1 Abstract Description

In this project we provide a data gathering scenario by exposing some air quality sensors around the "Espace Fauriel" neighbourhood. We configure the sensors to be able to transmit data instantaneously and be accessed by requests to get the current updates. The data is stored maintaining interoperability and it is highly interchangeable in a machine readable manner where it could be useful for any future use-cases such as simple and complex queries or even data analysis.

2 Objective and Expected Results

The expected results are acquiring data about the air quality of the espace fauriel neighbourhood that will be found as RDF format on a triple store that can be useful for any proposed solution later.. In order to get the instantaneous status of a given sensor, we can rely on one of the two ways:

- Every specific time δ, the sensor sends his current status formatted as JSON, XML, or CSV... to the influxdb server, which will override the old status by the new one. Using this way, when the user queries the current parameters of the sensor, the data found on influx db will be the answer to his query.
- Every specific time δ, the sensor sends his current status to a triple store where his status will be added with a mark that it's the last update. When the user queries the current parameters of the sensor, the last update on the triple store will be the answer of his query.

In this document, we are relying on the first way (influxdb).
The objective is to visualize these data through a web page, query it, and ask questions about the current and the previous air quality using a question answering system like QAnswer (https://qanswer-frontend.univ-st-etienne.fr/).

3 Envisioned Use Cases

- **Store data in RDF format for semantic usages:**

  1. Instantaneous data sent from sensors will be converted to RDF format and stored in a triple store.
  2. The triple store will accept simple and complex SPARQL queries to process the data and reply with the correct answer.

- **Update of the influxdb using real time data:**
    1. Sensors will send instantaneous data to the influxdb.
    2. Influxdb will accept queries to get current state of the sensors.
- **Provide thing description to ask questions about data**
- **Visualization of the current state of the sensors:**
    1. Get the current state if each sensor from ifnluxDB.
    2. Update the we-bpage with the current data
- **Visualization of the history of the sensed data:**
    1. Get the data about a sensor from the triple store between two dates
    2. provide a graph showing the variation of the state.

4 Functional and Technical Requirements

- 6 sensors responsible for sensing, receiving and sending data about air quality.
- host an online machine to run an Apache server for PHP scripts, also run a triple store on a specific port to be contacted by the web-page for further usages.
- Real-time database for storing and processing the data received from the sensors (InfluxDb) where sensors publishes data continuously and those data are accessible by the web-page for data visualization .

5 Planning of Realization

We provide the planning of the project but this would be modified later especially the dates and maybe we could add some tasks or even cancel some that wouldn't match the project's goals.

Task	Start Date	End Date	Desciption
Sensors Installation	19-11-2019	22-11-2019	Configure and deploy the air quality sensors then connect them to the network.
Sensors Synchronization	25-11-2019	29-11-2019	Transmit data from sensors to an online database.
Data in RDF	2-12-2019	4-12-2019	Model data in RDF and provide the mechanisim to insert in triple store
Data on influxDB	5-12-2019	9-12-2019	Transmit data from sensors to influxDB instantaneously
Visualization webpage	10-12-2019	17-12-2019	Visualize data transmitted by sensors in a web-page using the triple store and influx DB as databases.
Things description	13-1-2020	15-1-2020	Provide the things description for the sensors and test it on Qanswer

6 System Design

In figure 1 we show the architecture of the system we are creating. We provide a synchronous communication between the air quality sensors deployed and the two kinds of databases we have, on top of those a web-page available for the visualization of the data released acquiring it's content from the databases. Also we provide a REST API to read directly the current state and this is integrated as things description which tells Qanswer what are those sensors and how they can be accessed for information retrieval.

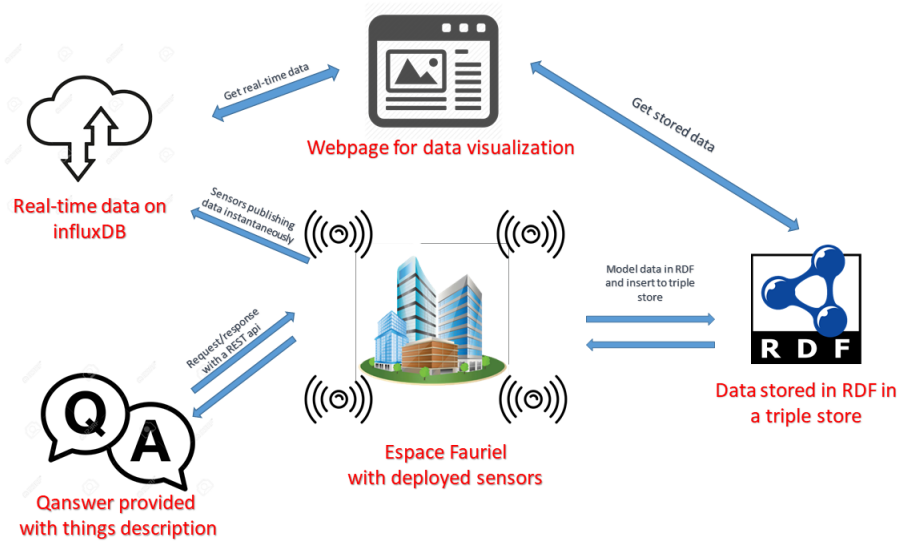

Figure 1: System design and data flow.