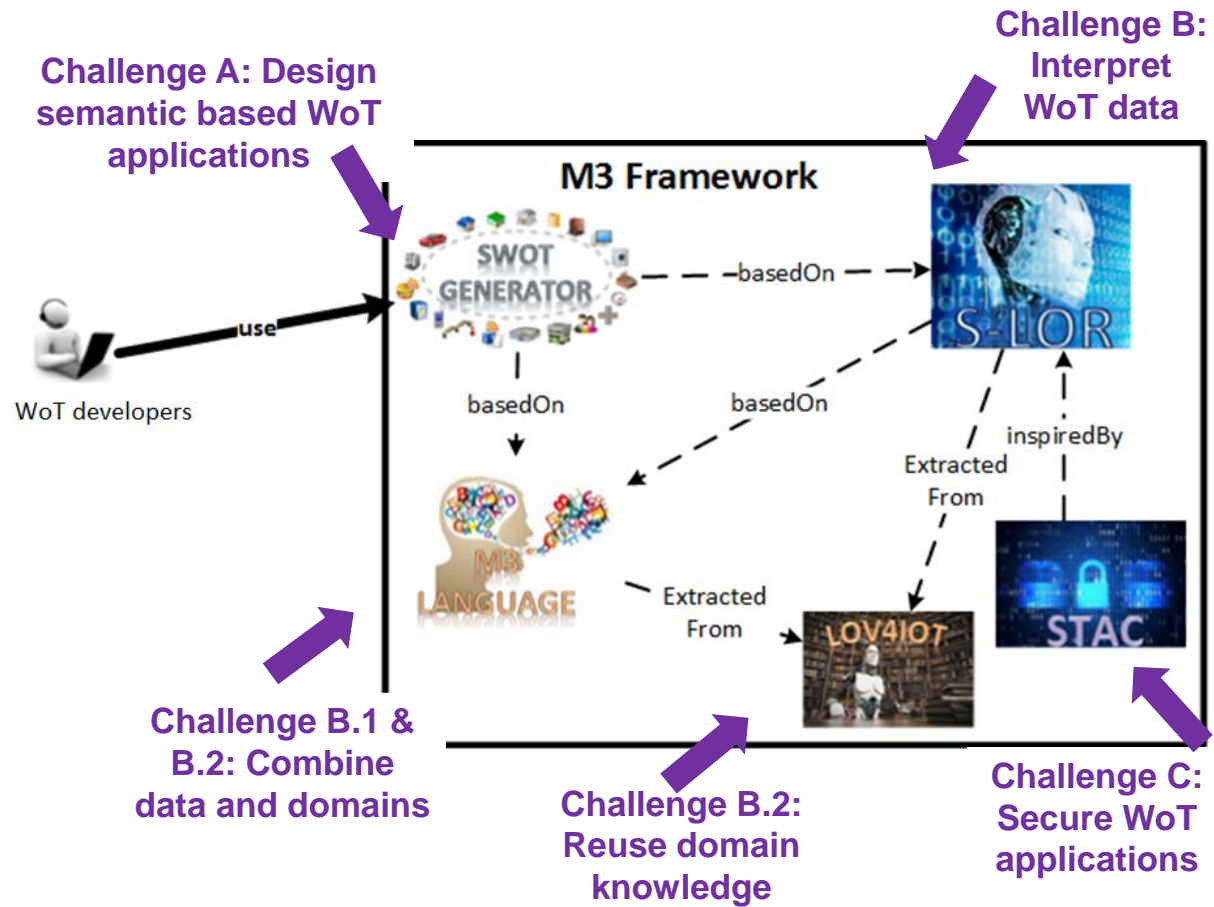


Figure 2. M3 framework and its components

III. Understanding what you can do with M3

- Go there: http://www.sensormeasurement.appspot.com/?p=end_to_end_scenario
- Follow the tutorial:
 - STEP 1 is using the SWoT generator
 - STEP 2 is using the M3 language and its semantic annotator
 - STEP 3 is the core based on the Jena framework to build semantic web of things applications
 - STEP 4 executes the reasoning engine
 - STEP 5 executes the query engine.
 - STEP 6 provides smarter data to visualize in a user-friendly interface.

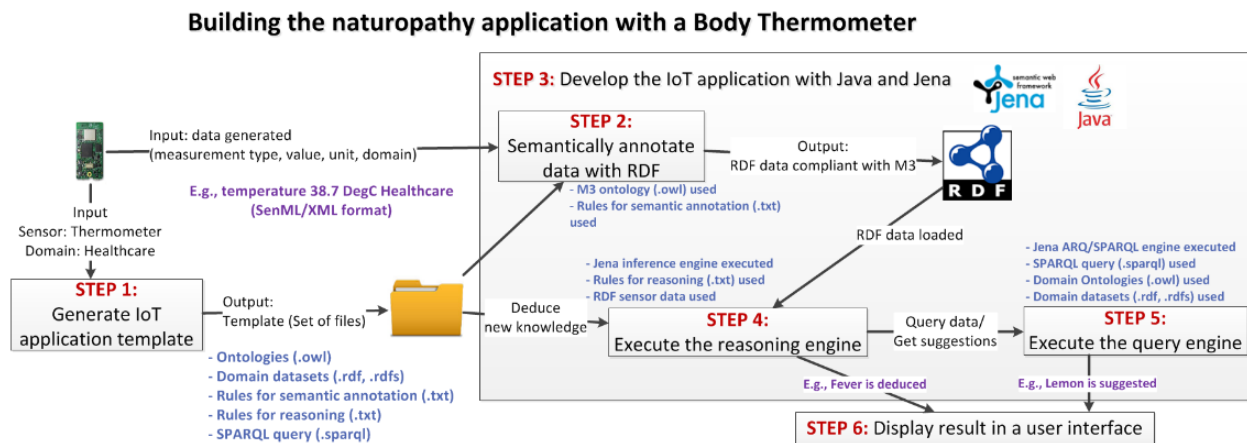


Figure 3. Building Deducing meaningful information with M3 and its components

IV. Generating Semantic Web of Things templates with the SWoT generator

The main purpose of the template generated is to interpret IoT data to provide suggestions.

- ➔ Go to this web page: <http://www.sensormeasurement.appspot.com/?p=m3api> (see Figure 2)
- ➔ Choose a sensor (e.g., Precipitation)
- ➔ Choose a domain (e.g., Weather)
- ➔ Click on the button “Search IoT application template”
- ➔ The drop-down list in STEP 2 is not empty anymore

- ➔ Choose a template (e.g., Precipitation, Transportation and Safety devices)
- ➔ Click on the button “Generate zip file”
- ➔ A zip file has been generated with interoperable M3 and domain ontologies, rules and datasets (Figure 3).

Generate IoT applications to reason on sensor data

STEP 1: Search IoT Application Template

1. Choose a sensor (e.g., Light/Illuminance Sensor) Precipitation Sensor, Pluviometer
2. Choose the domain where is deployed your sensor (e.g., Weather) Weather Forecasting, Meteorology
3. Search IoT Application Template

STEP 2: Choose IoT Application Template

- Choose an application template: Precipitation, Transportation and Safety D...

STEP 3: Download IoT application

- Generate zip file

Precipitation, Transportation and Safety D...
Snow, Transportation and safety devices
Snow, Tourism and Garment
IoT application to suggest safety devices according to the precipitation (e.g., rainy -> low beam)
Precipitation, Transportation and Safety Device

Figure 4. Generating Semantic Web of Things templates

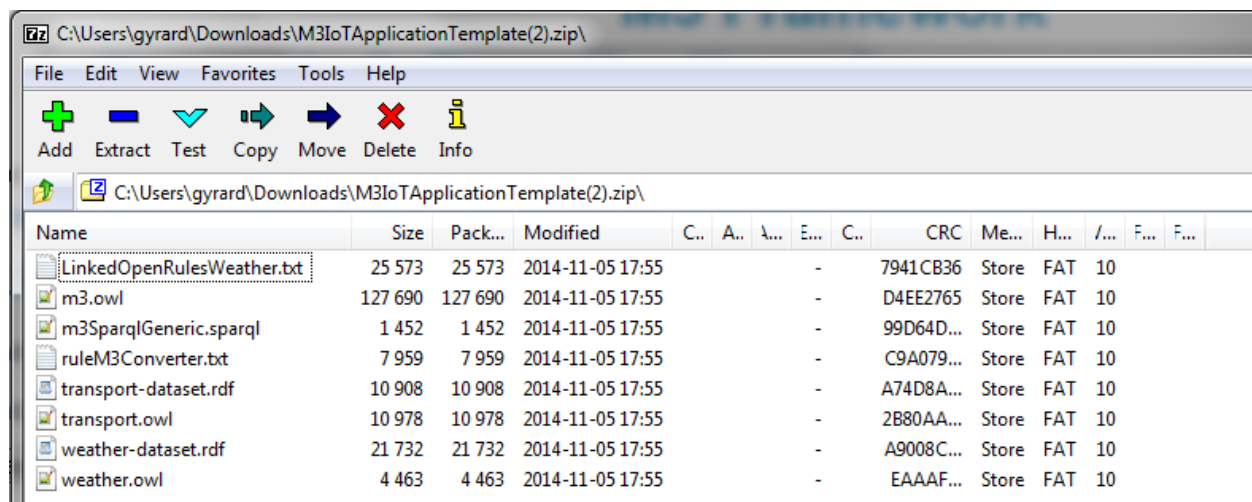


Figure 5. Zip file generated with domain knowledge for interpreting sensor data

V. Interpreting IoT data with SLOR

Go to this web page: <http://www.sensormeasurement.appspot.com/?p=slor>

- ➔ Select a sensor to find all rules interpreting sensor values as depicted in Figure 4 (e.g., Precipitation)
- ➔ The demonstration will show all rules related to the sensor chosen by the user to interpret sensor values.
(e.g., if precipitation = 0 mm/h then NoPrecipitation)
- ➔ You have both the rule for humans and for machines (click on the LinkedOpenRules link)

Sensors used in your application?

Choose a sensor (e.g., accelerometer sensor) Precipitation Sensor, Pluviometer **Choose a sensor**

Rules using this sensor (e.g., choose Wind speed sensor)

- Rule: TropicalStormRain, IF m3:Precipitation greaterThan 4mm THEN TropicalStormRain
Project: [Paul Staroch 2013]. See LOV4IoT for more details.
Linked Open Rules URL: <http://sensormeasurement.appspot.com/RULES/LinkedOpenRulesWeather.txt>
- Rule: HeavyRain, IF m3:Precipitation greaterThan 4mm THEN HeavyRain
Project: [Paul Staroch 2013]. See LOV4IoT for more details.
Linked Open Rules URL: <http://sensormeasurement.appspot.com/RULES/LinkedOpenRulesWeather.txt>
- Rule: MediumRain, IF m3:Precipitation greaterThan 1mm AND m3:Precipitation lessThan 4mm THEN MediumRain
Project: [Paul Staroch 2013]. See LOV4IoT for more details.
Linked Open Rules URL: <http://sensormeasurement.appspot.com/RULES/LinkedOpenRulesWeather.txt>
- Rule: RainySpeedSafetyDevice, IF Rainy THEN RainySpeedSafetyDevice
Project: [Ruta et al. 2010]. See LOV4IoT for more details.
Linked Open Rules URL: <http://sensormeasurement.appspot.com/dataset/transport-dataset>
- Rule: NoPrecipitation, NoRain, IF m3:Precipitation = 0 mm THEN NoPrecipitation **Interpretation of sensor values**
Project: [Kofler et al., ThinkHome, 2011]. See LOV4IoT for more details.
Linked Open Rules URL: <http://sensormeasurement.appspot.com/RULES/LinkedOpenRulesWeather.txt>
- Rule: HeavyPrecipitation, IF m3:Precipitation greaterThan 4mm THEN HeavyPrecipitation
Project: [Kofler et al., ThinkHome, 2011]. See LOV4IoT for more details.
Linked Open Rules URL: <http://sensormeasurement.appspot.com/RULES/LinkedOpenRulesWeather.txt> **Implementation of rules**

A pluviometer is a sensor measuring the amount of precipitation/rainfall.

Figure 6. Finding rules to interpret sensor data with S-LOR

VI. Reusing domain knowledge with LOV4IoT

- ➔ Go to the Linked Open Vocabularies for Internet of Things (LOV4IoT) web page (see Figure 5):
<http://www.sensormeasurement.appspot.com/?p=ontologies>
- ➔ Choose 1 domain by clicking on the image (e.g., transportation) as depicted in Figure 5.

Linked Open Vocabularies for Internet of Things (LOV4IoT)

Reusing domain knowledge expertise

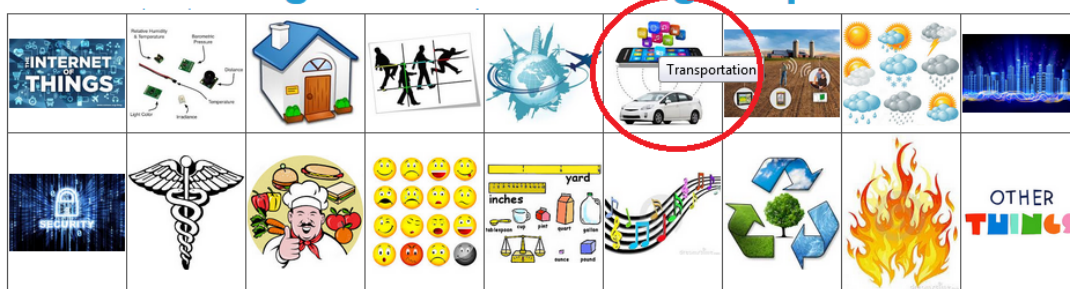


Figure 7. Ontologies classified in various domains

➔ You will find a table with the following information as depicted in Figure 7:

- Domain experts names (authors)
- Year of publication
- Research articles
- Ontology URL of available
- Technologies used in their project
- Sensors used in their project
- Rules designed

-Ontologies and projects have been classified according to different colors (see Figure 6):

- Red: the ontology is not available
- White: we do not have any links to get the ontology
- Orange: we contacted authors to get their ontologies. They answered us they will share ontologies and rules soon.
- Yellow: we retrieve the ontology URL or get a copy
- Green: Ontologies published online, cannot be referenced on the Linked Open Vocabularies (LOV)¹ project due to a lack of best practices.
- Dark green: The ontology is referenced on the Linked Open Vocabularies project. It checks best practices.

The ontology will never be available (lost, confidential, etc.) :-(We are waiting the response of the authors to publish the ontology online	Authors are publishing online the ontology (ongoing work)	Ontology published online but the Semantic Web best practices are not complied with.	Ontology published online and referenced by LOV since Semantic Web best practices are adopted! :-)))	Already on LOV - No email sent
---	---	---	--	--	--------------------------------

Figure 8. Classification of projects according to the reusability

¹ <http://lov.okfn.org/dataset/lov/>

Intelligent Transport Systems

Authors	Year	Paper	Url onto	Technologies	Sensors	Rules	LOV status	Security
Bermejo, Astrain Escola Mail: 14/02/14, Response: 18/02/14	2014	Paper: Ontology based road traffic management	No server (response), but sent us the OWL ontology copy URL Application ? Concepts: 24 classes, 12 properties, 77 rules. Vehicle	SWRL (DLSafeRule), Jess reasoner, extension of A3ME ontology, OWL API 3.4.2	LastBeforeGap, FirstAfterGap, NotOvertaker, Overtaker), distance sensor -> Space (Lateral, Ahead, Behind, NoSpace), Acceleration sensor (BigSpeedDifference, SpeedWith,	77 rules/actions (swrl dlsafe rule in the ontology) change line, decelerate, accelerate, maintain distance with car in front, maintain speed	Priority 1, responsive, ongoing, lov metadata, uri deferencable 26/03/14, ask for label and comment	
Morignot, Pollard et al.	2013	Paper: An ontology-based Model to determine the automation level of an automated vehicle for co-driving Paper: An ontology-based approach to relax traffic regulation for autonomous vehicle assistance	Ontology URL Ontology URL Concepts: emotion driver, Weather Conditions (foggy, cloudy, snowy, sunny, rainy), lighting conditions (day, night, setting sun), road (highway, campaing, urban, mountain), obstacles	Pellet, SWRL (DLSafeRule), Protege, SWOOPS (ontology editor)	Position, velocity, acceleration/braking and steering actuators	foggy -> mode manual, search for parking place, stopped, hasNextMotion	content negociation, uri def error, vapour rdf/xml	

Figure 9. Screenshot of LOV4IoT

VII. *Simulating SenML sensor measurements*

The following interface enables to simulate sensor data:

<http://emulator-box-services.appspot.com/senmladmin/zones>

These data are compliant with the SenML² format.

You can simulate heterogeneous domains (healthcare, smart kitchen, smart home, etc.), as you can see in Figure 8. You can create a new domain (Add sub zone button).

² <http://www.ietf.org/archive/id/draft-jennings-senml-10.txt>

New sub zone

nameZone

Add Sub Zone

Sub Zones

Name	SubZones	Sensors		
Location	No	Yes	Edit	Delete
health	No	Yes	Edit	Delete
kitchen	No	Yes	Edit	Delete
smarHome	No	Yes	Edit	Delete
tempBuiding	No	Yes	Edit	Delete
temperatureOutside	No	Yes	Edit	Delete

Figure 10. Simulate M2M area networks

Click on the button “Edit” associated to the kitchen zone.

You go to the following URL:

<http://emulator-box-services.appspot.com/senmladmin/ahdzfmVtdWxhdG9yLWJveC1zZXJ2aWNlc3IWCxIJWm9uZUFkbWlulgdraXRjaGVuDA/edit>.

You can simulate smart devices (sensors, actuators, transducer, controllers and RFID tags).

In this use case, we simulate RFID tags embedded on ingredients.

We simulate SenML measurements (in XML or JSON).

A measurement has a name, a value, a unit, and the date.

Example 1: Measure name: Temperature, Unit: Degree Celsius, Value: 35

Example 2: Measure name: banana, Unit: Gram, Value: 1000



Use the M3 nomenclature³ to describe sensor measurements.

To be sure that the M3 converter will semantically annotate correctly the sensor measurements.

It will ease the process to build the application with the Semantic Web of Things template.

New Sensor

title sensor

Add Sensor

Sensors

Title	UUID	Measures				
fridge	695eb2c6-befc-43a5-b2d8-a7025063de32	measure name	acidity	float	now	Delete
		value	add a Measure			
	banana: (kind: mass) 1000 g @ now	{ "v": "1000", "u": "g", "t": 0, "n": "banana" }				Delete
	chocolate: (kind: mass) 1000 g @ now	{ "v": "1000", "u": "g", "t": 0, "n": "chocolate" }				Delete
	milk-expires: (kind: time) 5 d @ 2013.02.01 AD at 10:12:10 UTC	{ "v": "5", "u": "d", "t": 1359713530376, "n": "milk-expires" }				Delete
	milk: (kind: volume) 2 l @ now	{ "v": "2", "u": "l", "t": 0, "n": "milk" }				Delete
	peach: (kind: mass) 1000 g @ now	{ "v": "1000", "u": "g", "t": 0, "n": "peach" }				Delete

Figure 11. Simulating sensor measurements

VIII. Converting senML sensor data with the M3 language and the Semantic Annotator

Go to the M3 converter to semantically annotate SenML data with RDF according to the M3 ontology.

³ www.sensormeasurement.appspot.com/documentation/NomenclatureSensorData.pdf

Web page: http://www.sensormeasurement.appspot.com/?p=senml_converter



Use Chrome to get the data in a text format, with Firefox you only have the JavaScript alert popup.

SenML to RDF Converter

Copy/paste your SenML/XML sensor data here :

```
-09eb-4303-  
ae3c-d5d23149ee96">  
<e n="blood pressure" t="0"  
u="Pa" v="56">  
<e n="heartbeat" t="0"  
u="beet/m" v="155"></e>  
<e n="temperature"  
t="1374069830362" u="Cel"  
v="40"></e>  
</senml>
```

SenML/XML to RDF Converter

M3 interoperable IoT data

```
<rdf:Description rdf:about="http://sensormeasurement.appspot.com/m3#Measurement4">  
  <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">1.374069830362E12</m3:hasDateTimeValue>  
  <m3:hasValue rdf:datatype="http://www.w3.org/2001/XMLSchema#decimal">40.0</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>  
</rdf:RDF>
```

M3 inferType

Figure 12. Semantically annotating IoT data with the M3 converter user interface

SenML to RDF Converter

SenML to RDF Converter (Use chrome)

1. [Simulate your data](#)
2. [Get SenML/XML data](#)
3. Enter an url (see previous link):

<http://emulator-box-services.appspot.com/senml/zones/ahdzfmVtdWxhdG9yLWJveC1zZXJ2aWNlc3IVCxlJWm9uZUFkbWlulgZoZWFSdGgM>

SenML to RDF Converter

Wait 1 minute!

Figure 13. Semantically annotating sensor data

IX. Testing our scenarios

- ➔ Go to the menu bar
- ➔ Go to the tab called "Scenarios".
- ➔ Choose a scenario (e.g. tourism)

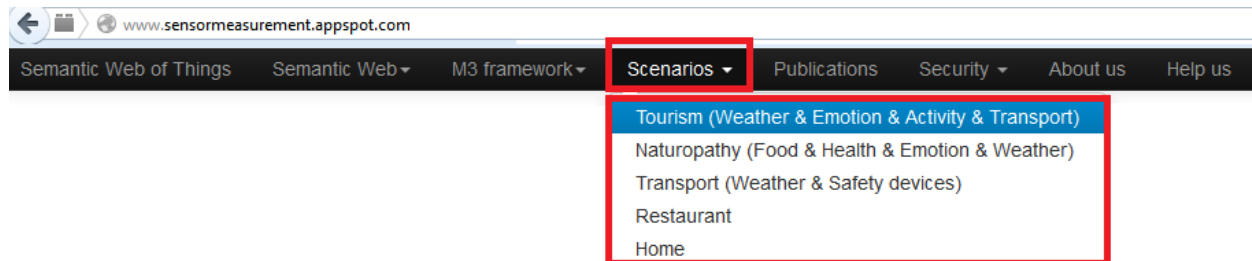


Figure 14. Testing our scenarios

If you choose the tourism scenario. You will have the following web page.

- ➔ Click on the "Activity & Precipitation" button.
- ➔ It will display the M3 results after the reasoning process

Tourism (Weather & Emotion & Activity & Transport)

Weather & Activity

- 1. This scenario is based on these [M3 RDF sensor data](#)
- 2. We deduce the weather outside.
- 3. We propose activities according to the weather.
- 4. M2M Application (Temperature => weather => Activity):
- 5. M2M Application (Luminosity => weather => Activity):
- 6. M2M Application (Precipitation => weather => Activity):
- 7. M2M Application (Wind speed => weather => Activity):

Activity & Temperature

Activity & Luminosity

Activity & Precipitation

Activity & Wind Speed

[Click here to display m3 reasoning results](#)

Input data

Inferred data after M3 process

• Name=precipitation, Value = 1.0, Unit=m	InferType = Precipitation, Deduce = LightRain, Suggest= Paintball
• Name=precipitation, Value = 1.0, Unit=m	InferType = Precipitation, Deduce = LightRain, Suggest= Squash
• Name=precipitation, Value = 1.0, Unit=m	InferType = Precipitation, Deduce = LightRain, Suggest= Concert
• Name=precipitation, Value = 1.0, Unit=m	InferType = Precipitation, Deduce = LightRain, Suggest= Opera
• Name=precipitation, Value = 1.0, Unit=m	InferType = Precipitation, Deduce = LightRain, Suggest= Bowling
• Name=precipitation, Value = 1.0, Unit=m	InferType = Precipitation, Deduce = LightRain, Suggest= Theater

Do not hesitate to try other scenarios.