USING A FULLY OPEN SOURCE APPROACH TO WORKING WITH OPENSTREETMAP

BLAZEJ CIEPLUCH, RICKY JACOB, PETER MOONEY, PADRAIG CORCORAN, AND ADAM C. WINSTANLEY

Abstract. OpenStreetMap is a very exciting and vibrant project aiming to make accessing geographic data easier. Our research group at the Department of Computer Science NUI Maynooth Ireland is carrying out research into a broad range of topics including: map data generalisation, geographic shape complexity, web map services, map-based interface for pedestrian navigation. One of the common themes across this research is the use of OpenStreetMap as the principal source of geospatial data. In this paper we describe how our research productivity, research collaboration, and general data interoperability have been greatly enhanced from our early adoption of a fully open source GIS approach to working with OpenStreetMap. While one can work successfully with OpenStreetMap in non-Open Source environments the flexibility offered by an open source approach is a major advantage. This flexibility is delivered in many *flavours* including: a wide choice of software, interconnectability of software packages and components, a wide network of support through documentation, message boards, and free and open exchange of ideas.

1. Introduction

The advantages for organisations and individuals in using open source software are well publicised and documented. Anderson and Moreno-Sanchez (2003) find that using open source software has a number of advantages for organisations with scarce resources: no software costs, software tools are easily learned by personnel with general IT background, small software footprints, no need to commit to proprietary software, freedom to extend the software with functionality not present in commercial software and compatibility with existing IT infrastructure. OpenStreetMap (OSM) is a collaborative project to create a free editable map of the world. The maps are created using data from portable GPS devices, aerial photography, other free sources or simply from local knowledge. The data collected for the OpenStreetMap project is freely available for download in OpenStreetMap XML (OSM XML) data for all countries and continents directly from the OpenStreetMap servers. On a weekly basis the most current fully QA/QC version of OpenStreetmap is made available as an OSM XML file. This XML file is in **.osm** format and conforms to the OpenStreetMap XML schema.

The OpenStreetMap software tool chain provides a "healthy software ecosystem" for building the basemap and developing applications based on OpenStreetMap data (Ciepluch et al., 2009). The full chain stretches from the raw data editors (JOSM, Merkator), spatial database (PostGIS), renderers (Mapnik, Osmarender), tile servers (tilecache, Map Server), viewers (uDIG, QGIS, web browser, etc), applications (`bliin.com`, `cloudmade.com`), and routing applications (openrouteservice.org). Due to space constraints in the abstract we now provide summary details on two OpenStreetMap related projects currently ongoing in our research group. Each example attempts to highlight the advantages of using a fully open source GIS approach to our research. We begin with a description of a web-based application to explore an OpenStreetMap database. This is followed by the description of the application of map generalisation techniques to geographic features stored in OpenStreetMap databases.

1.1. **Development of an OpenStreetMap database explorer.** One of the most popular ways for those interested in OpenStreetMap to access the spatial data on their local machines is to load OpenStreetMap XML (OSM XML) into a PostGIS database. Once the spatial data is stored in PostGIS it can be used in any number of ways. However for users without significant OpenStreetMap experience exploring the database can be a little confusing. OpenStreetMap data can be explored online from `www.openstreetmap.org`. This interface requires users to have some knowledge of how OpenStreetMap data is organised before they can take maximum benefit from the interface. To address these issues we are developed an OpenStreetMap database explorer application. The application is written in PHP. OpenLayers is used as the Javascript map container. AJAX technology is used to ensure that query results are returned in response to user actions on the map without the need for page refresh or movement of the map from the current user context. The MAPNIK tool has been used to generate our own custom OpenStreetMap tiles. We have access to a collection of data from the Irish Environmental Protection Agency for research purposes. This data is mostly point and polygon based spatial data. Using a set of PHP scripts we take this data (from CSV files and Shapefiles) and insert into the local copy of OpenStreetMap in PostGIS on our server. We configure the map style file for the Mapnik tool to ensure that the newly generated set of map tiles for Ireland contains the newly added features. The user can also explore these newly added *local* features through the map explorer interface. The query fields on the map explorer interface are generated dynamically. If a new column is added to the PostGIS database for a new feature - for example chimney stack for a factory then this option is available automatically for query from the interface. In this way we can easily explore other OpenStreetMap databases in a seamless way without the need to change the interface when a new PostGIS database is loaded.

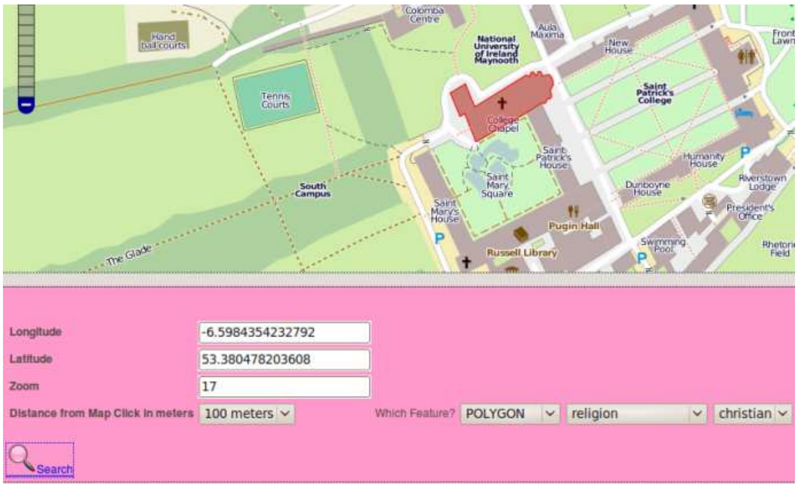

FIGURE 1. A screenshot of the OpenStreetMap database explorer application. A user has searched for places of worship within a 500 meter radius of their mouse click on the map

In Figure 1 we show a screenshot of one particular use case of the OpenStreetMap database explorer. The user has clicked on the map and chosen to run a query which will return all places of worship within a 500 meter radius of the mouse click. All polygons which match these criteria are returned and are highlighted on the map. The query results are displayed in a dynamically created HTML table underneath the map. From here the user can further explore the results for example: retrieve all tags associated with a specific polygon from the search results, links to view the polygon on the OpenStreetMap website explorer at `www.openstreetmap.org`, download KML version of the polygon(s) and other features to view in other ways such as Google Earth. Figure 2 shows another example this time with an administrative boundary (Kildare county) shown.

Unlike the traditional proprietary paradigm of software development, users of open source software have free access to the source code in open source environment which can be modified to fit with changing problem specifications and user needs. One excellent example of this occurs with the ***osm2pgsql*** tool. This tool takes raw OpenStreetMap XML data and automatically inserts this data into a PostGIS database. By default *osm2pgsql* imports a set number of spatial attributes for each geographic feature. This set is only a subset of the total number of spatial attributes which could potentially be imported. However one can easily edit some of the configuration files of *osm2pgsql* to alter the number of spatial attributes imported and to specify the import of additional attributes. This is particularly useful in the case of users how maintain their own local copy of OpenStreetMap and have either (1) integrated their own data into the PostGIS database with OpenStreetMap or (2) added additional localised spatial attributes.

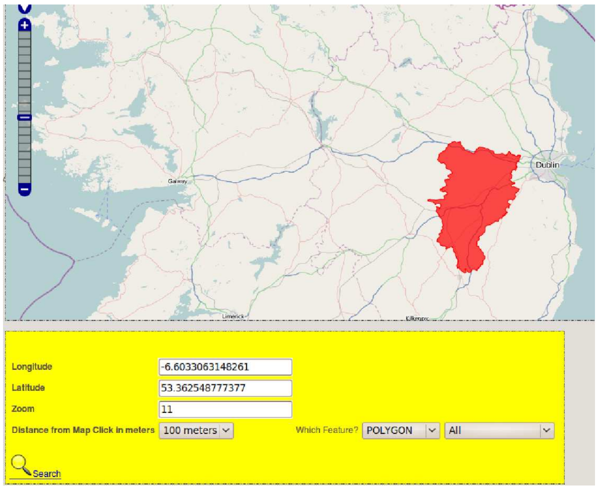

FIGURE 2. A screenshot of the OpenStreetMap database explorer application. A user has searched for all polygons and has choosen an administrative boundary - County Kildare

To summarise the following software products and tools have been utilised in this project:

- The PostGIS database
- The GDAL (Geospatial Data Abstraction Library) library allowing us to export data from PostGIS from a command line interface.
- PHP and the Apache Webserver
- phpMyAdmin and pgAdmin for backend administration of the PostGIS database
- Mapnik: for the generation of custom sets of web map tiles
- Tilecache: to provide a web-map service to serve out locally generated and stored map tiles.

1.2. **Map Generalisation.** OpenStreetMap contains spatial data on polygons for almost every geographic feature class. In some cases, depending on the characteristics of the real world object, these polygons are sampled at a very high resolution. It is not uncommon for polygons representing lakes or ponds less than $0.5km^2$ in area to have over 300 nodes or sample points. To transfer such polygons, in vector format, to a mobile device accessing a location-based service (LBS) this amount of data can possibility be: slow to download to the device, or slow to load and provide a visualisation to the user. Our group is investigating a map generalisation approach, by simplification, for OpenStreetMap data. Software has been developed in Java and C++ to investigate if a polygon, or group of polygons and other features such as roads, should be simplified. The shape characteristics of the polygons are analysed and the results of this are used to quantitatively decide if the set of features should be simplified. Simplification is recommended if the feature can have a significant number of nodes removed without altering the shape characteristics of the feature to a point where the simplified version is unrecognisable from the original feature. We now provide an example. In Figure 3 the polygon on the left is a large lake in Ireland. The original polygon is represented by 647 nodes. The simplified version of this polygon is shown on the right of Figure 3 and is represented by 39 nodes. For transfer of this polygon, in vector format, to mobile devices accessing location-based services this is a very significant reduction in data transfer size. The polygon is simplified to the extent that it is very suitable for viewing at scales where high resolution representations are not required. The simplification algorithms are developed such that all topological relationships are maintained. This simplification process and associated work shall be explained in greater detail in the full paper. The researchers involved in the development of both the Java and C++ code both cite the open availability of powerful code libraries as a major advantage. Rather than expending resources into writing ones own code for well known problems it makes more sense to reuse libraries such as: the standard template library, CGAL for computational geometry, PLPLOT for plotting in C++, and the Java Topology Suite (JTS) in Java. The focus shifts away from the tedious job of re-inventing the wheel to developing new research ideas. Steiniger and Bocher (2009) agree with this conclusion when they remark that "free and open source software (FOSS) preserves the researcher from 're-inventing the wheel"'. In a perfect resonance for our work here the authors make a specific example of map generalisation research by stating that "one can avoid re-implementing generalisation algorithms previously developed by others on other platforms" (Steiniger and Bocher, 2009).

2. Concluding Remarks

Open Source GIS software have proven to be able to provide robust capabilities for geospatial database creation, spatial analysis and geospatial web applications. The growing trend of the adoption of open source technologies for GIS is largely due to the fact that many successful OSS projects have proven to perform at acceptable and sometimes exceptional levels as compared to proprietary products (Serrão et al., 2003; Raghunathan et al., 2005). Our experience in working in GIS with a fully open source software approach has, so far, proven this correct. At this point in time we can confidently speculate that even as yet unknown future research needs and directions will not provide reason for our research group to stray outside the open source GIS paradigm. Steiniger (2009) strongly advocates a free and open source GIS approach because such an approach will help to mitigate against issues of "current proprietary software licenses and pricing and development models which limit access to broader community growth and implementation". One of the more tangible advantages of this approach

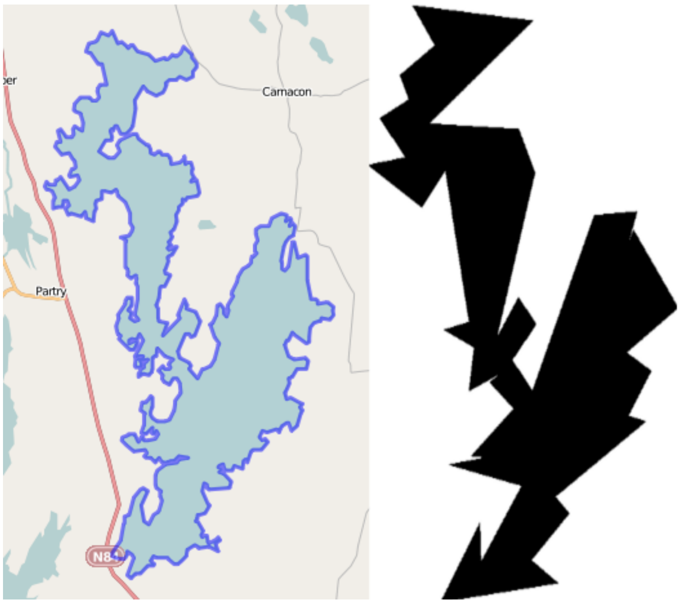

Figure 3. An example of the generalisation, by simplification, of a polygon from OpenStreetMap. The original polygon is on the left while the simplified polygon is on the right

from a research collaboration point of view is that researchers using proprietary computing environments have never found issues or problems with our use of an open source approach. Our software choices such as PostGIS, GDAL, uDIG (User Friendly Desktop and Internet GIS), etc means that we can consume spatial data and services from other computing environments easily. Alternatively we can exchange spatial data and services in an interoperable fashion meaning that software choice is encapsulated away from those outside our own research group. We close with remarks from Steiniger and Bocher (2009) who closes his paper with the statement that "while open source software may not be suited for everyone if the price tag argument is set aside. But it is the best choice for research in GIS".

References

Geoffrey Anderson and Rafael Moreno-Sanchez. Building web-based spatial information solutions around open specifications and open source software. *Transactions in GIS*, 7(4):447–466, 2003. URL `http://dx.doi.org/10.1111/1467-9671.00158`.

Blazej Ciepluch, Peter Mooney, Ricky Jacob, and Adam C. Winstanley. Using openstreetmap to deliver location-based environmental information in ireland. *SIGSPATIAL Special*, 1(3):17–22, 2009.

S. Raghunathan, A. Prasad, B. K. Mishra, and Hsihui Chang. Open source versus closed source: software quality in monopoly and competitive markets. *IEEE Transactions on Systems, Man, and Cybernetics, Part A*, 35(6):903–918, 2005. URL http://dx.doi.org/10.1109/TSMCA.2005.853493.

Carlos Serrão, Daniel Neves, and Paulo Trezentos. Open source security analysis - evaluating security of open source vs. closed source operating systems. In *International Conference on Enterprise Information Systems*, pages 433–440, 2003.

Stefan Steiniger. Free and open source geographic information tools for landscape ecology. *Ecological Informatics*, 4(4):183 – 195, 2009. ISSN 1574-9541.

Stefan Steiniger and Erwan Bocher. An overview on current free and open source desktop gis developments. *International Journal of Geographical Information Science*, 23(10):1345 – 1370, 2009.