COSMOS

Cultivate Resilient Smart Objects for Sustainable City Applications

Adapting Cloud SLA metrics approaches for supporting IoT related Use Cases

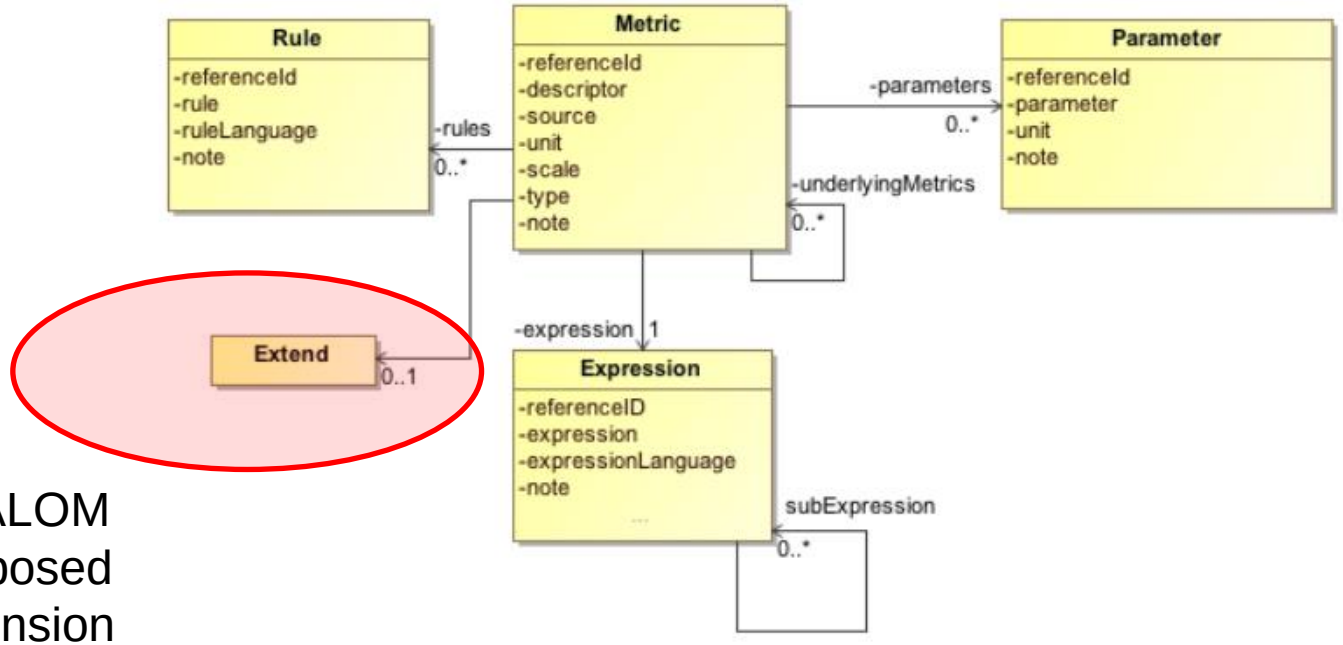
George Kousiouris, Dimosthenis Kyriazis, Andreas Menychtas, Efstathios Karanastasis, Vasiliki Andronikou and Theodora Varvarigou

EuCnC 2016

SLALOM Intro

- H2020 CSA for
  - Development of SLA specification terms
  - Contribution to abstract metric / function applicable to different metrics
  - Submission of our work to ISO-IECJTC1-SC38-WG3 for standardization in the context of the current draft standard 19086-2
  - Main focus: how to enable SLAs to be completely defined and thus monitored/auditable
- Focus of this work:
  - Can we reuse it for IoT Use cases and not duplicate work for a new standard?

ISO 19086-2 Draft Metric model



SLALOM proposed extension

- [from current version of draft standard 19086-2, to be made available in the upcoming weeks]
- **Why is this extension so important?**
- **It enables us to instantiate it differently per case, thus concretely defining the sampling process per type of SLA and metric**

Cloud Examples

- The model has been successfully applied for describing popular Cloud SLAs such as
  - AWS EC2
  - GAE Data Store
  - Microsoft Azure Blob Storage
  - Generic vCore performance metrics
- **Does it make sense to extend it for IoT services?**


```
"samples": [
            {
              "name": "STORAGE GET BLOCK LIST API CALL
response time",
              "referenceId": "SAMPLE_001",
              "scale": "interval",
              "value_limit": PARAM_003,
               "unit": "seconds",
              "protocol": "REST",
              "operation": "GetBlockList",
              "note": "example sample to measure the response
time of the service"
            }
          ]
```

```
{
  "name": "GET BLOCK LIST LIMIT",
  "value": "60",
  "unit": "seconds",
  "referenceId": "PARAM_003"
 }
```

COSMOS & SLALOM Collaboration

- FP7 COSMOS undertook the role of answering this as an IoT project
- Examples of our own services
- What kind of metrics could be offered
- Which ones actually make sense?
  - Questionnaire circulated from March for external input

COSMOS Examined cases

IoT Domain Services	Aspects per category			
Sensing Services	Quality of Data Value	Sensitivity	Battery Life	Minimum Sample Interval
Data Delivery	Availability	Latency	Throughput	#users
Event Processing	Event reaction time	Computed Events per second	Size of Complex rule	
Intelligence /Prediction	% of error	Prediction Horizon	Response Time	
Encryption	Key bit size	Encryption Delay	Data block size	Encryption Algorithm Selection
Privacy	Field selection from data schema		Parametric Blurring of Values	

COSMOS Examined cases

IoT Domain Services	Aspects per category			
Sensing Services	Quality of Data Value	Sensitivity	Battery Life	Minimum Sample Interval
Data Delivery	Availability	Latency	Throughput	#users
Event Processing	Event reaction time	Computed Events per second	Size of Complex rule	
Intelligence /Prediction	% of error	Prediction Horizon	Response Time	
Encryption	Key bit size	Encryption Delay	Data block size	Encryption Algorithm Selection
Privacy	Field selection from data schema		Parametric Blurring of Values	

Example of applying 19086-2 on IoT metrics (details): Missing values limit on data acquisition


```
"parameters": [
    {
        "name": "monitoring cycle",
        "referenceId": "MC_001",
        "unit": "hour",
        "parameter": "12",
        "note": "promise to deliver 12 values per hour from each data item"
    }
],
"underlyingMetrics": [
    {
        "name": "Total number of samples gathered",
        "referenceId": "COUNT_001",
        "unit": "",
        "scale": "interval",
        "expression": {
            "expression": "COUNT_001= COUNT(SAMPLE_001)+COUNT(SAMPLE_002)",
            "expressionLanguage": "ISO80000"
        },
        "samples": [
            {
                "name": "Traffic throughput sensor Data",
                "referenceId": "SAMPLE_001",
                "scale": "interval",
                "value": "Car Throughput",
                "unit": "vehicles/hour"
            },
            {
                "name": "Traffic Speed sensor Data",
                "referenceId": "SAMPLE_002",
                "scale": "interval",
                "value": "Speed",
                "unit": "km/hour"
            }
        ]
    }
```

Example of applying 19086-2 on IoT metrics (higher level)


```
"name": "Sensor Service guarantee for estimated quantity of data ",
"referenceId": "QOD_001",
"scale": "NOMINAL",
"expression": {
    "expression": "PRV_001 > PARAM_002",
    "expressionLanguage": "ISO80000"
},
"parameters": [
    {
        "name": "Unreceived values percentage limit",
        "referenceId": "PARAM_002",
        "unit": "%",
        "parameter": "10"
    },
    {
        "name": "Calculation cycle",
        "referenceId": "CC_001",
        "unit": "day",
        "scale": "INTERVAL",
        "parameter": "1"
    }
],
"underlyingMetrics": [
    {
        "name": "Percentage of received values",
        "referenceId": "PRV_001",
        "unit": "%",
        "scale": "RATIO",
        "expression": {
            "expression": "PRV_001= COUNT_001/24*MC_001*LENGTH(SAMPLES)",
            "expressionLanguage": "ISO80000",
            "note": "More generic parametric expression based on size of samples
        },
        "parameters": [
            {
```

Conclusions

- Some of the metrics are almost identical to Cloud services
  - Availability
  - Latency
  - Throughput

- Others portray differences
  - Quality of Data Value (QoI)
    - Would be considered a must in Cloud services (no erroneous values when accessing e.g. a DB service)
    - Can be varying in IoT during data acquisition due to sensor features, transfer channels etc., and not necessary to be 100% accurate or existent

- **But as a structure and logic, Cloud based standards can be used in principle to describe them if adapted to the IoT rationale**

Thank you!
Any questions?

George Kousiouris
(ICCS/NTUA)

www.iot-cosmos.eu

The research leading to these results has received funding from the EC Seventh Framework Programme FP7/2007-2011 under Grant Agreement n° 609043