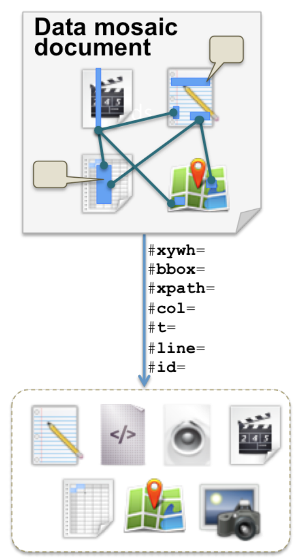
Stitching data mashups from geodata fragments

Philippe Duchesne
phd@highlatitud.es

Abstract. The ongoing CrossLinks project aims at building a data mashup standard and platform. This paper presents issues and solutions that emerged in the scope of this project, with respect to geospatial data. In particular, special attention is given to GIS services, as these are used for a large part of online geodata. As the project focuses on the extended use of URI fragments and the definition of a representation format for so-called data mosaics, it will highlight how existing GIS standards and practices fit with the chosen linked data approach.

Keywords. Linked Data, GIS, URI Fragments, Standards, Data mashups, Collaboration, Visualization

Overview



The CrossLinks project aims at developing both a representation format and a platform to author, exchange and visualize so-called data mosaics, i.e. aggregations of heterogeneous media fragments. The project is rooted in the recurrent need of selecting and stitching together various online data sources to produce contextualized dashboards.

This paper presents an overview of the main issues and solutions that emerged in the scope of this project, with respect to the handling of geospatial data sources. In particular, it will build on the notions of URI fragments and data mosaics developed in a previous paper [1], and extend them with geospatial considerations. The goal is to include geospatial data in dashboard documents just as the other media types (text, video, spreadsheets) while retaining and properly exploiting the geospatial dimension.

As a large part of online geospatial data is exposed through web services - in particular governmental data, as a result of directives such as INSPIRE in Europe - special attention is given to supporting such services in the developed solution.

Fragments of geospatial resources

The formalization of URI fragments is one of the building blocks of the project. The need for an explicit definition of URI fragment dimensions and their relation with mime types has been described in [1]. These fragments are required to be able to identify and extract data chunks from within a

URI-identified resource, and later be part of a data mosaic. Several types of fragments, corresponding to the possible dimensions of hypermedia resources (textual, temporal, pixelspace, geospatial, ...), have been proposed. Of these fragment types, two are of special interest when linking geospatial datasets : `bbox` and `id`. The `bbox` fragment obviously applies to resources with a geospatial dimension, while the `id` fragment applies to resources with named subentities, which is the case for many GIS resources.

`bbox` fragment

While the `bbox` fragment is potentially applicable to any geo resource, it is mostly useful to reference parts of geo raster data, as vector is more likely to be referenced by `id` fragments described below. Typical raster data targeted by `bbox` fragments are georeferenced images like GeoTIFF, and web services like WMS or WCS, as in:

```
http://vmap0.tiles.osgeo.org/wms/vmap0#id=basic&bbox=-6,49,5,54
```

In this example, also note the `id` fragment used to identify the WMS layer at stake.

Different fragment types supported by the resource can be combined. For example, with a KML document that can also have a temporal dimension :

```
http://example.org/routes.kml#bbox=-6,49,5,54&t=20,30
```

`id` fragment

When it comes to structured data, and in particular vector data, the `id` fragment is best suited to select any named entity within a given dataset. Resolution of the `id` fragment must be specified on a per-datatype basis : matching the `fid` in a GML document, the `id` in KML, as in:

```
http://example.org/placemarks.gml#id=placemark1
```

Some data structures require the ability to identify elements within a tree structure, i.e. not directly accessible by a single id. This is for instance the case for a WFS where features are part of feature types, yielding fragments such as :

```
http://demo.opengeo.org/geoserver/wfs#id=states/states.39
```

Fragments of data services - query vs fragments

In the examples above, fragments applied to web services look redundant with query parameters such as the `GetFeatureById` operation of a WFS, or the `bbox` parameter of most GIS services.

It must first be noted that the use of fragments is technically not equivalent to URL query parameters : while the latter is sent to and processed by the server, the first is meant only as a hint for the client and not guaranteed to be sent to the server. The optional query parameters should be opaque to the client, and the client should not be expected to understand the inner structure of a URL, up to the fragment part. Conceptually, this means query parameters take part in defining the resource that the viewing client is looking at, while the fragment is a hint at what part of that resource is actually of interest. How to interpret this hint is left at the discretion of the client.

In the context of data mosaics described here, it is important to be able to distinguish the complete resource and the fragment, as the first one provides contextual information and puts the resolved fragment in perspective. Taking a WFS as example, `http://mywfs.org#layer1/001` holds more contextual information than `http://mywfs.org?[...]&typeName=layer1&featureId=001`:

the latter yields a GML feature disconnected of its context, while the first tells the client the given feature is part of a larger set.

Linked data granularity in GIS services

A recent OGC Best Practice document [2] describes the three levels of granularity that are subject to semantic links in OGC services, namely the capabilities (describing the service itself), the datasets (exposed by the service) and the data entities (that constitute each dataset).

Ideally, in a linked data environment agnostic of OGC services, it is desirable to be able to refer to these three levels by URL, i.e.

- `http://mywfs.org` being the root of the service should also resolve to its capabilities
- `http://mywfs.org/featureTypeId` should resolve to a feature type description
- `http://mywfs.org/featureTypeId/fid` should resolve to a feature

Besides, to achieve broader acceptance by mainstream tools, these URL patterns should return specific mime types, so as to have a client that relies only on URL resolution and mime type negotiation to process resources.

Short of such URL patterns, fragments provide a way to achieve linked data with these three levels of granularity. They do not solve the mime type issue though, and links to such entities must explicitly mention the expected mime type, as shown in the example below.

Data Mosaics representation

Using the conventions proposed above, a data mosaic representation format was defined. For instance, this data mosaic document is a mashup of video, tabular data, geo data and HTML excerpts:

```
{"name": "Mosaic example",
 "content" : [
   {"name": "Some youtube video",
    "src": "http://youtu.be/kc8BAR7SHJI",
    "annotations": [
       {"src":     "#t=40,42",                                // temporal fragment
        "target": "http://example.org/placemarks.kml#id=..."},     // link within a KML file
       {"src":     "#t=02:23,02:25",
        "target": "http://example.org/data.csv#col=2-3&row=0-100"}// link to a CSV subset
    ]},
   {"name": "Some WMS layer",
    "src": "http://example.com/wms#id=layer1",
    "mimetype": "wms",
    "annotations": [
       {"src":"#bbox=2.5,49.5,6.4,51.5",                      // geospatial fragment
        "target": "http://example.org/doc.html#xpath=//DIV[2]"},  // link to HTML excerpt
       {"src": "#bbox=3,52,5,53",
        "target": "http://example.com/wfs#id=ft1/feature.001",     // link to WFS feature
        "targetMimetype": "wfs" }
    ]},
 ]}
```

(The example above deliberately omits several features like tabular data styling, semantic tagging and annotation content to focus on the geospatial example)

Relation with other models

While the encoding above is a simple, proprietary JSON-based encoding, the data mosaic model is compatible with the OpenAnnotations model [3], and an encoding based on OpenAnnotations is under development.

The data mosaic model also overlaps the OGC Context model [4], both in target use cases and technological challenges, but differ in that data mosaics lack the styling ability (and advanced OGC protocol configuration), while the OGC contexts are not meant to model linked data.

Perspectives

Beyond the usage in data mosaics, geo fragments offer a way to express finer-grained links within geospatial datasets. Furthermore, in the absence of directly resolvable URL patterns, they provide an alternative to refer to entities that constitute OGC services. Geo fragments therefore help making the bridge between existing, sometimes monolithic geospatial datasets and the overall web of data.

As for the data mosaics encoding, the current model lies at the intersection of both the W3C Open Annotations and the OGC context. Working towards an integration/fusion of these models looks promising, and would constitute an efficient mean to store and exchange collaborative contexts of work.

Lastly, allowing support of geospatial services by relying solely on URL resolution and mime type negotiation looks like a key to getting wider acceptance of these in a wider, non-geospatial linked data environment.

References

1. Duchesne P. (2013) Aggregating media fragments into collaborative mashups: standards and a prototype, http://www.w3.org/2013/04/odw/odw13_submission_28.pdf
2. Duchesne P., Houbie F., Maué P., Schade S. (2012) *Semantic annotations in OGC standards*. OGC Best Practice Document 08-167r2, https://portal.opengeospatial.org/files/?artifact_id=47857
3. Ciccarese P., Sanderson R., Van de Sompel H. (2013) *Open Annotation Data Model*. W3C Community Draft, http://www.openannotation.org/spec/core/
4. Brackin R., Gonçalves P. (2014) *OGC OWS Context Atom Encoding Standard*. OGC Standard, https://portal.opengeospatial.org/files/?artifact_id=55183