Transforming Meteorological Data into Linked Data

Ghislain Atemezing [a,c,*], Oscar Corcho [a], Daniel Garijo [a], José Mora [a], María Poveda-Villalón [a], Pablo Rozas [b], Daniel Vila-Suero [a] and Boris Villazón-Terrazas [a]

[a] *Ontology Engineering Group, Departamento de Inteligencia Artificial,Facultad de Informática, Universidad Politécnica de Madrid*
E-mail: {ghis,dgarijo}@delicias.dia.fi.upm.es, {ocorcho,jmora,mpoveda,dvila,bvillazon}@fi.upm.es
[b] *Agencia Estatal de Meteorología*
E-mail: prozasl@aemet.es
[c] *eurecom, Sophia-Antipolis, France*
E-mail: auguste.atemezing@eurecom.fr

Abstract. This paper describes the process followed in order to make some of the public meterological data from the *Agencia Estatal de Meteorología* (AEMET, Spanish Meteorological Office) available as Linked Data. The method followed has been already used to publish geographical, statistical, and leisure data. The data selected for publication are generated every ten minutes by the 250 automatic stations that belong to AEMET and that are deployed across Spain. These data are available as spreadsheets in the AEMET data catalog, and contain more than twenty types of measurements per station. Spreadsheets are retrieved from the website, processed with Python scripts, transformed to RDF according to an ontology network about meteorology that reuses the W3C SSN Ontology, published in a triple store and visualized in maps with Map4rdf.

Keywords: meteorology, ontology, Linked Data, Sensor Networks

1. Introduction

Governments and their associated agencies worldwide are making some of their data sources available under open data licenses, so as to ensure consumption by the general public and other public and private organisations. In this context, AEMET[1], the Spanish Public Weather Service announced on November 2010 a major change in its data policy, offering a gradual, free and public access to its data in electronic format. As a first step AEMET made publicly available in its website meteorological and climatic data registered by its weather stations, radars, lightning detectors and ozone soundings. These data are currently published as spreadsheets in the AEMET FTP server, linked from its website and detailed in section 3.

Our work aims at facilitating the use of these data by processing them and offering them as Linked Data. Following our method for Linked Data generation [2], which we have used successfully in other domains, we start by processing these data and generating RDF according to a meterology ontology network that extends the W3C Semantic Sensor Network Ontology (SSN). The aim of this ontology network is to represent knowledge related to measurements made by weather stations. Each of these measurements represents the state of the atmosphere (humidity, pressure, temperature, wind, etc.) in a particular place and time, and is conducted through the sensors equipped at each weather station. Finally we publish these data according to Linked Data principles and visualize it with map4rdf [2].

The structure of this paper corresponds to each of the steps of this method. After reviewing briefly some of the state of the art in the area of Semantic Sensor

[*] Alphabetical order

[1] `http://www.aemet.es/`

Web, Section 3 describes the data sources selected to be converted as Linked Data. Section 4 presents the design decisions for the URIs, while Section 5 describes the RDF generation process. Section 6 explains the development of the Ontology Network for Weather Data. Next, Section 7 presents the exploitation of the metereological Linked Data, and Section 8 describes some recommendations for publishing Linked Sensor Data. Finally, Section 9 presents the conclusions and future lines of work.

2. Related work on Semantic Sensor Web

Several research efforts have addressed the bridge between linked (sensor) data and the Sensor Web. We will now describe some of these works briefly.

Patni et al. [9] present a framework for publishing the sensor data on the Linked Open Data Cloud. The proposed framework converts raw sensor observations to RDF and links with other datasets on the LOD.

Le Phuoc et al. [10] describe SensorMasher, an infrastructure that makes sensor available following the linked open data principles and enables the integration of such data into mashups. SensorMasher allows users to semantically describe raw sensor readings, then the descriptions can be exploited in mashups.

Page et al. [8] present a prototype API for exposing data from Channel Coastal Observatory in the UK. The prototype supports Semantic Web clients, OGC GML clients, and hybrid applications by combining REST and Linked Data principles.

From all these works, none of them consider how to design the URIs of the resources coming from the sensors or the reuse of well-known available ontologies for modelling sensor networks. None of them follow either an iterative and incremental linked data life cycle in the process.

3. AEMET Data Sources

Among all of the data made available in the FTP server from AEMET, we have focused on surface meteorological observing stations, and more precisely in measurements taken in ten minute interval times[2]. The data are recorded in several points in the Spanish geography and can be represented as a set of variables, for instance temperature or pressure, on points in a map. A full description of the variables considered is included as part of the ontology used [3]. These stations belong to the Regional Basic Synoptic Network (RBSN) in Spain operated by AEMET, in accordance with the World Meteorological Organization[4] (WMO). The stations are globally identified by a code and have to accomplish strict quality controls. Their data are used to feed meteorological models and to draw up climate studies. AEMET has around 250 automatic weather stations of this network, registering pressure, temperature, humidity, precipitation and wind data every 10 minutes. Data from the different stations are provided in CSV files, compressed with gzip, updated every hour and kept for seven days. This means, every hour six new files are added, corresponding with periods of ten minutes, and every day a new folder is created to store the files for that day.

AEMET data are stored in comma separated values and are available online for public download and reuse. The name of each file in this site follows the naming convention "`yyyymmddhhmm_datos.csv`". This allows processing directly a set of specific files without parsing (or even downloading) all of them just to check which files are relevant for a specific time range. For example, the filename "`201102121900 _datos.csv`" means the data are from the year 2011, the month of February, on the day 12 at 19:00h. When looking inside the files we can see some data about the stations, as the position or ID, with one row for each station. In the same row we find data for the observations of a specific station, having each column the ID of the observation type and the value recorded.

4. URI design for meteorological data

URI[5] (Uniform Resource Identifier) design has proved to be a challenging issue for the Linked Data community. Therefore, there has been a lot of discussion and work on defining guidelines for the effective use of URIs [6,7] .

The main conclusion behind these discussions is that URIs should be designed with *simplicity*, *stabil-*

[2] ftp://ftpdatos.aemet.es/datos_observacion/observaciones_diezminutales/

[3] https://github.com/boricles/aemet/tree/master/model

[4] http://www.wmo.int/

[5] http://tools.ietf.org/html/rfc3986

[6] http://www.w3.org/Provider/Style/URI

[7] http://www.w3.org/TR/chips/

ity and *manageability* in mind, thinking about them as identifiers rather than as names for Web resources.

In this sense, the seminal guide can be found in "Cool URIs for the Semantic Web W3C Interest Group Note" [8]. These guidelines have been followed throughout the development of the present work wherever they were applicable to its specific characteristics and needs, especially the ones regarding the structure of the base URIs and content-negotiation strategies, as it will be discussed later on this section.

Another related topic is the need for following clear naming conventions for ontology concepts (classes and properties), some survey-based guidelines can be found in [11], or more specific to OWL ontologies and natural language processing in [6], where the authors propose general style guidelines for naming OWL classes and properties in order to assist them in the application of ontology verbalization and natural language processing techniques. In some of the reviewed works, the authors refer to labels when meaning local IDs or local names (the part of the URI that makes it distinct from others sharing the same base URI). This is a common misunderstanding, that often: (1) leads to the misuse of annotation labels, such as *rdfs:label* and others like *skos:prefLabel*, producing difficulties in the application of NLP techniques or the development of Semantic Web applications; (2) gives the false impression that URIs' local IDs are intended to name or label resources, when "Uniquely identifying a resource is different from merely naming that resource" as stated in [3]. As it will be discussed in section 6, the ontology engineers have abided by a clearly defined set of naming conventions as detailed in the following points. Additionally, a correct use of multilingual labels (currently Spanish and English) in natural language has been applied for both the concepts, properties and instances.

Following this set of principles, this section presents the design decisions regarding the assignment of URIs to the elements of the knowledge base, as well as the set of conventions used in the project: (1) base URIs structure and conventions; (2) URIs and naming conventions of the AEMET Ontology (TBox), described in section 6; and (3) URIs of the generated instances (ABox), analysed in section 5.

4.1. Base URI structure

The base URI, common to all elements in the knowledge base, is `http://aemet.linkeddata.es/`, which is also the domain hosting the explotation solutions described later in Section 7. After having the base URI settled, it is important to define a pattern that helps to separate different kinds of entities (according to the Cool URI recommendations) and that complies with the design principles of simplicity, stability and manageability. Therefore, the following decisions have been made:

1. Separate the TBox from the ABox establishing the following URI schemes: (1) `http://aemet.linkeddata.es/ontology/` for the TBox; (2) `http://aemet.linkeddata.es/resource/` for the ABox.
2. For the TBox, append the name of the class of resource identified by the URI (e.g. `http://aemet.linkeddata.es/resource/Point/`), this pattern will be further analysed and explained in section 6.

4.2. TBox URIs

As mentioned before, the base URI scheme for TBox elements is `http://aemet.linkeddata.es/ontology/`, to which the concept local name or local ID is appended. In order to assign names to the different elements, two kind of elements have been diferentiated: classes and properties, including both OWL object and datatype properties. The naming of each kind of element abide by different conventions:

- **Classes**: The camel case style has been followed starting with upper case, using English names.
- **Properties**: The camel case style has been followed starting with lower case, using English names.

Class and property labels have been added in two different languages (Spanish and English). The main motivation behind this decision is that such natural language descriptions associated to ontologies have proven to be of major importance, not only to ontology users but also to assist in tasks such as ontology mapping, information extraction, ontology verbalization and, even more relevant to the goal of the AEMET Linked Data initiative, in the development of Linked Data applications.

[8] `http://www.w3.org/TR/cooluris/`

4.3. ABox URIs

If the aim of the presented work and its associated data is to provide a meaningful set of interconnected assertions and relationships, special attention must be paid to the unique identification of such sets of assertions. In other words, it is important to carefully assign identifiers (URIs) to the resources within the generated dataset and to ensure that every distinct real-world object gets its “unique” URI. The rationale behind this is to mitigate issues such as *co-reference*, described in [7] as “the problem of ensuring that two different entities do not share the same identifier, and conversely identifying when two identifiers refer to the same entity” or the *instance unification problem* described in [4] and analysed in [1]. In this sense, throughout the development of the present work a series of decisions have been made in order to preserve the integrity and semantics of the data.

The first decision regarding ABox URIs was to use *Patterned URIs*[9], as described in "Linked Data Patterns". Such solution, by adding the class name to the base URI, mitigates *co-reference* problems between two distinct individuals of different type and same local ID, while enabling URIs to be algorithmically constructed in consumer applications, by following a simple and clear pattern.

The second decision was to use *Natural keys*[10] whenever the source data allowed to, or in other words when the source group of resources already had a unique identifier. This is the case of the synoptic weather stations and their unique identifier: *INDSINOP*.

The usage of the aforementioned patterns also facilitates the linking process to and from other datasets in the LOD cloud as the URIs structure they provide helps disambiguation and separation of resource classes, making easier for data users and engineers to make statements about the resources having some sort of certainty about the resources identities.

Finally, some other ad hoc URI structures have been created in order to uniquely identify the resources, most of them could be referred to as a "composite" pattern, where the local ID is formed by several interconnected pieces of information related to the resource being identified (e.g. `http://aemet.linkeddata.es/resource/Observation/at_1306446000000_of_08202_on_PREC`).

A summary can be found in Table 1, presenting the different local IDs assigned to each type of individual in the ABox, according to the aforementioned design decisions and patterns.

5. RDF Generation

RDF has been generated with ad hoc Python scripts. These scripts are executed in two processing steps, integrating with ease the generation of RDF and tasks as crawling the FTP server where the CSV files are located. The generated RDF is stored in Virtuoso[11], which integrates with Pubby[12] for the publication of the results and makes them available for humans as well as computers. Both processing steps are described below.

The first step generates the information about the automatic stations, which is static, and thus needs to be executed only once. Even if the script is executed just once, developing a script to generate this information takes shorter than doing it manually. The number of stations is 784 and the time required to develop the script less than one hour. Having a script does also provide a better maintainability for future changes than manually generating the RDF, which is something usual in an evolutionary prototype context.

The second step generates the information about the observations on a regular basis, keeping the data updated. The observations are obtained by crawling the FTP server from AEMET. When new files are added or old files are modified, these are downloaded and processed. The process is very similar to the previous script and as far as has been observed it may be generalizable when using plain text files to generate RDF.

Both steps use templates that contain the RDF to be generated, in this case expressed using the N3 notation. The fields of these templates are obtained from the files processed and the instantiated templates are written to the RDF files. The main limitation of the approach is that there is no validation for the templates, thus may not be a convenient solution for those not accustomed to coding in computer languages. The two most relevant parts of the task are the parts that have to be replicated in each case. The first part is defining the templates that allow to express in RDF the information that

[9] `http://patterns.dataincubator.org/book/patterned-uris.html`

[10] `http://patterns.dataincubator.org/book/natural-keys.html`

[11] `http://virtuoso.openlinksw.com/`

[12] `http://www4.wiwiss.fu-berlin.de/pubby/`

Table 1
ABox URI patterns

Resource class	Local ID pattern	Example
Station	INDSINOP	http://aemet.linkeddata.es/resource/Station/08202
Point	INDSINOP	http://aemet.linkeddata.es/resource/Point/08202
Observation	at_<timestamp>_of_<INDSINOP>_on_<observedProperty>	http://.../Observation/at_1306446000000_of_08202_on_PREC
Interval	tenMinutes_since_<timestamp>	http://.../Interval/tenMinutes_since_1306446000000
Instant	timestamp	http://aemet.linkeddata.es/resource/Instant/1306446000000
DateTime	timestamp	http://aemet.linkeddata.es/resource/DateTime/1306446000000

is to be generated. The second part is extracting the information from the CSV files and other plain text files, this may require defining and matching regular expressions in the most complex cases. AEMET CSV did not require regular expressions but just splitting each line.

Considering this, even though the code has been generated ad hoc for this use case the experience suggests that the process should be fairly generalizable. Additionally the use of a high level language as Python eases the integration of the different tasks that have to be performed, in this case crawling an FTP server, converting CSV.gz files to RDF and storing the results in Virtuoso. Other improvements over the process could involve software to assist in the generation of the templates and regular expressions.

6. An Ontology Network for Weather Data

The development of our ontology network has been based on the guidelines proposed in [14], which follow ontology engineering methodologies like the NeOn Methodology [12].

6.1. Related Ontologies for modeling Weather and Phenomena

In order to reuse existing ontologies, we have searched for ontologies in the domain of meteorology using two popular ontology-oriented Semantic Search Engines (SSE), Watson[13] and Swoogle[14]. Our search has been based on the following three functional requirements: (i) Language (the ontology should be in English or Spanish), (ii) interoperability (the ontology should be implemented in a Semantic Web language (OWL or RDF(S))) and (iii) usability (the ontology should be easy to analyse and reuse).

The search operation has been performed using the following set of keywords, considering the structure of the AEMET dataset : weather - forecast - temperature - phenomena - pressure - wind - stations location and visibility.

- **Weather ontology (WO)**[15]: Ontology created by the AgentCities group at the University of Aberdeen (UK) and implemented in DAML. WO models weather, pressure, temperature and visibility as events. They contain the classes WeatherEvent, PressureEvent, TemperatureEvent and VisibilityEvent respectively. Many of the abovementioned clases have a strong time component divided in three hours, six hours or twenty four hours. For example, the class SixHourMinimumTemp models the minimum temperatures occurred in the previous 6 hours in Celsius degrees . The PressureEvent class contains the average sea level pressure registered by the station. Also, it is possible to qualify the weather to describe it with an appropriate descriptor.
- **Forecast ontology (FO)**[16]: Small ontology dedicated to forecasting and created by the University of Edinburgh. The ontology presents four basic classes for forecasting: (i)The class Forecast, which represents a particular weather forecast, contains four subclasses related to aviation, general public, mountain and shipping purpose. (ii) The class TimeFrame modelled the temporal scope of a forecast. Finally, location is represented by the continents (Asia, America, Europe) within the class World, for which the forecast is given. This FO ontology does not reuse any of the existing vocabularies and thus, reusing it with

[13] http://kmi-web05.open.ac.uk/WatsonWUI/
[14] http://swoogle.umbc.edu
[15] http://www.csd.abdn.ac.uk/research/AgentCities/
[16] http://homepages.inf.ed.ac.uk/stephenp/OWL/ForecastOntology.owl

Table 2

Ontologies modeling phenomena in meteorology

Feature	Ontology URL
Temperature, weather	`http://www.csd.abdn.ac.uk/research/AgentCities/WeatherAgent/weather-ont.daml`
Forecast	`http://homepages.inf.ed.ac.uk/stephenp/OWL/ForecastOntology.owl`
Rainfall	`http://metadata.net/WildNET/Climate.owl`
Phenomena	`http://sweet.jpl.nasa.gov/ontology/phenomena.owl`
Semantic sensor	`http://www.w3.org/2005/Incubator/ssn/wiki/Report_Work_on_the_SSN_ontology`

other purpose is difficult, even in the same domain.

- **Rainfall ontology (RO)**[17]: Lightweight ontology ontology dedicated to rainfall and contains three basic classes: ClimateReading, ClimateSensor and Location. It models the climate data, the information about sensors and the location of the site used for the prediction.
- **Phenomena ontology (PO)**[18]: Ontology developed within the Nasa project Semantic Web for Earth and Environmental Terminology (SWEET)[19]. It consists on around 6000 concepts in 200 separate ontologies. The PO ontology contains 254 classes and 14 object properties. However, only a few classes are really interesting to the purpose of the AEMET data, these are AtmosphericPhenomena, WeatherPhenomena, Pressure, Rainfall and Temperature.

In Table 2 the URIs of the relevant ontologies obtained after the searching operation are listed. The ontologies mentioned above are rather too large to be used (case of PO) or targeted to forecasting or rainfall. Instead, our focus is on modeling data observations gathered from weather stations, reason why this ontologies were not used for building our model.

Besides, the Semantic Sensor Ontology provided by the W3C Semantic Sensor Network Incubator Group more appropriated as the ontology can be used for sensor perspective (with a focus on what senses, how it senses and what it senses) as well as for data or observation perspective (with a focus on observations and related metadata) or a combination of both. Table 2 lists the URIs of the most relevant ontologies obtained after using the SSEs and exploring the main literature.

6.2. AEMET Ontology development process

The development of the AEMET ontology network has been performed following an iterative approach based on the reuse of existing knowledge resources, both ontological (including ontologies and Ontology Design Patterns[20]) and non-ontological resources as proposed by the NeOn methodology [12]. This approach has allowed the generation of a first prototype in a time-saving way. At the moment of writing this document, three iterations have been carried out over the prototype according to requirements in AEMET.

The current version of the AEMET ontology network follows a modular structure [13] consisting of a central ontology [21] that links together a set of ontologies that describe different sub domains involved in the modelling of the meteorological measurements. Figure 1 presents the main classess and properties of the this ontology network. This model contains four modular ontologies: Sensor, Time, Location, and Measurement.

The ontology network has been implemented in OWL DL and contains 83 classes, 102 object properties, 80 datatype properties, and has a SROIQ(D) expressiveness. Next, we present a brief description of each of the sub domains, as well as the knowledge resources that were reused in each ontology.

- **Measurement ontology**. It models the knowledge related to meteorological observations. Main concepts modelled in this ontology are: “ssn: Observation”, “ssn:FeatureOfInterest” and “ssn: Property”, reused from the SSN ontology. These concepts are related through the relationships “ssn:featureOfInterest” (relating “ssn:Observation” to “ssn:FeatureOfInterest”) and “ssn:hasProperty” (relating “ssn:FeatureOfInterest” to “ssn:Property”). As part of the customization of the SSN ontology to a particular use case, the concept “ssn: Property” has been extended and populated ac-

[17] `http://metadata.net/WildNET/Climate.owl`

[18] `http://sweet.jpl.nasa.gov/ontology/phenomena.owl`

[19] `http://sweet.jpl.nasa.gov/`

[20] `http://ontologydesignpatterns.org`

[21] `http://aemet.linkeddata.es/models.html`

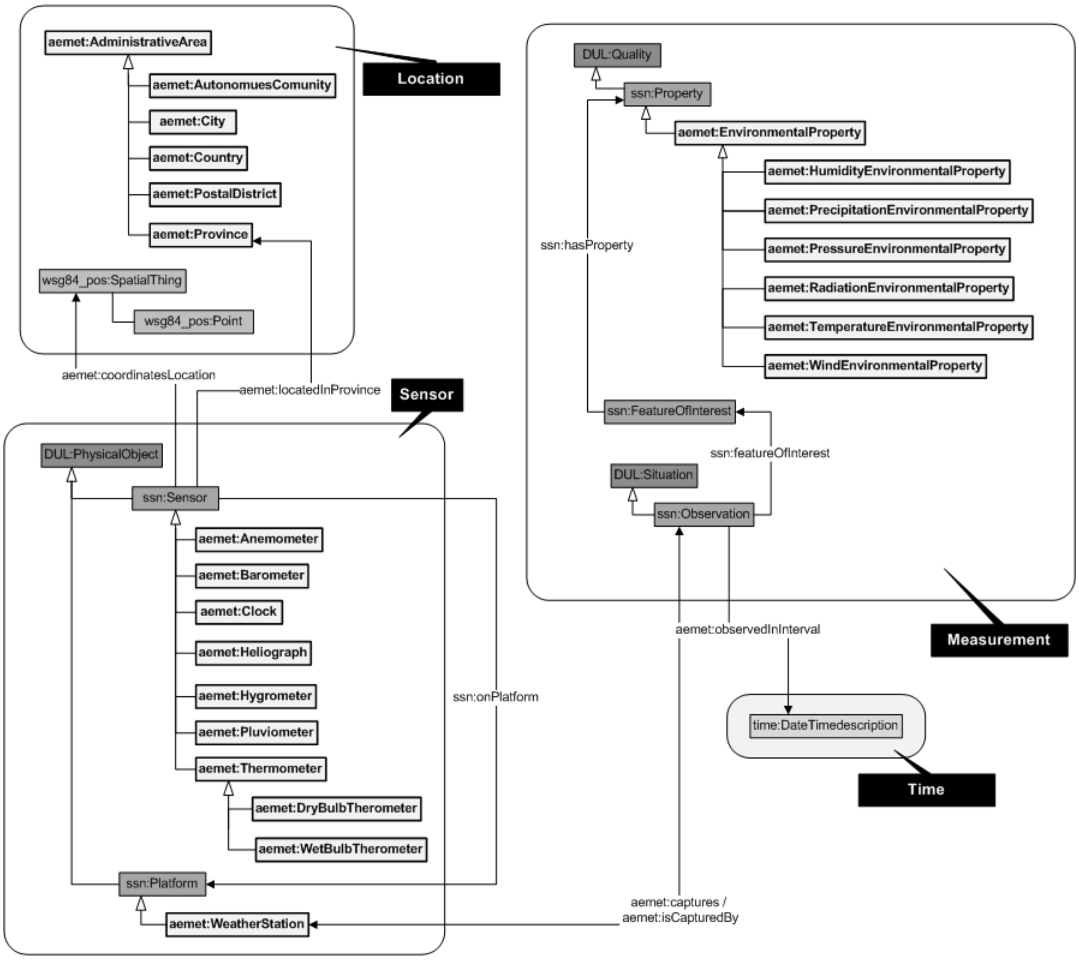

Fig. 1. AEMET ontology network detailed model

cording to the specific properties that AEMET gathers in its observations. For this purpose, we have used the tool NOR$_2$O[22] to transform the non-ontological resource "Describe_VAR.csv" file provided by AEMET to an ontological resource.

- **Location ontology**. It models knowledge about locations, such as administrative limits and coordinates. The wgs84_pos vocabulary developed by the W3C Semantic Web Interest Group[23] has been reused with the aim of supporting the representation of geometric positioning by means of the concept "Point". Also, a mapping between the relationships "aemet:locatedIn" and "wgs84_pos:location" has been defined. In addition, we reused part of the Geobuddies ontology network[24] to represent knowledge about the Spanish administrative division. Concretely, we reuse the concept "AdministrativeArea", its subclasses, and the mereological relationships and axioms between them.
- **Time ontology.** It models knowledge about time such as temporal units, temporal entities, instants, intervals, etc. This ontology has been mainly developed by reusing the OWL Time ontology, which is a reimplementation of the *DAML time* ontology[25] carried out by the Semantic Web

[22] http://mayor2.dia.fi.upm.es/oeg-upm/index.php/en/downloads/57-nor2o

[23] http://www.w3.org/2001/sw/interest/

[24] http://mayor2.dia.fi.upm.es/oeg-upm/index.php/en/ontologies/83-geobuddies-ontologies

[25] http://www.cs.rochester.edu/~ferguson/daml/

Best Practices and Deployment Working Group[26] (SWBPD) from the W3C. At this stage of the project only the concept "DateTimeDescription" and its properties are being used with the aim of representing the time when an observation has been measured.

- **Sensor ontology** The modelling of the network of sensors and weather stations was based on the reuse and extension of the Semantic Sensor Network Ontology (SSN), developed by the W3C SSN incubator group[27]. SSN supports the description of the physical and processing structure of sensors. Here, sensors are not constrained to physical sensing devices. Instead, a sensor is anything that can estimate or calculate the value of a phenomenon, so as a device or a computational process or a combination of both. The representation of a sensor in the ontology links together what it measures (the domain phenomena), the physical sensor (the device) and its functions and processing (the models). The ontology can be used for a focus on any (or a combination) of a number of 4 perspectives: (i)A sensor perspective, with a focus on what senses, how it senses, and what is sensed; (ii) A data or observation perspective, with a focus on observations and related metadata; (iii) A system perspective, with a focus on systems of sensors; and (iv) A feature and property perspective, with a focus on features, properties of them, and what can sense those properties. The main classes of the Semantic Sensor Network ontology have been aligned with classes in the DOLCE Ultra Lite (DUL)[28] foundational ontology to facilitate reuse and interoperability. During the aligment process between the SSN ontology and the AEMET ontology network the concept “ssn:Sensor” has been extended by means of a hierarchy of types of sensors used by AEMET. Also, the “ssn:Platform” concept has been extended by means of the “aemet: WeatherStation” concept.

7. Exploitation

The potential uses and exploitation triggered by the availability of the dataset online have been studied by the group, analyzing how it could be useful for different kinds of users in the Web. This section describes how the observations can be retrieved and filtered through SPARQL queries to the endpoint, that otherwise would require users to spend much time regarding downloading and processing the public files in AEMET’s home page. It also describes the resulting application using those queries, designed for browsing and visualizing the contents of the dataset.

7.1. Retrieving the contents from the observations

Thanks to the modeling proposed in section 6, each observation can be accessed by any of its attributes. For example, given the location of a weather station and a date and an hour, we could retrieve the observations for that exact time with the following query:

```
PREFIX xsd:
<http://www.w3.org/2001/XMLSchema#>
PREFIX ssn:
<http://purl.oclc.org/NET/ssnx/ssn#>
PREFIX aemet:
<http://aemet.linkeddata.es/ontology/>
PREFIX w3ctime:
<http://www.w3.org/2006/time#>
PREFIX geo:
<http://www.w3.org/2003/01/geo/wgs84_pos#>

SELECT distinct ?station ?obs ?est ?prop ?value
?q ?h ?min ?dia ?mes ?anno
WHERE
{
?station geo:location
<GeoPosURI> .
?station aemet:nombreEstacion ?est .
?obs ssn:observedBy ?station .
?obs ssn:observedProperty ?prop .
?obs aemet:valorDelDatoObservado ?value .
?obs aemet:calidadDelDatoObservado ?q .
?obs aemet:observadaEnIntervalo ?inter .
?inter w3ctime:hasBeginning ?instant .
?instant w3ctime:inDateTime ?dateTime .
?dateTime w3ctime:hour ?h .
?dateTime w3ctime:minute ?min .
?dateTime w3ctime:day ?dia .
?dateTime w3ctime:month ?mes .
```

[26] http://www.w3.org/2001/sw/BestPractices/

[27] http://www.w3.org/2005/Incubator/ssn/wiki/Report_Work_on_the_SSN_ontology

[28] http://www.ontologydesignpatterns.org/ont/dul/DUL.owl

```
?dateTime w3ctime:year ?anno .
?dateTime w3ctime:inXSDDateTime <myDateTime> .
}
```

The query also retrieves the name of the weather station measuring the observation, the measured property, the value of the measure and the quality of the value. <GeoPosURI> would correspond to the geo:Point with latitude and longitude of the station (e.g. http://aemet.linkeddata.es/resource/Point/GeoPos_08210), and <myDateTime> would be the date at which we are interested to recover observations, in xsd:DateTime format (e.g. "2011-05-27 01:40:00 ").

Additional filters can be added, e.g. if the observations to retrieve from a weather station must be during a time interval, the final part of the query can be replaced with:

```
?dateTime w3ctime:inXSDDateTime ?dt .
FILTER
(?dt >= xsd:dateTime("timeIntervalStart")). FILTER
(?dt <= xsd:dateTime("timeIntervalEnd")).
```

Or if the objective is to recover a single property of the weather station, the next SPARQL fragment can be added to the query:

```
?obs ssn:observedProperty ?prop .
?prop a <desiredProperty> .
```

Where <desiredProperty> could be, for example, the velocity of the wind in m/s (http://aemet.linkeddata.es/resource/WindAmbientProperty/VV10m). Other possibilities are filtering by the quality of the observation (the numeric "q" value in the query), proximity to a location (by comparing latitudes and longitudes), etc.

7.2. An application for representing and retrieving metereological data

We have built a simple application that consists on a viewer based on a modified version of map4RDF[29] [5], a tool that uses Google's Web Toolkit framework[30] to visualize and access to Linked Data resources.

Each automatic weather station is geolocated, enabling its representation in the map with a marker. Since there are approximately 300 stations, they can be all represented at the same time. When a user clicks on one of the markers, a new pop up window appears showing the last stored measure and the value of each of its observations (wind, temperature, pressure, hour of the measurement, etc.). The value of the last measurement is retrieved dynamically through a SPARQL query to the endpoint, enabling the application to show up an updated value at the specific point where it was measured. This point can be generalized to any measurement that is done at a time by a sensor which has a position.

The windows also show a menu to display the graphics of the values for an observation by the specific weather station. These graphics have been implemented using gfot[31] and incorporing it to the map4RDF application. The application retrieves the contents of the observation through SPARQL queries also, allowing to display various types of graphics (day, month, week).

An example can be seen in Figure 2, where we show the pressure of the station in Ciudad Real on time intervals of ten minutes. In the rest of the map we can also see the markers of the other available stations. The visualization in this way takes a small amount of assumptions regarding the data that is visualized, which is associated with a point in time and space. The geographical information allows to associate the information with the specific point in the map where it should be displayed, while the temporal dimension allows creating graphics with the evolution of that specific value during a period of time. More specific visualizations could be used, as colorings for the map or iso-lines are innumerable. The approach taken takes very few assumptions on data and is hence more general while keeping maintainability, being possible to add new visualizations for different types of data.

However, the true potential for the published dataset shows off when combining it with other available datasets in Linked Data. The first potential use is with the geolinkeddata dataset [5] with all the geospatial information that it contains (ranging from airports or rivers to villages and provinces). As an example, the next query would retrieve all the stations located in a province along with their label:

[29] http://oegdev.dia.fi.upm.es/projects/map4rdf/

[30] http://code.google.com/intl/es-ES/webtoolkit/

[31] http://code.google.com/p/gflot/

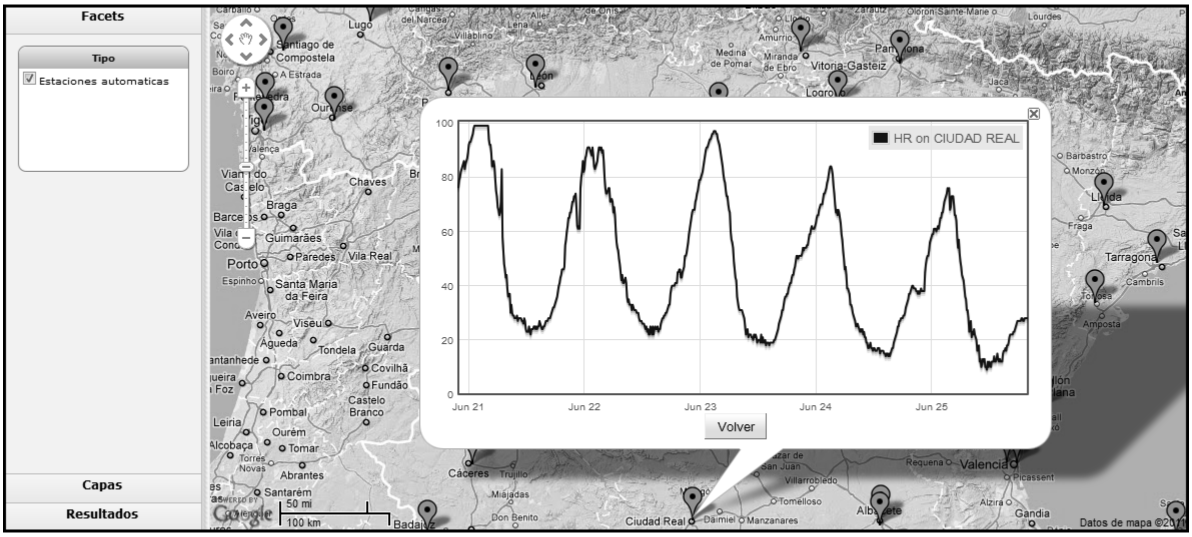

Fig. 2. Overview of the Metereological Linked Data Application.

```
select * where {
SERVICE<http://aemet.linkeddata.es/sparql> {
?s <http://aemet.linkeddata.es/ontology/isLocatedIn>
?provincia }
SERVICE<http://geo.linkeddata.es/sparql>{
?provincia <http://www.w3.org/2000/01/rdf-schema#label>
?label .
?provincia <http://www.w3.org/1999/02/22-rdf-syntax-
ns#type>
<http://geo.linkeddata.es/ontology/Provincia>
}
}
```

The query could be modified easily to provide the measurements from the weather stations close to a lake, river, beach, etc. This is one of the main advantages of Linked Data. The integration with other datasets, thanks to the semantics provided by the links, is to some extent automatic. Reusing the data is easier as well, as compared with other formats, which should offer equivalent semantics and have a purpose as general as Linked Data to provide the same advantages in such a broad domain.

8. Conclusions

In this paper we have presented the process followed to transform some of the public meteorological data from AEMET resources into Linked Data. To do so, we have reused ontologies like the time ontology and the SSN ontology, which are widely used in other systems, and we have shown how to retrieve the data, create and update the triples automatically, and show the data in a generic visualizer that considers the geoposition of that data. This process has followed an agile approach, and it could be easily adapted to other datasets with similar data, like sensor networks, or in other domains, as far as the focus is set on the geographical component of the data. This is because the assumptions made on the properties of the data presented are very few, as are the assumptions regarding the source format. For the former, the spatial nature of the data allows the map visualization, and the temporal nature of it allows to show the evolution through time graphically. For the latter, anything that can be parsed in a similar way to a .csv file can be processed similarly, since the scripts have been built to be generalizable. Finding the right spot between automation and freedom is still an open line of research. The advantages for the regular user may not be obvious, as it is the development time invested to reach results that may look similar, but by using semantic technologies and well established standards the development time is reduced, and the products obtained are more easily maintainable and reusable.

Other systems can consume this data and offer visualizations, aggregations, reason over it, etc. Once the data has been extracted and is offered as Linked Data, applications that use Linked Data can instantly benefit from it. Besides, it allows using the data instantly without having to browse, download and read all the .csvs published daily by AEMET, providing a much more convenient access to the same information. Moreover, the advantages of publishing the dataset as linked data allows modularity and reusability, for example less time would be required for the development of applications based on this data (as we have already shown with the visualizer). Finally, the ontology we used to model the data can be aligned with other ontologies (time, ssn, etc.). By complying with well established standards interoperability is enhanced, for instance within other sensor networks and applications using these ontologies, and the linking in other domains, such as medical, geographical, biological, etc.

Some limitations have been detected during the agile development process, specially in the application of evolutionary prototypes in a process that depends on immutable URIs. This limitation can be overcome in two different ways. As a first and more methodological approach, design and specification of URIs can be prioritized in the whole process. As far as URIs are stable there should be no problems with all the products that depend on them. This is a feasible task when considering a specific methodology to define URIs. As seen on Section 4 this is a challenge that remains open, although a noticeable progress can be observed in this paper. The second and more technical solution consists on developing systems that are aware of the URIs in some way so that changes can be propagated to all relevant instances.

9. Acknowledgments

This work has been supported by the Spanish project myBigData. We would like to kindly thank Alexander de León, Idafen Santana, Miguel Angel García, Raúl Alcazar, and Soledad Hurtado for their help.

References

[1] N. Aswani, K. Bontcheva, and H. Cunningham. Mining information for instance unification. In I. Cruz, S. Decker, D. Allemang, C. Preist, D. Schwabe, P. Mika, M. Uschold, and L. Aroyo, editors, *The Semantic Web - ISWC 2006*, volume 4273 of *Lecture Notes in Computer Science*, pages 329–342. Springer Berlin Heidelberg, 2006.

[2] L. M. V. Blázquez, B. Villazón-Terrazas, V. Saquicela, A. de León, Ó. Corcho, and A. Gómez-Pérez. Geolinked data and inspire through an application case. In *GIS*, pages 446–449, 2010.

[3] D. Booth. URIs and the Myth of Resource Identity. In *Workshop on Identity, Meaning and the Web (IMW06) at International World Wide Web Conference. (Edinburgh, Scotland. 2006) ACM*, 2006.

[4] J. de Bruijn and A. Polleres. Towards an ontology mapping specification language for the semantic web. In *DERI Technical Report 2004-06-30*, 2004.

[5] A. De León, V. Saquicela, L. M. Vilches, B. Villazón-Terrazas, and F. Priyatna. Geographical linked data : a spanish use case. In *I-SEMANTICS 6th International Conference on Semantic Systems*, September 2010.

[6] G. Fliedl, C. Kop, and J. Vöhringer. From owl class and property labels to human understandable natural language. In Z. Kedad, N. Lammari, E. Métais, F. Meziane, and Y. Rezgui, editors, *Natural Language Processing and Information Systems*, volume 4592 of *Lecture Notes in Computer Science*, pages 156–167. Springer Berlin Heidelberg, 2007.

[7] A. Jaffri, H. Glaser, and I. Millard. Uri disambiguation in the context of linked data. In *In Linked Data on the Web (LDOW2008)*, 2008.

[8] K. Page, D. D. Roure, K. Martinez, J. Sadler, O. Kit, K. Taylor, A. Ayyagari, and D. D. Roure. Linked sensor data: Restfully serving rdf and gml. In *Proceedings of the 2nd International Workshop on Semantic Sensor Networks (SSN09) at ISWC 2009*, volume 522, pages 49–63. CEUR Workshop Proceedings, November 2009.

[9] H. Patni, C. Henson, and A. Sheth. Linked sensor data. In *2010 International Symposium on Collaborative Technologies and Systems*, pages 362–370. IEEE, May 2010.

[10] D. L. Phuoc and M. Hauswirth. Linked open data in sensor data mashups. In A. A. Kerry Taylor and D. D. Roure, editors, *Proceedings of the 2nd International Workshop on Semantic Sensor Networks (SSN09) ISWC 2009*, volume Vol-522, pages 1–16, Washington DC, USA, 10 2009. CEUR Workshop Proceedings, CEUR.

[11] D. Schober, B. Smith, S. E. Lewis, W. Kusnierczyk, J. Lomax, C. Mungall, C. F. Taylor, P. Rocca-Serra, and S.-A. Sansone. Survey-based naming conventions for use in OBO Foundry ontology development. *BMC Bioinformatics*, 10(125), 2009.

[12] M. C. Suárez-Figueroa. *NeOn Methodology for Building Ontology Networks: Specification, Scheduling and Reuse*. PhD thesis, Universidad Politécnica de Madrid, Spain, June 2010.

[13] M.-C. Suárez-Figueroa, S. Brockmans, A. Gangemi, A. Gómez-Pérez, J. Lehmann, H. Lewen, V. Presutti, and M. Sabou. D 5.1.1 neon modelling components. Available at: http://www.neon-project.org, March 2007.

[14] B. Villazón-Terrazas, M. C. Suárez-Figueroa, and A. Gómez-Pérez. A Pattern-Based Method for Re-Engineering Non-Ontological Resources into Ontologies. *International Journal on Semantic Web and Information Systems*, 6(4):27–63, 2010.