Prospectus: Open Geocoding from OpenGeo

I. Summary

Geocoding is an essential function for many organizations and businesses. But currently there are no robust, enterprise-ready, open source geocoding solutions available. OpenGeo proposes to develop such a geocoding solution, and is looking for partners to provide funding. In return, partners would receive advance use of the software, and would be able to guide its development direction.

II. Goals

OpenGeo aims to provide an enterprise-ready, open source geocoding solution as part of the OpenGeo stack.

- **Enterprise ready.** The OpenGeo geocoding solution will have features and reliability worthy of its users' dedication. It will also be backed by quality support.

- **Open source.** Beyond its license, an open source geocoding solution requires community infrastructure and participation. These are necessary for the main benefits of open source software: frequent releases, responsive mailing lists, reliability, and flexibility. OpenGeo will develop the community for this geocoding solution, in addition to developing a code base.

Our plan for how OpenGeo can use its expertise to make open source geocoding a reality is described below.

III. Roadmap

Our approach to providing an enterprise-ready, open source geocoder is to work with an existing project to develop new features, integrate it into our supported OpenGeo stack, and provide community infrastructure and commitment.

In particular, we will be working with GeoCoder::US 2.0, originally developed for FortiusOne by veteran developer Schuyler Erle.

While GeoCoder::US performs much of the functionality needed from a geocoding service, there are a number of improvements needed for us to integrate it with the OpenGeo stack and provide enterprise-level support.

A. Technology

OpenGeo's open source, enterprise-ready geocoding solution requires the following developments of the existing GeoCoder::US technology.

1. Adapting to PostGIS

PostGIS adds support for geospatial features to PostgreSQL and is *the* open source spatial database of choice for enterprises.

Currently, Geocoder::US 2.0 works against SQLite, which is not spatially enabled. Adapting the geocoder to use PostGIS will help integrate it into the OpenGeo stack and

open up the possibility of future extensions that take advantage of PostGIS' spatial functions.

Work items for this adaptation include:

- Changing dependencies to use PostGIS contributed modules for edit distance and metaphone hashing.

- Modifications to existing SQL.

- Editing data import scripts to work with PostGIS.

- Expand database interaction code to account for PostGIS features.

- Splitting off the database backend portion of the application and adding database drivers for further database flexibility.

2. Returning multiple matches

Users sometimes want to be presented with a number of possible geolocation matches to an address string. But currently Geocoder::US 2.0 provides only the top match.

Providing the option of having multiple matches returned would allow for uses common to geolocation web services and provide a foundation for future work on crowd-sourced quality control.

3. Running on Java virtual machine

Geocoder::US is written largely in Ruby, with some C dependencies. In order to make it

easily deployable with the OpenGeo stack, it needs to be adjusted to run with JRuby—a Java implementation of the Ruby language--so that it uses the Java Virtual Machine.

4. Reverse geocoding

An advantage of the adaptation to PostGIS is the possibility of providing reverse geocoding functionality through the same geocoding software. To take full advantage of this opportunity, Geocoder::US needs additions to its API and changes to its data import process so that feature geometries are indexed.

5. Plot data

The current geocoding technology depends on address range data, with which the location of particular addresses is interpolated along a line (such as a particular street). Some address data sets, such as Ordnance Survey's Address Point product, provide a geographic coordinate for every individual address. For greatest flexibility, the geocoding technology must be extended to work with plot data.

B. Community

In addition to improvements to the geocoding technology, OpenGeo also plans to use its expertise in open source processes and connections among developers to encourage a lasting, nurturing community around the project.

1. Infrastructure

For an open source project to have a viable community, it requires a number of hosted services. OpenGeo will host and maintain:

- Mailing lists for users and developers.

- A public repository for project code

- An issue tracker for bug reports and feature requests.

- A public facing website, including a blog of project news and wiki for documentation.

2. Documentation

While many open source projects are poorly documented relative to their proprietary counterparts, OpenGeo's documentation specialists make sure that the projects in our stack are accessible to users.

For Geocoder::US, OpenGeo will invest in documentation, both for developers (JavaDocs and Ruby Docs) and for users.

3. Community building

Building an open source community takes time and effort beyond the items mentioned above. A development community grows when core developers take time to review and commit contributions from the periphery. A user community grows when developers take time to address the questions and requests that users make on mailing lists, issue trackers, etc. And the entire community grows when the project is successfully marketed, for example with a compelling website.

OpenGeo has years of experience growing these kinds of communities, and it would complement the deliverables listed above with a commitment to the community around the enterprise-ready geocoder.

IV. The Geocoding Future

The items listed above are sufficient for an open source, enterprise-ready geocoder integrated with the OpenGeo stack. They are only the beginning. Once the core geocoding technology and community is in place, OpenGeo plans to expand the project in a number of directions.

A. Additional data sets

Geocoder::US currently works only with the US Tiger/Line data set, a publically available data set published by the U.S. Census Bureau. Though this data set provides a good foundation for a U.S.-only geocoding service, there are more complete data sources available from commercial vendors. An OpenGeo-supported geocoder could be extended with new import scripts that would allow it to be used against these additional data sets

B. Internationalization

Geocoder::US's address parser currently assumes that addresses to be geocoded are in the address format used by most English-speaking countries. Proper internationalization of the geocoder will require, in addition to import scripts for additional data sets (see above), changes to the address parser to make it work with other address formats.

The current geocoding implementation also depends on a metaphone hash that is language-specific. Internationalization requires a metaphone hash for each new language, unless suffix tree indexing is implemented (see below)

C. Workflows for crowd-sourced data correction

One of the biggest obstacles for a truly open source geocoding system is the availability of open and accurate address data. For cases where the geocoding application is used with open data, OpenGeo intends to build an interface that would let users correct results manually as they geocode. When exposed as a web service, this user correction will facilitate crowd-sourced correction of address data, improving the quality of results.

D. Workflows for crowd-sourced weighting correction

When the geocoding technology provides users with multiple possible matches for a particular query this opens the possibility of letting users manually refine the relevance rating of each match. When exposed as a web service, this would facilitate the crowd-sourced correction of the match weighting function, improving the quality of results.

E. Refinement of U.S. address parser

Geocoder::US currently only implements a subset of the CASS U.S. address format standard. Complete support for this standard would improve the quality of results.

F. Support for points of interest

Geocoder::US currently only supports geocoding of mailing addresses. A complete geocoder would also support geocoding of points of interest that are not mailing addresses.

G. Suffix tree indexing

As an advanced technique that would dramatically improve performance and flexibility, our open source geocoding system should provide suffix tree indexing on its address data. This is a non-standard database feature that would provide myriad benefits, from speed to internationalization.

V. Pricing

OpenGeo seeks funding of $60,000.00 to build the technical and communal foundation of an enterprise-ready, open source geocoding service.

In addition to the services itemized below, this initial investment would allow OpenGeo to take on geocoding as one of its core projects and include it as part of The OpenGeo Suite. This means OpenGeo will continue in-kind community investment well beyond the scope of the foundational effort.

Please inquire with OpenGeo for quotes on the items described in Section IV. "The GeoCoding Future."

To get in touch for further information or a custom quote contact us at inquiry@opengeo.org