# Linked Spatial Data for location-aware services

Rodríguez, Aimar
aimar.rodriguez@deusto.es

Eduardo Castillejo
eduardo.castillejo@deusto.es

López-de-Ipiña, Diego
dipina@deusto.es

June 19, 2014

**Abstract**

Geospatial data is becoming more common in the last years. Several platforms on the web offer spatial data through APIs and open datasets. This data has te potential to be used to build powerful location based services when merged into a uniform dataset. This paper describes the SORELCOM platform, designed to merge data from heterogeneous web platforms into a single uniform dataset, in order to build a rich service for end user and to facilitate the usage of this data to other developers, by offering a repository of spatial data created following Linked Open Data best practices.

keyword: Linked Open Data, Semantic Web, Spatial Data, Geosparql, GIS

## 1 Introduction

Geospatial data is becoming more and more relevant in our world. In roughly 30 years, we have evolved from a region based spatial information, to human and device centric information. The amount of location-aware devices has multiplied and with this, location-based services are thriving. Users can retrieve directions and information based on their current location, the number of social media users which set their devices to include location information on their phones is growing. There is no doubt that the amount of spatial data is increasing and that it is making an impact in our societies.

As a consequence, the amount of Linked Open Data with spatial context made available on the web has also increased. The principles of Linked Open Data[2] encourage the usage of the RDF and SPARQL to query and model data. This is useful for querying for the explicit relations on the datasets, however, the implicit spatial relationships cannot be easily queried.

To fight this issue, the Open Geospatial Consortium[1] (OGC) has defined a standard, named GeoSPARQL to address the issues of spatial information representation and querying on the semantic web. Thanks to this, it is possible to include a spatial factor in the reasoning done over RDF datasets.

The platform SORELCOM (Social Routes Empowered by Linked Contents and Context Mining) aims to use GeoSPARQL to create a user-centred GPS community which is able to offer location aware service to users and to publish the data generated in a spatially aware linked dataset.

The platform aims to combine semantic and spatial technologies to create a tool to capture routes and enrich their information from sources around the web, providing in addition a powerful location and preference based recommendation service.

In the end, the goal is to make the users more aware of their surroundings, merging heterogeneous data found on the world wide web onto a uniform model and publishing the data, thus facilitating the development of third party spatial aware applications.

## 2 Related Work

GPS communities such as Wikiloc[2] and EveryTrail[3] allow users from around the world to share their routes, whether they are touristic, sports trails or of any other kind. However, due to the heterogeneity of the data found around the web it is quite costly to exploit the diversity of data that can be found on these platforms.

The usage of RDF data, following the principles of Linked Open Data, could bring a solution to this, allowing the creation of rich datasets to empower location based services. However, even in the Semantic Web, there is still a certain amount of segregation regarding the expression of spatial data. The Basic Geo vocabulary[3] allows a very basic representation of coordinates, however, it is hardly possible to compute spatial reasoning over the data. Triplestores such as AllegroGraph[1] or Strabon[7] offer their own, more powerful, vocabulary for the encoding of the geospatial data, as well as means to add spatial context to queries, however, the difference between the existing vocabularies does not help when it comes to creating aggregations or services based on Linked Data.

The OGC established the standard GeoSPARQL to solve this problem. It offers a very basic representation of spatial data, based on the concepts of *Feature* and *Geometry*, as well as a way of encoding the spatial information using the WKT and GML standards, also from the OGC. Additionally, a series of functions and query transformation rules are provided to allow spatial queries

[1] http://www.opengeospatial.org/
[2] http://en.wikiloc.com/
[3] http://www.everytrail.com/

using SPARQL.

To store the spatial data, several serialization formats exist currently. The GPS eXchange Format (GPX)[6] is the most widely used among GPS communities for several reasons. It encodes information directly recorded from a GPS into a XML formatted file. This format specifies how to represent routes and points, and delivers additional information about the recording, such as the time when it was made. This data contains a good amount of information, but it is hard to encode on a spatial database and costly to send over a network. The GeoJSON[4] [5] format specifies how to represent geospatial data in the JavaScript Object Notation (JSON) format, in a manner similar to that used by GeoSPARQL, that is, features and geometries. In addition, the OGC is working on including the GeoJSON serialization format on their standard, as well as a series of rules to transform the spatial data found in these files into RDF.

However, spatial information by itself is not enough to build a user-centric location based service, it is necessary but not sufficient. A problem persistent through GPS communities is evaluation of the technical difficulty of a route, for this information is crucial when deciding which one to show or recommend to a particular users. Generally the approach has been simply allowing the author of the route give a subjective qualitative qualification, for example, "hard". Seen the problem a subjective qualification brings, the ibpindex (http://www.ibpindex.com/) team, has developed a system to give a objective evaluation of GPX files using mathematical methods. This way, it is possible to obtain a score that can be easily interpreted by machines and humans. Combining these mathematical methods with user-defined interests or preferences, powerful recommendation services can be built.

Definitely, there is a good amount of work being done regarding geospatial data, and there is no doubt that GIS systems are thriving. In our vision, this efforts can be combined in a single system to create a powerful location-aware service to aggregate the data spread over the web in a single semantic model compliant with the principles of Open Linked Data.

# 3 SORELCOM

The first prototype of the SORELCOM system is composed of the following elements:

- A mobile application, which allows users to browse the routes and points of interest on the dataset, to record and upload their own routes, as well as receiving notifications of the nearby features in real time. This application

[4]http://geojson.org/

will also be used to develop the concept of *Geo Post-it*, detailed in section 3.2.

- A web application providing the same browsing functions as the mobile one, as well as the ability to import already existing GPX files and editing them before finally storing them in the data store.
- A server located between the web and mobile applications and the data store, which takes care of answering the queries of the applications (as a traditional web server) and of querying other platforms and merging the data to the dataset.
- A public API and SPARQL endpoint, to allow developers of other spatial application to reuse the data gathered on the system.

The platform operates similarly both on the web and mobile application. The user can browse any data in the application without any need for registration, however, if it wishes to contribute it needs to register through the application. Once this is done the user can import their GPX files from other platforms, for instance Wikiloc, or record their own routes from the mobile application. The SPARQL endpoint will be publicly accessible in read-only function, if a third party application want to write they will need to use the API, for which registration is needed.

The usage of RDF data allows to create a complete service for the search and retrieval of data, as well as to link the data on the platform to other datasets, however the main benefit is another one. By using SPARQL on the platform it is possible to construct a very powerful route recommendation service for the platform, which can take into account the preferences of the user, the semantics of the data and the spatial characteristics of the features.

## 3.1 Linked data

It is necessary to model the data on the application following some RDF specifications, in order to publish it as Linked Open Data. The idea behind this is to use previously defined reusable vocabularies such as the Friend of a Friend[4] (FOAF) vocabulary, which defines the concept of Person and related properties, along a particular ontology used to define the specifics of the systems. To achieve this, 3 ontologies will be reused: The Dublin Core (DC) ontology, used to define metadata; the GeoSPARQL vocabulary, which defines spatial objects and their relations and the FOAF vocabulary for the representation of users.

The SORELCOM ontology (figure 1) defines three classes for the representation of Features as well as two other classes for the media associated with the feature. First, any data generated on the platform is considered a `Feature`, which is an object with a spatial representation, however, only three types of

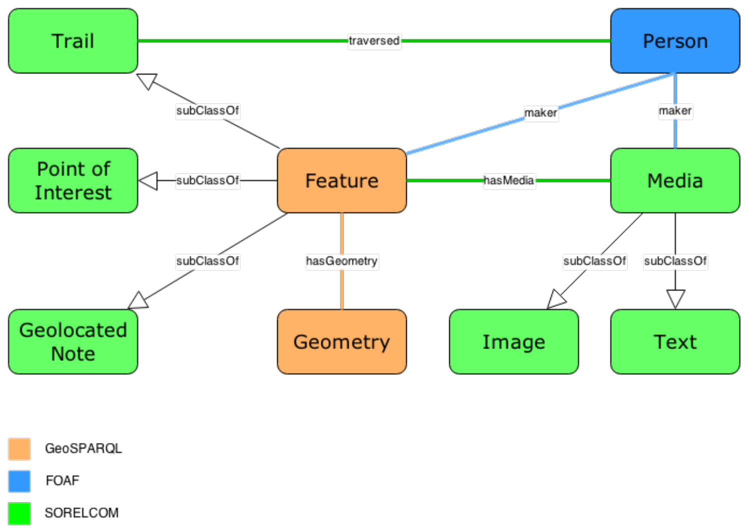


Figure 1: SORELCOM RDF data model

features belong to the model, defined by the classes `Trail`, `PointOfInterest` and `GeolocatedNote`. Since we believe the information of a feature should be represented in multimedia format (images of a point of interest, comments of a route, videos of a route, etc.) there is also a `Media` class defined in relation to the features. On this first prototype, media can be either of type `Text` or `Image`.

In addition to the classes representing the resources that will be stored on the platform, several properties that define their characteristics are defined. The `difficulty`, `distance` and `slope` properties of a trail are stored among others. Besides, since the data on the platform is user generated, there is a need to keep track of the creators of the data, for which DC and FOAF relations are used.

### 3.2 Geo Post it

A Geo Post it is a geolocated note with a temporal characteristic, that is, a piece of multimedia (text, image or video) information located in a certain coordinates which is only valid for a limited time. The idea behind this notes is to bring a new way of communicating for the users of the platform. Imagine a case of a cyclist following a trail with a group of sports mates and the cyclist takes the lead to the point where the rest of the group lose track of his whereabouts. In this use case, the user would be able to post a note when reaching a bifurcation indicating the way that his mates should take, by being notified on their phones when arriving there.

Of course, the uses of geolocated post-its are much wider, even located adver-

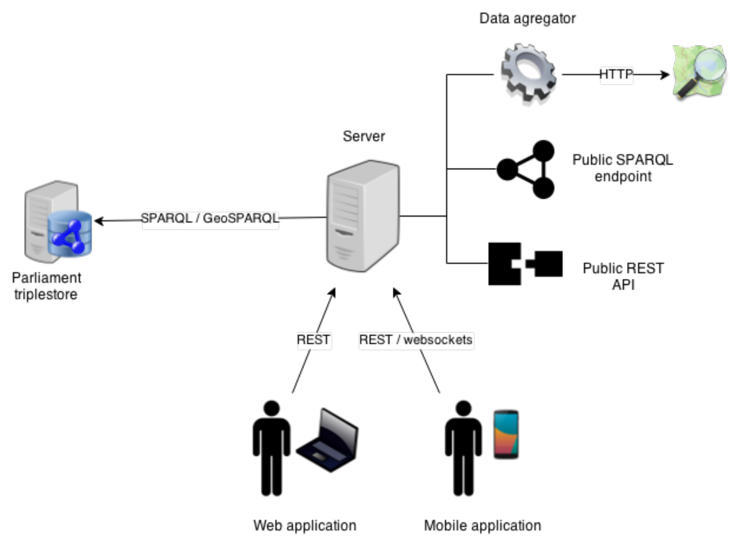


Figure 2: High-level architecture of SORELCOM

tisement. In the end, it aims to give a spatial context to the opinions, warnings and other kinds of useful information that users can generate on the web. Since it is easy to get lost in the huge amounts of information that can be generated, it is possible to create a location based filter for this information with these located notes.

However, while some notes will be made public so that everyone can receive them, others will be made private by the user, so that only a certain person or group can read them.

In SORELCOM, geo post-its are put in practice using the mobile application. A registered user will simply need to write a note, take a photo or record a video (or all of them) and specify the a degree of visibility, that is, who can see the note. Then the application will create a Geolocated Note in their current location.

### 3.3 Implementation Details

The architecture of the SORELCOM platform is shown in figure 2 at a very high level. The web and mobile application both access the information through an internal REST API offered by the server, as well as using some pieces of the public API. In addition, in order to offer real time services, the mobile application can establish persistent connection with the server using the HTML5 websocket protocol.

The server, in addition offers it's data to third parties through two access

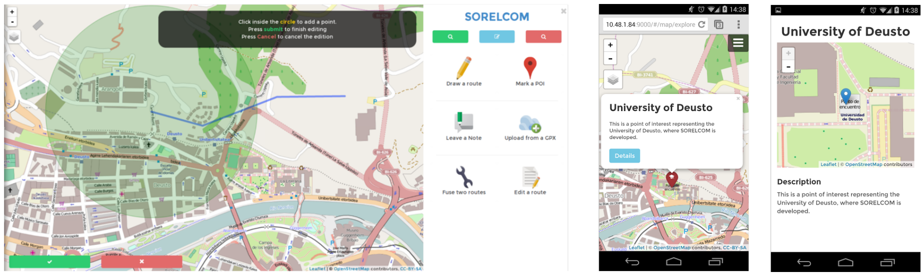


Figure 3: SORELCOM service from desktop and mobile

points. The first of them is a public API which follows the REST standard and allows access to features and users in the dataset as well as some predefined spatial operations. The second access point is a SPARQL endpoint which will route its queries to the database and will return the results. This is done to filter out any attempt of using the endpoint for modifications.

All APIs, both internal and external, returns the results of the queries in GeoJSON format in case of querying for spatial data and in plain JSON format in the rest of the cases. In the future, the public API will be extended to offer results in other formats, mainly GPX.

The platform uses a Parliament$^{TM}$[5] triplestore on the back-end. This data store implements the GeoSPARQL and thus it is among the few triplestores which have te ability to make spatial queries. Only the server will communicate with the database, and will do so by using SPARQL for the read operations and the SPARQL update/modify protocol for the write operations.

One final piece of the system worth mentioning is the data aggregator. The aggregator acts independently of the rest of the server and is used to collect data from other platforms. The task of this component is to make HTTP queries to the different APIs that offer spatial data around the web, mainly OpenStreetMap[6] and Geonames[7], transform that data to a RDF format and aggregate it to the datastore of the platform. This way, it is possible to use information obtained from third parties to enrich the user-generated data.

The SORELCOM server has been developed using NodeJS[8] on the server and Phonegap[9] on the mobile application.

At the development time, the final user has always been the focus of the platform, thus the interfaces have been developed to be as easy to use and interpret as possible. In order to do this with spatial data, we have developed an interactive map using Leaflet, which can be viewed from mobile and desktop

[5]http://parliament.semwebcentral.org/
[6]http://www.openstreetmap.org
[7]http://www.geonames.org
[8]http://nodejs.org/
[9]http://phonegap.com/

```
SELECT DISTINCT ?trail WHERE {
        my:exampleUser foaf:knows ?known .
        ?known sorelcom:hasTraversed ?trail .
        ?trail a sorelcom:Trail ; geo:hasGeometry ?tg ;
           sorelcom:difficulty ?difficulty .
        ?tg geo:asWKT ?twkt .
        BIND(geof:distance(?twkt,
           "POINT(-2.922424 43.261433)"^^geo:wktLiteral,
           units:metre)
           AS ?distance) .
        FILTER(abs(?difficulty -50) < 25) .
} ORDER BY ?distance
```

Figure 4: Possible query to obtain recommendations

browsers. All the operations, including the editing and the hand-drawing of new trails can be done on the map itself through a point-&-click interface. By using this interface, as shown in the examples in figure3 the interpretation of the spatial data is immediate and there is no need for the user to have any geographic knowledge to manipulate it.

One of the key points of SORELCOM is the ability to provide the users with personalized recommendations. This recommendations are based on several factors, however, the main weights are proximity and difficulty. The difficulty of a route is calculated using a mathematical model which obtains the slopes and the distance for the whole route and then obtains a score from 1 to 100. The more slopes there are in the route and the harder they are, the higher the difficulty will be.

Equation 1 shows the formula for calculating the score of a trail, considering l the maximum slope considered (by default 20), $d$ the total distance of the route and S the list containing the distance traversed in slope, that is, the distance when the slope is of 1%, of 2%, etc.

$$\sum_{i=1}^{l}(i/l) * (100/d) * S_i \tag{1}$$

Once the difficulty is obtained, it is just stored on the database, where queries can be made. By combining this difficulty attribute with spatial queries it is possible to build powerful recommendation systems. For example, the SPARQL query on figure 4 obtains all the trails that the people a user knows have traversed, filter only the ones with a similar difficulty to the average of the trails of the user and order them by distance. This way it is possible to obtain a list of potential recommendation candidates.

Of course, there are many more factors to take into account, such as which routes has the user traversed already, the code on figure 4 is just an example. Besides, this recommendation idea can also be extended to the points of interest, using the information about the interests of each user.

This way, a location aware service which provides personalized information about their surroundings is provided. Data from over the web and data provided by the users is merged and relations among them are found in order to create a rich dataset which developers can access to build their own applications.

## 4 Conclusions and Future Work

The SORELCOM platform will provide users a easy to use system to enrich their knowledge about their surroundings in a personalized manner. Developers will be able to use the data and functions offered by the system to create their own services or to aggregate the information to their own datasets. Agents around the web will be able to reason spatially on the data, as is the goal of the Semantic Web. Last but not least it is important to mention, the importance of developing Semantic Geospatial services, for even if the technology is ready to create this kind of systems, they are not nearly as popular as non semantic GIS systems.

Of course there is still much work to be done. First, the current ontology expresses a minimal set of concepts, very specific to the application. This ontology will be further improved in the future, for example, providing a extensible way of representing Point of Interest categories (churchs, monuments, etc). OpenStreetMap offers a wide and quite complete taxonomy for classifying Points of Interest, which can be used to define the categories in the vocabulary. With new data on the model, the recommendation algorithms will be improved.

In addition to this, there are plans to use other services provided by already existing platforms to enhance the functionality offered on the service. One example of this is using the geocoding API offered by Geonames (in addition to its data) to offer utilities to users when searching and creating features.

Finally, the data on the web has been used to the benefit of SORELCOM, so we believe that the platform can also give something back. Future plans include providing other platforms, such as OpenStreetMap, with the user generated information in the systems.

## References

[1] Jans Aasman. *Allegro Graph: RDF Triple Database.* Technical report. Technical Report 1, Franz Incorporated, 2006.

[2] Christian Bizer, Tom Heath, and Tim Berners-Lee. “Linked data-the story so far”. In: *International journal on semantic web and information systems* 5.3 (2009), pages 1–22.

[3] Dan Brickley. “Basic geo (WGS84 lat/long) vocabulary”. In: *Collaborative informal document* (2006).

[4] Dan Brickley and Libby Miller. “FOAF vocabulary specification 0.98”. In: *Namespace Document* 9 (2012).

[5] Howard Butler, Martin Daly, Allan Doyle, Sean Gillies, Tim Schaub, and Christopher Schmidt. *The GeoJSON Format Specification*. 2008.

[6] Dan Foster. “GPX: the GPS exchange format”. In: *Retrieved on* 2 (2007).

[7] Kostis Kyzirakos, Manos Karpathiotakis, Konstantina Bereta, George Garbis, Charalampos Nikolaou, Panayiotis Smeros, Stella Giannakopoulou, Kallirroi Dogani, and Manolis Koubarakis. “The spatiotemporal RDF store strabon”. In: *Advances in Spatial and Temporal Databases*. Springer, 2013, pages 496–500.