OAM Catalog Tech Challenge Proposal

Release 1.0

Micho Garcia, Oscar Fonts

March 15, 2015

Contents

1 Cover Letter

Dear OpenAerialMap team,

We are Micho Garcia and Oscar Fonts, two freelance Open Source Geospatial developers and active dynamizers in the the Spanish Open Source Geospatial communities. We are charter members in the OSGeo Spanish chapter, started our local "Geoinquietos" (GeoGeeks) groups in our respective places (Galicia and Barcelona), and usually sponsor and organize spanish FOSS4G events (GeoCamp unconferences, annual SIG Libre conference).

We believe in freedom and open collaboration not only for code or data, but in our professional lives. There are ethical, but also practical reasons for this. That's why we chose to be "free as in freelance", but at the same time created the "geomati.co" professional network, so we can learn from each other and team up for non trivial developments. Besides from being stimulating, working in a team leads to better quality outcomes, especially regarding design and documentation. Whenever we are allowed to, our code is public.

We work mainly as integrators of different components to build custom geospatial solutions, generally web map viewers or data visualization portals. Despite not being core commiters, we have had the opportunity to dive into many projects, and contributed modest patches to most of them. For example, in the Java front, we have contributed some lines of code to GeoTools, GeoServer, GeoNetwork, Thredds, gvSIG, 52n SOS server, or OpenTripPlanner.

But we lately tend to use Python to build our custom APIs (Django, Flask, web2py), while the "web application" tends to be 100% Javascript in the browser, or Android, so it is independent from the server technology. Of course it helps a lot to have standard web services and client implementations.

Experience told us that developing software is hard, but mantaining it is even harder. So with te time we tend to think much more on lowering the barrier to future change: flexibility and understandability, and have in mind the future mantainers (could be ourselves!).

When we came across this challenge call, we felt like it fitted us like a glove: two part-time Open Source Geospatial experienced developers for an Open Data initiative used in Humanitarian response... Wow! We just hope that our technical skills are appealing enough for you, and, either way, we are happy to see OAM project back alive!

2 Micho García

2.1 Bio

Geospatial Developer with 9 years of experience but following GIS scene from the last century.

Worked for multiple GIS companies and Administrations in a variety of GIS projects with privative and open source GIS tools.

In 2006 attended at Free GIS Conference in Girona and met the spanish open source crew. From then, leads life to work only with open source projects.

It is not until 2011 when established as an Open Source Geospatial freelance developer.

OSGeo Spanish Chapter charter member since 2009, OSGeo Spanish Board Member between 2010 - 2012 and OSGeo Charter Member since 2013.

Co-organized the 2013 and 2014 Geocamp.es editions, a GIS unconference. geocamp.es

Was a Scientific Committee member in the 6th, 7th and 8th gvSIG conferences gvSIG Conferences

Co-founded the Galician GeoGeeks group Xeoinquedos, collaborating to spread the "geoinquietos" movement, the only open source organization with real capacity for world domination. geoinquietos.org

OpenStreetMap dynamizer:

- Talking about OSM with Xeoinquedos
- Mapping Party of Galpon Free Saturdays
- Organizer of the Night of the living maps in Vigo

Co-founded the geomati.co GIS Freelance Network geomati.co

Is professor in UNIGIS Professional Master in Geospatial Technologies, in the "Sharing Geographical Information" module, where works with OpenStreetMap data.

2.2 Code contributions to Open Source Geospatial projects

- Geotalleres: Contributed base GeoServer documentation. Geotalleres is a creative commons collectively built documentation project for Spanish training on Geospatial FOSS. GeoWorkshops
- XOLViewer: vitamins for OpenLayers, a OpenLayers based framework to develop web map viewers quickly and modularly. Xolviewer `[jQuery]`
- Other minor patches / pull requests: Ol3-layerswitcher, UNIDATA Thredds, gvSIG CE

2.3 Own developments

- CIIFEN Geoportal `[Thredds, jQuery, Django, Bootstrap, PostGIS]`
- ikiMap Android `[Android, Java]`

2.4 Conference talks

- http://www.osgeopt.pt/sites/default/files/files/_geoinquietos-SASIG2011-2.pdf
- http://www.sigte.udg.edu/jornadassiglibre2014/uploads/2014/abstracts/r9.pdf
- http://www.sigte.udg.edu/jornadassiglibre2010/uploads/Articles/a13.pdf
- http://downloads.gvsig.org/download/events/gvSIG-Conference/7th-gvSIG-Conference/articles/Article-7j-Cliente_ikiMap.pdf
- http://dugi-doc.udg.edu/handle/10256/7664

3 Oscar Fonts

3.1 Bio

Telecommunications Engineer and Master in Intelligent Systems.

Senior Geospatial Developer with 10 years of experience.

In 2007 attended the first SIG Libre spanish conference and FOSS4G in Victoria, and changed his life.

Worked for the Catalan Official Mapping Agency (2006-2008), helped stabilize their raster WMS services and deployed the first Tile Cache services. Lead the development of their main web map viewer based on OpenLayers 2, and developed their official geocoding service.

OSGeo Spanish Chapter charter member since 2009.

Established as Freelance geospatial developer in 2010.

Co-organized the FOSS4G 2010 Barcelona conference.

Is a Scientific Committee member in the SIG Libre conference.

Co-founded the Geoinquiets Barcelona, the first local GeoGeeks group that quickly expanded to other spanish and latin american cities.

Co-founded the geomati.co GIS Freelance Network.

Has been professor at UNIGIS Professiona Master in Geospatial Technologies.

Has been a consultant at the United Nations REDD program, providing training and technical advice on the use of Open Source Geospatial technologies for Forest Monitoring System deployment in developing countries: http://www.nfms4redd.org/doc/

3.2 Code contributions to Open Source Geospatial projects

- GeoNetwork: Improved OpenSearch-geo interface. Ticket & patch: http://trac.osgeo.org/geonetwork/ticket/190
- GeoTools and GeoServer: Incorporated NTv2 Grid Shift and Custom (user-defined) coordinate transforms. The contribution in GeoServer's user manual: http://docs.geoserver.org/latest/en/user/advanced/crshandling/coordtransforms.html The GeoTools technical 'making of': http://geomati.co/icc_datumshift/geotools.html
- *CatalogConnector*: Added KML, Atom and GeoRSS response formats; added OpenSearch-geo interface. CatalogConnector launches distributed CSW queries to a collection of remote catalogs, and displays all the responses in an integrated view: real-time distributed search as opposed to massive copy-pasting (harvesting). Documentation on my contribution: http://www.geoportal-idec.cat/geoportal/catalogconnector/OpenSearchCatalogConnectorManual.pdf
- Sensor Widgets Visual components for SOS sensor data visualization. Own project.
- Other minor patches / pull requests: 52n SOS server, OpenTripPlanner, web2py, OpenLayers 2 sandbox

3.3 GitHub Repos

- https://github.com/oscarfonts?tab=repositories
- https://github.com/geomatico
- https://github.com/geodata

4 Common projects

Despite Micho and Oscar work as independent freelance developers, they have worked together in many occasions. Some common developments:

4.1 Integrated Carbon Observation System Data Portal

A dataportal to help climate change scientific research community. The general architecture is very similar to what we need for OAM Catalog:

1. A data repository: collection of NetCDF files, made available through Thredds Data Server.
2. An indexed catalog: customized GeoNetwork instance harvesting and indexing the former repository metadata.
3. A search & preview web interface: simple search form with text + bbox + timerange filters; paged results with in-browser data preview; "shopping cart" where to add the desired datasets for full download in original format.

Built on existing Open Source products (Thredds, GeoNetwork, Lucene, OpenLayers, ExtJS), we put the glue and fine-tunning.

Developed by Micho Garcia, Oscar Fonts and Fernando Gonzalez at geomati.co. Presented at various FOSS4G european conferences (Spain, Netherlands, UK):

- Article and Slides: http://dugi-doc.udg.edu/handle/10256/4196
- Demo video: https://vimeo.com/44806500
- GitHub repo: https://github.com/geomatico/dataportal
- Documentation (spanish): http://geomati.co/dataportal/

4.2 Geotalleres

Geotalleres is a compilation of reusable spanish language workshop materials for Open Source Geospatial training. It's a collective creative-commons knowledge base. Micho and Oscar contributed various initial PostGIS and GeoServer chapters.

- GitHub repo: https://github.com/geotalleres/geotalleres

4.3 SatAgro

A Satellite Information Service for farmers. Including:

1. Backend scripts to extract significative indicators for farmers (timeseries and time-dependant raster layers) out of free MODIS and Landsat imagery. Mainly via Python GDAL scripting and OpenForis.
2. Automatic raster data publishing in GeoServer through REST API, using *gsconfig* python client.
3. A browsable REST API using Django Rest Framework.
4. A web application structured with Backbone and RequireJS, and using Bootstrap, Leaflet and Highcharts.

- Demo site (dev version, not production ready): https://dev.satagro.pl

4.4 GeoPoster

Distributed map edition: markers and rich text descriptions. Wants to be like having a map on the wall where a group of people can stick pins and add notes effortlessly.

Techncally, an exercise to:

- Completely separate JS Client from REST API;
- Conceive the simplest possible UI with in-line rich text and map editing;
- Use Flask Python microframework.
- API: https://github.com/geomatico/geoposter/tree/master/geoposter-api
- UI: https://github.com/geomatico/geoposter/tree/master/geoposter-client

4.5 Mapa Literari Català

Not a techically complex application, but a very popular one. Links literature and territory, proposing different routes related to classical witers' texts.

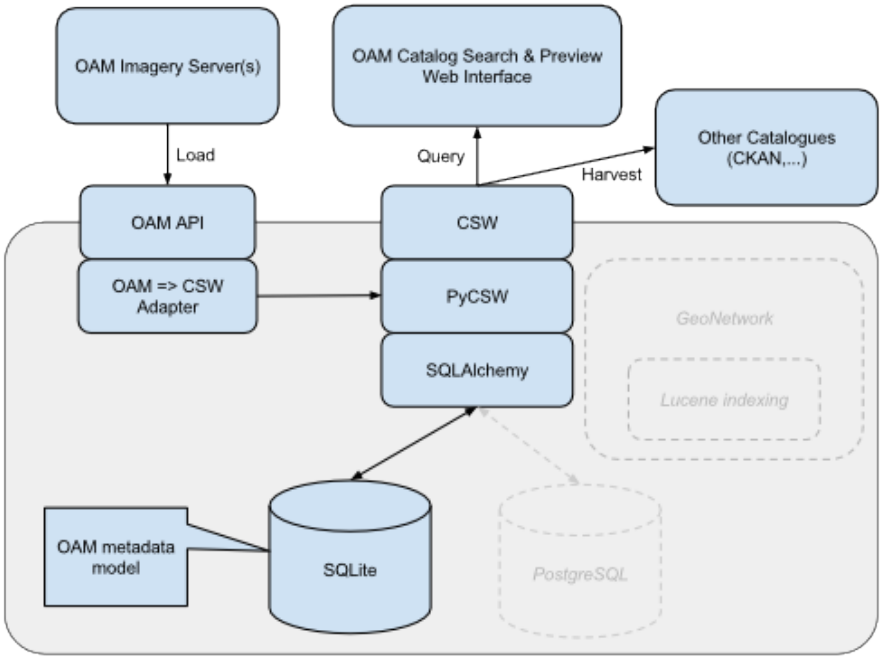
Oscar optimized the web application, separating the UI from the Data Acces, and replacing Google Maps with Leaflet. Django + PostgreSQL in the server, jQuery and Leaflet the browser. Micho developed the Android app and iOS app using Cordova for multiplatform development.

5 Technical Proposal

We have done something similar before, using GeoNetwork plus Lucene indexing tweaks to adapt it to specific search needs. It's a powerful technical setup, on the paper.

But this time we would like to start with something simpler and as lightweight as possible: PyCSW on top of SQLite. We ensure a low entry barrier for both deployers and future maintainers: Less hardware requirements, less code lines to maintain. We can then introduce more advanced components as real needs demand it. For example, the centralized online catalog may switch to PostGIS instead of SQLite if the catalog grows to hundreds of thousands of records, while the "take away" versions could keep the database-in-a-file SQLite benefits.

The web client will query the catalog through the standard CSW protocol. No custom query API. Using a standard protocol decouples server from client implementations, and ensures reusability and interchangeability on both sides. Not RESTful, but standard. We shouldn't be scared of CSW because of the monstruosity of the ISO metadata model, or some not-so-usable catalog instances out there. The protocol is just a standard HTTP API.



We would start by validating the OAM metadata model proposal, writing some reference metadata records based on real sample datasets. This will raise some questions and will let us refine the model, trying again to keep it as

simple as possible: gather only the relevant metadata for our use cases. Then we can configure PyCSW to use the OAM metadata model, and load the reference records.

The web client will have:

- Search by text
- Filter by BBOX
- Filter by time range
- Paged results
- Footprint preview
- Data preview (for WMS or TMS services)

The web client will use the Javascript libraries and tools that helped us keep the code tidy in our past projects: RequireJS for modular development, Bootstrap for UI goodness, Grunt (maybe Gulp) for task automation, Bower for dependency management, Jasmine for tests. We can adapt to other toolsets used by the HOT developer community, but will avoid using intrusive frameworks or immature tools that condition our future flexibility or sustainability just because they are the last trendy tool (Javascript is a wild world). For instance, developing an AngularJS directive would feel like digging our own vendor lock-in trap.

A less clear area (because we never did it) is how to wrap Catalog instances in a "take away" package that can be deployed on emergency scenarios. But there are some Windows "Portable Python" projects, and we can pack all the python dependencies in a virtual environment. So there is a good chance we can make an USB-stick distributable without a need for an installer or the overhead of a Virtual Machine. That would be our first bet.

OAM API development and integration with other OAM components (namely, the data server(s)) depends heavily on third parties and we don't have enough information to make a detailed proposal. We will use a well established Python web framework to build the api (preferably Django, could be Pyramid, would avoid weaker Flask).

We propose to create a specific "OAM-Catalog" public GitHub repo belonging to hotosm organization, using the Wiki pages for proposals, the issues for specific tasks, and these first-level directories:

- "docs": The OAM-Catalog technical documentation, based on Sphinx ReStructuredText (could be rendered out to readthedocs.org).
- "OAM-C-service": The python service, including PyCSW as a submdule.
- "OAM-C-viewer": Search & Preview web interface.

And of course a gh-pages branch to setup a landing page with links to source code, documentation, distributables, maybe some tutorial materials, and contact info.

Any potential improvement to the CSW service should be commited to (or pull-requested to) the upstream official PyCSW repo.

Regarding communication, we propose to have a weekly meeting with the OAM Project Manager to keep in sync, quickly iterate over the ongoing tasks and take decisions. Having direct occasional contact with real users and their needs would be also very helpful.