Demonstration: Web-based Visualisation and Monitoring of Smart Meters using CQELS

Maxim Kolchin, Dmitry Mouromtsev, Sergey Arustamov

ITMO University
Saint-Petersburg, Russia
kolchinmax@niuitmo.ru, {d.muromtsev,sergey.arustamov}@gmail.com

Abstract. In this demonstration paper we describe a web application that allows visualisation of current and historical readings of smart meters and monitoring their current states using CQELS for an integration of static and streaming RDF sources.

Keywords: Ontologies, RDF stream processing, Smart Meter, Data Integration, Visualisation

1 Introduction

Semantic Web technologies have already proved [1, 2] that they may become a tool for solutions aiming to solve issues with syntactic and semantic integration of data in heterogeneous formats, represented by different models. RDF and OWL provide a common model and vocabulary for such type of data with SPARQL to query data sources and Reasoning to infer "new information" using first-order predicate logic and inference rules. Moreover, RDF stream processing technologies allow combining background knowledge and streaming data together.

These sort of ideas may be applied with Smart Meter data where integration of static and streaming data is important to monitor effectively the state of the meters and the whole network and also respond immediately to emergency situations and unpredictable events.

In our case we have three types of smart electric meters and totally 55 instances that are able to transmit their readings each 5 minutes to a server. Our goal is twofold: (a) to be able to see the current reading in *real-time* and previous ones within a period with charts and (b) more importantly to automate identification of emergency situations, i.e too high voltage value in a line.

In this demo we show how we have tuned the web application to access streaming data coming from smart meters the same way it access a SPARQL endpoint.

The source code of the web application can be found on Github[1].

[1] The source code of the application, URL: https://github.com/ailabitmo/daafse

2 Architecture & Implementation

The architecture of the application is shown on Fig 1. The streaming data (the current readings) coming from smart meters is collected and transformed by RDF stream publisher to RDF streams that are in their turn published through the Message Bus implementing AMQP[2] protocol. Each RDF stream has an URI represented by AMQP URI[3] schema, i.e `amqp://example.com:10000/vhost?exchangeName=exchange&routingKey=mercury230_13534128`. The static data in SPARQL endpoint presents information about smart meters, like a type, serial number, installation location, precision, operating range and etc. Integration of static and streaming sources is performed through CQELS engine and an extension of SPARQL language supporting RDF streams. CQELS engine and the query language are presented in the paper [3].

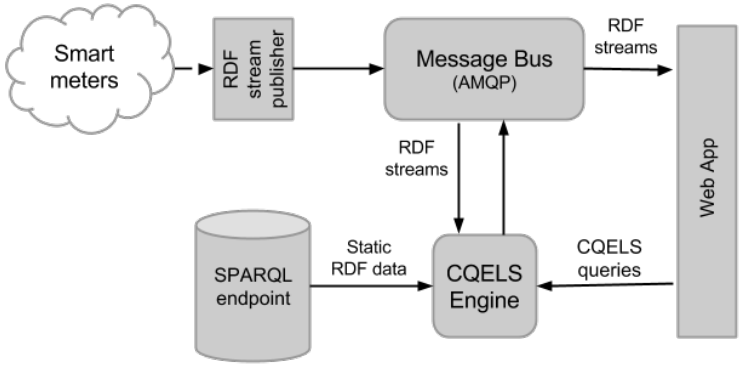

Fig. 1. System architecture: The App, Smart Meters, SPARQL endpoint and RDF streams

The Web App visualises the readings in a form of charts in *real-time* and allows to register a CQELS query that goes off alerts in case if there are some issues in functioning of smart meter or the network, i.e. there is a way to register a query the signals of a too high voltage value ($\geq 220V + 10\%$) measured at least by one of *known* smart meters.

The result of the query is a bag of triples representing an alert that is published to an RDF stream. See an example query and alert in Section 2.2.

[2] Advanced Message Queuing Protocol (AMQP), URL: `http://en.wikipedia.org/wiki/Advanced_Message_Queuing_Protocol`

[3] AMQP URI spec, URL: `https://www.rabbitmq.com/uri-spec.html`

2.1 Ontologies

Several domain ontologies are used to represent the readings, smart meters, and alerts: Semantic Sensor Network (SSN)[4] ontology, the Electric Meters (EM)[5] ontology that is an extension of SSN ontology developed for this application and The DOLCE+DnS Ultralite ontology[6]. On Fig. 2 you can see a smart meter and an observation represented with these ontologies.

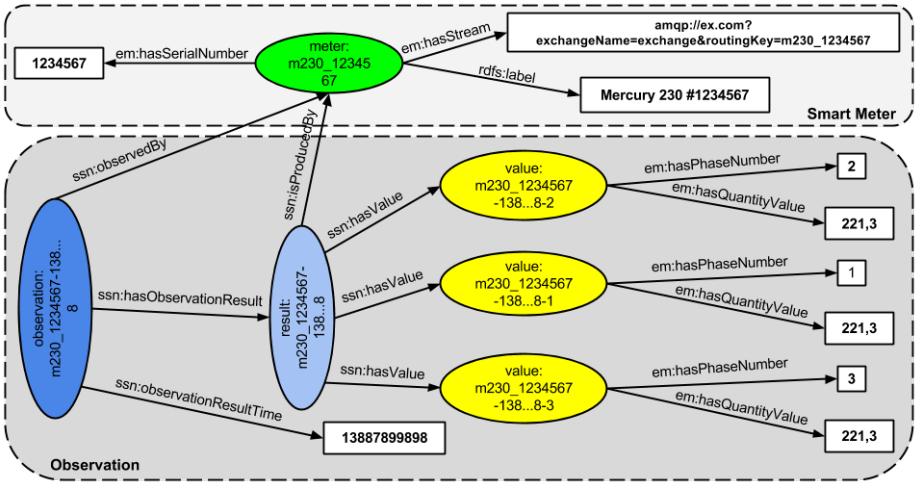

Fig. 2. Smart Meter and Observation

The EM ontology extends SSN ontology with several classes and properties:

- *em:ElectricityFeature* is a subclass of *ssn:FeatureofInterest*,
- *em:PolyphaseVoltageObservation* is a subclass of *ssn:Observation*,
- *em:hasStream* object property representing an RDF stream URI,
- and other classes and properties.

2.2 An example query

Below we show an example of query firing an alert each time when voltage value in any of electric phases measured by one of smart meters is $\geq 220V + 10\%$. All alerts are sent to a predefined RDF stream and consumed by the Web App.

[4] Semantic Sensor Network ontology, URL: `http://purl.oclc.org/NET/ssnx/ssn#`
[5] The Electric Meters ontology, URL: `http://purl.org/NET/ssnext/electricmeters#`
[6] DOLCE+DnS Ultralite ontology, URL: `http://www.loa-cnr.it/ontologies/DUL.owl#`

```
CONSTRUCT {
    ?alert a :TooHighVoltageValue ;
        dul:hasEventDate ?time ;
        dul:involvesAgent ?meter .
}
FROM NAMED <http://ex.com/sparql>
WHERE {
    GRAPH <http://ex.com/SmartMeters/> { ?meter em:hasStream ?stream }
    STREAM ?stream [NOW] {
        ?observation a em:PolyphaseVoltageObservation ;
            ssn:observationResultTime ?time .
        ?output a em:PolyphaseVoltageSensorOutput ;
            ssn:isProducedBy ?meter ;
            ssn:hasValue ?value .
        ?value em:hasQuantityValue ?qvalue .
    }
    FILTER(?qvalue >= (220 + 220*0.1))
    BIND (IRI(CONCAT(STR(?meter), '/alerts/', STR(?time))) AS ?alert)
}
```

RDF stream URIs of smart meters are queried from *http://ex.com/SmartMeters/* graph in *http://ex.com/sparql* SPARQL endpoint. In a more complicated scenario the highest possible voltage value for different smart meters may vary and instead of creating a separate alert for each type of meters this value can be queried from a SPARQL endpoint.

In this example an alert is an instance of *:TooHighVoltage* class that is a subclass of *dul:Event* class. Below a set of triples describing the alert:

```
:TooHighVoltage owl:subClassOf dul:Event ;
    rdfs:comment ...@en ;
    rdfs:label ...@en ;
    dul:isClassifiedBy :Warning .
```

Acknowledgments. This work has been partially financially supported by the Government of Russian Federation, Grant #074-U01.

References

1. Tallevi-Diotallevi S., Kotoulas S., Foschini L., Lcu F., Corradi A.: Real-Time Urban Monitoring in Dublin Using Semantic and Stream Technologies. In: The Semantic Web–ISWC 2013, pp. 178–194. Springer Berlin Heidelberg (2013)
2. Balduini M., Della Valle E., Dell'Aglio D., Tsytsarau M., Palpanas T., Confalonieri C.: Social listening of city scale events using the streaming linked data framework. In: The Semantic Web–ISWC 2013, pp. 1–16. Springer Berlin Heidelberg, (2013)
3. Le-Phuoc D., Dao-Tran M., Xavier Parreira J., Hauswirth M.: A Native and Adaptive Approach for Unified Processing of Linked Streams and Linked Data. In: The Semantic Web ISWC 2011, pp. 370–388. Springer Berlin Heidelberg, (2011)