A NoSQL Geo-Data Solution for the Consumption of Services on the Web

P. Cau, S. Manca, R. Demontis, D. Muroni, C. Soru, L. Muscas, E. Lorrai, and P. A. Marras
Center for Advanced Studies, Research and Development in Sardinia (CRS4), Polaris, Pula
Email: plcau@crs4.it

Abstract—**Web applications and portals are strategic gateways to deliver tools, data, computational infrastructures and services over the Internet. Software and data interoperability is the key factor to enable the integration of knowledge and share common objectives. Web applications are using ever more big spatial data ecosystems that usually involve cross-border data flows and rely on open Internet. Demand of web GIS based applications, in particular, shows a steady growth over the last few years, indicative of a scenario where spatial-data infrastructures will be ever more consumed by mobile and web applications. Management and analysis of large and growing volumes of geo-data is challenging the scientific community without clear long term solutions. The INNO project's objective is to improve, develop and apply innovative, state of the art technologies to efficiently query, render and expose on the web spatially enabled data. The solution proposed is based on a NoSQL database infrastructure, on the set up of a efficient innovative communication protocol and the use of a light web-GIS client library to view results. Two specific goals are recognized to be of paramount importance; to improve the consumption of spatial data on the WEB and to build regional capacities on Global Earth Observation (GEO) proposing new standards and approaches.**

Index Terms—**No-SQL, Couchbase db, GIS, big data, GEOSS, WMS, WFS, web applications**

I. INTRODUCTION

GIS (Geographic Information System) based technologies over the last few years have seen a drastic increase in the production of GIS applications and projects in a vast verity of contests. Many traditional tools applications are being converted into client/server solutions exposed by mobile and web applications. This indicates a scenario where geo-data infrastructures will be consumed ever more in the future, with the massive use of web services [1], [2]. Reports on international market and research (e.g. Research and Markets - http://www.marketsandmarkets.com), highlights that GIS market is expected to grow at a rate higher than 10 % in the next years. Analysts state that such expansion is due to increased demand from all sectors (both private and public). On the same level, geo-data is steadily growing in dimension and spatial resolution. Due to the rapid growth in the volume of demand served from mobile devices and web-based tools, large numbers of geo-distributed data centres today are benefitting from modern cloud infrastructures.

A geo-database is a database optimized to store and query spatial tables (layers). Layers are objects defined in a geometric space and are usually referred to a geographic reference system. Currently, the WGS84 (World Geodetic System 1984) with revisions as recent as 2004, is the most used reference system worldwide. Geo-databases allow the representation of most geometric objects such as points, lines and polygons, 3D elements, *et al*. The experience made during many collaboration initiatives [3], such as the Global Earth Observation System of Systems (GEOSS), highlights that there is a dire need for increasing software and data interoperability for the sharing of information and knowledge between repositories from different sources.

So far, many open standards and web interoperability services are being considered such as the Web Map Service (WMS), the Web Feature Service (WFS), and Web Map Tile Service (WMTS) supported by the Open Geospatial Consortium (OGC) [4]. WMS is a standard protocol for serving geo-referenced map images over the internet that are generated by a rendering map engine using data from a GIS data source. WFS is an interface that allows to obtain geographical features across the web using platform-independent calls. While the WMS interface or online mapping portals like Google Maps return an image, the WFS interface provides the geometrical and alphanumeric data of the geographical layer, that end-users can edit or analyze. When exploiting the WFS, a web client obtains fine-grained information in GML (Geography Markup Language), a specialized XML format for geospatial data that describes both geometry and attribute. WMTS is a standard protocol for serving pre-rendered geo-referenced map subdivided in tiles over the Internet. This service aims at solving situations where short response times are necessary. WMS or WFS are not practical when dealing with massive parallel CPU-intensive use cases. As a matter of fact, to produce a image response, a WMS service can require some CPU seconds, depending on various factors. To overcome the CPU intensive, on-the-fly, rendering problem, pre-rendered map tiles can be used (e.g. google maps). On this regards, several schemes were created to manage these map tiles.

No existing web services solve all problems, and although their usefulness is widely recognized, the use of interoperability (e.g. OGC web services) standards is till limited.

II. OBJECTIVES AND METHODS

Scalability and flexibility of web applications, data accessibility and security are open issues tightly bound to technological development [5]. Such needs have been addressed by the INNO project by developing a suite of tools and in house solutions to exploit large geo-data sets, made available through a Storage Data Infrastructure (SDI). Such tools are loosely coupled components that aim at addressing specific needs such as data management and accessibility. A service oriented architecture has been optimised for deploying, storing, managing and querying GIS based data with a NoSQL approach. The solution we propose was developed and positively tested for the management of geo-data based on a varied version of the OGC WMTS implementation. Such solution is built on the use of vector tiles with different degree of resolution at different zoom levels. This ensures scalability and data can be replicated depending on available resources. Each tile is very light and is stored within the NoSQL db as a JSON document. Such documents store the information about the geometry of any geographical layer with different degree of resolution at different zoom levels (Fig. 1). This guarantees flexibility and fast response times by one side and, on the other side, it maintains the fine-grained information level of a WFS.
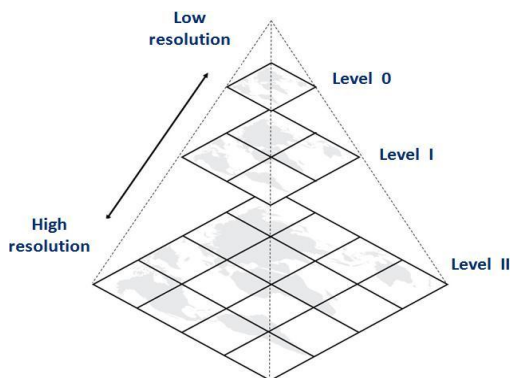

Figure 1.   Geo-data are processed to create vector tiles with different degree of resolution at different zoom levels.

JSON documents are used to represent all objects (map tiles - geometry and attributes) and their mutual relationships. The use of a document model enhances flexibility so that you can change application objects without having to migrate the database schema. Another advantage in a flexible, document-based data model is that it is well suited to represent real-world items and how you want to represent them. JSON documents support nested structures, as well as fields representing relationships between items which enable developers to realistically represent objects in the application.

The use of a NoSQL approach implies that appropriate algorithms are being implemented to allow an efficient access to the data. The communication protocol between back-end and front-end is designed so as to exchange the least amount of information possible: Only JSON documents (numeric vector map) of small size are transferred. These documents are then processed and rendered (e.g. transformed into images) by the front-end.

Connectors and interfaces have been developed for the transparent access to the data. Client applications can exploit API functions specialized to obtain the required data. These elements enable the user to remotely query the database that will respond with the necessary documents to the application requests.

III. BACK-END AND CLIENT

A suite of modules has been developed to manage the server side and the client side aspects. Creating the NoSQL database implies a pre-processing phase that, in our case, takes place on a postgreSQL/postGIS [6] environment. For each zoom level, vector tiles are created. In detail, a server side procedure processes the geographical layer to create a NoSQL instance. The geographical layer is subdivided into tiles with a varying precision at different zoom levels. Usually 18 zoom levels can be created for each geographical layer (the higher the zoom level, the greater the number of files to be created). At the 18th zoom level, a geographical area of 1 ° by 1 ° generates 1 million files. To limit the number of files for each zoom level, macro tiles can also be created. At lower zoom level, vector data are simplified in order to limit the transfer of data and the rendering done by the client. The simplification needs to be calibrated in order to create tiles with enough details. If it is too simplified on the client side, the image rendered is going to appear too coarse.

On the back-end an Extract Transport and Load (ETL) procedure has been developed that works as follow:

- Insert the layers (e.g. in shapefile format) into a PostgreSQL / PostGIS DBMS. A layer, in order to be loaded, must contain valid geometry elements (e.g. in Well Known text (WKT) format). This is required by the functions that transform the data from a PostGIS table into JSON documents (http://json.org/). The WKT has been chosen in the place of the GeoJSON because it is more compact.
- Process the data: the processing of the data takes place in the postGIS engine because NoSQL database engines are still not able to adequately manage and manipulate GIS data. The procedures create JSON documents for each zoom level and for each layer to be included within the NoSQL db.
- A semplification algorithm is applied to create the different zoom level,
- Deploy the JSON documents for all layers within the NoSQL db
- Create the indexes in the database.

A first version of a Client/Server Communication Protocol necessary for exploiting the geo-database infrastructure has been developed. It enables to query and access to the geographical layers and allows to retrieve

the vector tiles intended to be rendered and themed by the client.

In Fig. 2 and Fig. 3, we show the application of the simplification algorithm for the layer "Municipalities of Sardinia" at 10th zoom level. As shown in figures, this step requires a calibration phase in order to create documents that at each zoom level have enough information to represent realistically the real world.
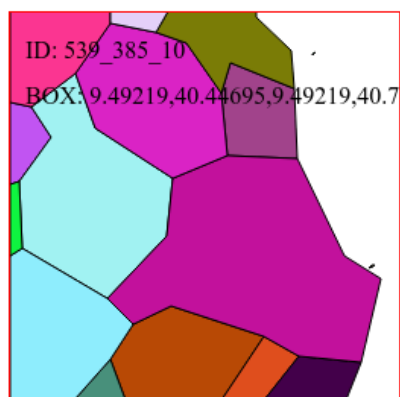

Figure 2. Example of the layer "Municipalities of Sardinia" rendered by the client. In this case the geometry is too simplified. The client uses the geometry contained in the Json documents at each zoom level required.
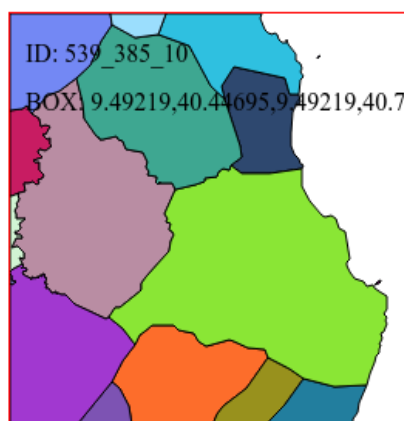

Figure 3. Example of the same layer "Municipalities of Sardinia", where the degree of simplification of the geometry has been calibrated. The image rendered represents more realistically the real world.

IV. THE TEST CASE

The use of a NoSQL database engine implies that the data structure needs to be designed particularly to meet requirements of the application. In this sense, a NoSQL database is an application oriented infrastructure, while a SQL database can be considered general and can serve many applications and purposes.

The inner logic within a NoSQL implementation is to be defined to satisfy a relationship type based on key-value. This approach does not provide explicit links between information.

A careful analysis of various NoSQL engines was carried out, paying particular attention on spatial extensions functionalities. MongoDB (www.mongodb.com), Couchbase

(www.couchbase.com/), and other NoSQL technologies were tested. Couchbase was chosen as it allows the creation of R-Tree type indexes. R-trees are data structures used for spatial access. Other NoSQL geospatial implementations are more primitive and offer only limited geospatial analysis capabilities. With the R-tree model, Couchbase has various functions to deal with spatial features (e.g. you can find polygons within a bounding box or a line that intersect a bounding box). Other NoSQL engines, such as Mongodb (one of the most popular NoSQL database software) can manage also spatial data, but do not provide sufficient spatial management capabilities and functions. Couchbase is a leading NoSQL distributed database, which supports key applications and is available as a software package of Enterprise level. The INNO tools were tested with various layers with different geometry provided by the following data-centers:

- The Sardinian Geo-portal (http://www.sardegnageoportale.it/): it provides high resolution geo-data of Sardinia (Italy). Data are exposed via web services (mostly WMS) and the infrastructures meets the requirements of the INSPIRE directives.
- Open Street Map (https://www.openstreetmap.org/about): it provides worldwide data such as buildings, railways, roads, waterways, land use, natural elements, locations points of interest. For each layer a descriptive table is also provided.

Tiles are generated using the definition "Slippy map Tilenames" (http://wiki.openstreetmap.org/wiki/Slippy_map_tilenames) although in our implementation no images are produced but JSON documents. Each zoom level is a directory. The zoom parameter is an integer between 0 (zoomed out) and 18 (zoomed in). 18 is normally the maximum, (some tile servers go beyond that). In the table below, it is shown for the layer "Municipalities of Sardinia", for each zoom level (from 4 to 18), the number of tiles, the maximum dimension of JSON documents, and the processing time required. As for the WFS, the rendering is also accomplished directly on the web-client. This operation is done each time a new layer is inserted into the NoSQL db. The Leaflet (http://leafletjs.com/) open-source JavaScript library has been chosen to visualize the data on the web-client.

This interface is particularly optimized for mobile, user-friendly interactive maps. It works efficiently across all major web and mobile platforms, making use of HTML5 and CSS3 on modern browsers while still being usable on older ones. In Fig. 2, we show a visualization of the communal borders of Sardinia stored in the NoSQL db.

In Table I, we show for the layer "Municipalities of Sardinia", for different zoom levels, the number of tiles, the maximum dimension of JSON documents, and the processing time required to process a geographical area within the Sardinian island (center of the Mediterranean Sea) of 1 °x 1 °.

TABLE I. For the Geographical Layer Named "Municipalities of Sardinia" and at Different Zoom Levels, We Show the Number of Tiles that Cover a Geagraphical Area of 1 °x 1 °, the Maximum Dimension of the JSON Documents, and the Processing Time Required.

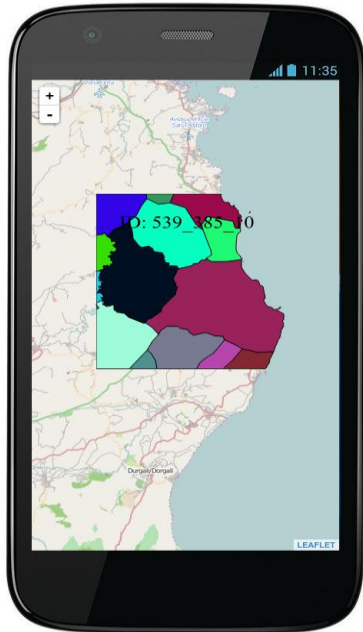| Zoom Level	N. Tiles (with macro tiles)	Tile Max Size (limit 30 Kb)	Processing Time
4	2	21442 byte	1 sec
6	2	33.272 byte	1 sec
8	6	23.297 byte	1 sec
10	44	5.154 byte	1 sec
14	5.318	785 byte	14 sec
16	28.962	1.536 byte	3 min 12 sec
17	61.757	934 byte	36 min
18	128.795	656 byte	108 min (1h,40m)

Figure 4. Visualization on a mobile client of a geographical layer stored within the NoSQL db.

V. Conclusions

Many enterprise organizations are integrating Geospatial Information Systems into various lines of business workflows. Public and private organizations are investing a great deal of work in GIS based infrastructures to build, maintain spatial data, and expose web services on the Internet. Web based technologies and services are key factors to many critical business decision systems. Scalability and flexibility of web applications are open issues and proper solutions need to be researched. Traditional OGC web services are not always well suited to serve the growing volume of geo-data.

This paper aims at improving existing interoperability methods for sharing data on the web. In this regards, the INNO project aims at improving scalability of big-geodata infrastructure and at providing customized spatial services and tools to enhance capabilities for geospatial data creation, queries, analysis, and data visualization. The experiments, we conducted, prove that our approach is valid and could be applied to many real situations.

References

[1] A. Sayar, M. Pierce, "Fox. Integrating AJAX Approach into GIS Visualization Web Services," *Advanced International Conference on Internet and Web Applications and Services* (AICT-ICIW '06).

[2] D. Gorgan, G. Giuliani, N. Ray, A. Lehmann, P. Cau, K. Abbaspour, K. Charvat, and A. Jonoski. (2013). "Black sea catchment observation system as a portal for GEOSS Community," *International Journal of Advanced Computer Science and Applications (IJACSA)*. [Online] Available: http://dx.doi.org/10.14569/SpecialIssue.2013.030301

[3] K. Charvat, P. Vohnout, M. Sredl, S. Kafka, T. Mildorf, A. De Bono, and G. Giuliani, "Enabling efficient discovery of and access to spatial data services," *International Journal of Advanced Computer Science and Applications(IJACSA)*, 2013.

[4] Open Geospatial Consortium, "OpenGIS Web Service Common Implementation Specification," pp.1-153, 2007.

[5] J. Pokorny, "NoSQL databases: a step to database scalability in web environment," *International Journal of Web Information Systems*, vol. 9, no. 1, pp.69 – 82.

[6] R. Obe, and L., Hsu, *PostGIS in Action. Manning Publications Co. Greenwich*, CT, USA, 2011. PP 520. ISBN 1935182269 9781935182269

Pierluigi Cau holds a full time position at Center for Advanced Studies and Research in Sardinia (CRS4). He has been working in the Environmental Sciences program of the Energy and Environment sector since 2000. His research topics are focused on computational Geographical Information Systems and development of innovative WEB ICT tools for the management of GIS data. He organized international workshops/conferences and taught advanced courses on hydrology and GIS at universities. He has tutored several stagers and early stage researchers.

Simone Manca has been with CRS4 since 2000. He works as an expert software engineer in the Distributed Computing Group. Currently he deals with software development for high-performance computing infrastructures, virtualization and distributed storage. In the past, he has worked in the biomedical field, developing interfaces and applications with health informatics standards such as DICOM and HL7. He is also experienced in the environmental and geographical information systems fields, having developed decision support system tools for planners, integrations of numerical models and web interfaces.

Costantino Soru is an Analyst Programmer. He works for the Environmental Sciences program of the Energy and Environment sector at CRS4. In 2009 he received a B.Sc. degree in Computer Science from the University of Bologna. He holds a strong experience on GNU/Linux development environment, C, C++ and Java programming, and PHP, Perl and Python scripting. His main scientific task is the development of data drivers for the handling of inputs and outputs produced by environmental models.

Davide Muroni holds a B.Sc. degree in Computer Technologies from University of Cagliari. Since June 2010, he works as a consultant for the Environmental Sciences group of the Energy and Environment sector at CRS4. He has experience in software development, relational and hierarchy database development, genetic algorithm and other approaches to optimization.

Laura Muscas is senior researcher at CRS4 where she has been working since 1992. She got a degree (cum laude) in civil engineering (structural section) at the University of Cagliari in 1990 and a PhD in geo-environmental engineering at the Politecnico of Torino in 1994. In the 1992 she attended the post-graduated courses: Hydrological Modelling of Flow and Pollution in Dry Regions at the Jacob Blaustein International Center (BIC) for Desert Studies of the Ben-Gurion University of the Negev. Her research interests are focused on the GIS (Geographic Information Systems) technology. She has experience on the development of GIS applications and DSS (Decision Support Systems) tools in the urban, industrial, tourist and environmental field; development of environmental applications based on GIS e GRID technologies; integration between GIS and environmental models (numerical and empirical). She held several short courses on GIS and has been tutor for several Stage, Master and Ph.D thesis.

Eva Lorrai is expert researcher at CRS4 where she has been working since 1996. She got a degree in civil engineering (transport section) at Cagliari University in 1995. Her research interests are focused on Geographic Information Systems (GIS) applied to urban, industrial, tourist and environmental planning, on the use of GIS as Decision Support Systems (DSS), on the integration between GIS and environmental models (numerical and empirical) and on the development of environmental applications based on GIS e GRID technologies. She held several short courses on GIS and has been tutor/co-advisor for degree and Ph.D thesis, Master and Post-Doc stage. She wrote about 30 scientific publications.

Roberto Demontis is expert technologist in the Bioengineering Research Program at CRS4 where he has been working since 2000. He got a degree in Computer Science at the University of Pisa in 1997. The scopes of research is on GIS (Geographical Information System) related technologies with particular reference to the treatment of environmental data and their correlation with data of other kinds. Uses and develops mainly open source software. He has experience on Big Data, DBMS and the development of WebGis applications and DSS (Decision Support Systems) tools.

Pier Andrea Marras holds a degree in Geological Science and Technology (University of Cagliari) and a Master in Geographic Information Systems for Regional Planning (University of Tor Vergata, Rome). Since September 2013 he has been involved in the research activities of the Environmental Sciences Program of the Energy and Environment Sector at CRS4. His work is focused on the development and application of environmental models, and Web and GIS technologies for water resources management. He is involved in the INNO project, dealing with the development of GIS based applications for the Web and mobile sector.