

Integrated Content and Geospatial Search: Open Source Software to Combine the Capabilities of Solr and PostGIS

“Blanco”

Abstract

There are a wide variety of use cases for information search functionality which can accept a combination of content and geospatial criteria. Open source software is available to address both of these individual requirements; [Solr](#) for content-based search and [PostGIS](#) for processing geospatial queries. Intellog Inc. proposes the development of a new open source software component (*“Blanco”*) which can take search requests with both types of criteria, dispatch them to the respective systems for processing, integrate the returned result sets and finally return the rationalized result set to the user or calling application.

Concurrent Proposal Submission

This proposal is being submitted simultaneously to the University of Calgary's [CPSC 594](#) (“594”) program and to the University of Helsinki's [Software Factory](#) (“SF”) program. It is intended for the proposal to be accepted by *both* programs concurrently. If successful, the combined student teams will have the additional challenge of coordinating development between the two cities separated by over 7000 kilometres and a nine hour time difference.



Sample Use Case

Many modern petroleum production companies maintain an electronic well file, which is a collection of electronic documents each of which is linked to a well master record. Each well master record contains geographic location information such as latitude, longitude and depth. The linkage between documents and the master record has the effect of geocoding each document in the well file -- the document has both content and geographic attributes which are the subject of searches.

For example, a user might ask; “*Find me all relevant information for horizontal multi-zone fracs within 10 kilometres of a town with 10,000 or more people in the province of Alberta.*” This query would be submitted to the new component, which in turn would dispatch the relevant criteria to Solr and PostGIS respectively. Solr would process a query for any document in its index containing the words *horizontal* and *multi-zone* and *frac* within 10 words of each other. PostGIS would process a geospatial query for every point representing a well which falls within 10 kilometres of a point representing a city, but only if that point’s population attribute has a value greater than 10,000, and where the point is located within a polygon which correlates to the province of Alberta. Once processing of the respective queries is complete, Solr and PostGIS would return their results to the new component, which in turn would match the document-hits from Solr and location-hits from PostGIS. Hits which cannot be matched between the two result sets are eliminated, and the remaining matched hits would then be returned to the user. The returned result set would meet all the content and geospatial criteria specified in the original query.

It is intended that the process described above will be entirely transparent to the calling application. Furthermore, it is intended to be sufficiently fast to support interactive use with a human user.

Educational Objectives

Students will have their learning experience enhanced in the following ways:

- *Whole Lifecycle Software Development* In addition to design, implementation and testing, it’s intended students will release the software to the general public. Students will have the opportunity to interact with real users attempting to use the software to solve real business problems. Students will be called to respond to the unexpected but inevitable issues arising from the release of the software to a user community they do not directly control.
- *‘Distributed Agile/SCRUM’* This project will provide experience with the Agile/SCRUM methodology where physically distributed team members will be working asynchronously. This reflects an emerging pattern for the development of open source software.
- *Open Source Development* Development tools for this project will be limited to those available with an open source license. Their use will enable the students to



assess the impacts, both positive and negative, of the open source approach as opposed to one based on commercial, proprietary development tools.

- *Start-Up Experience* This project will include many of the steps students will encounter should they decide to create their own start-up companies subsequent to graduation. Students will be provided with real-world experience with this potential career choice, and help them prepare for a future where traditional career development principles may not apply.

Development Environment and Technical Guidelines

Development tools will be limited to mainstream, open source products. The preferred implementation language is [Ruby on Rails](#). However, an alternative implementation language might be substituted if deemed a more suitable choice for technical or other reasons. PostGIS is an extension of the [PostgreSQL](#), so the latter is a logical choice for the project's relational database. Source code will be managed with [Git](#), using the [GitHub](#) code repository. Project management will be facilitated using the [Basecamp](#) online project management & collaboration tool. Team members will also make regular contributions to the project [blog](#), which is implemented with [WordPress](#). Current generation text editors ([TextMate](#) and [E Text Editor](#)) are preferred over full interactive development environments such as [Eclipse](#).

This project is intended to be *complementary* to the Solr and PostGIS projects. As such, no capability which can be found in either Solr or PostGIS will be replicated in the new component. For example, Solr has a growing list of geospatial capabilities. These will be investigated and determine if they can be employed to meet the project objectives.

If there are specific technical skills which students do not possess at the beginning of the project, they will be afforded a reasonable opportunity and resources to learn them. This is based on the assumption that students at an advanced level should be able to quickly adapt foundation skills to the specific tools chosen for the project.

Development Methodology and Guidelines

A modified version of Agile with SCRUM will be used. In keeping with this methodology, the student development team is expected to organize itself and the work to accomplish the project objectives. There are some additional high-level guidelines:

- *Sprints* The 2010-09 Sprint would be used to establish project scope, firm up the Product Backlog, finalize the specific development environment, specific development practices, and pursue any other 'Sprint Zero' activities deemed necessary. This will be followed by development Sprints in each calendar month 2010-10 through 2011-04, and each will deliver an increment of potentially shippable product functionality. Public beta testing may start as early as the conclusion of the first development Sprint.



- *Team Size* As with other open source projects, there is no preconceived notion as to the size of the team assigned to it. Within reason, the project should productively employ whatever number of students can be assigned to it. Accordingly, the Sprint Planning meeting conducted at the beginning of each Sprint will take stock of resources available for that particular Sprint and scale the goals of the Sprint accordingly.
- *Other Committers* As is normally the case with open source software, other developers would be invited to contribute to the development effort as [committers](#), but only *subsequent* to the conclusion of the development period discussed in this document.
- *When Is It Done?* Active open source projects continually evolve and are never really 'done' in a formal sense. The goal of this project is to make as much progress against the overall objectives as possible within the constraints of time and resources available. Hopefully, this will provide enough momentum that the project will continue after the 594 and SF programs have concluded.

Licensing and Non-Disclosure Agreements

Output from this project will be licensed under a mainstream open source software license with the specific choice as yet to be determined. As a result, students would be free to carry on the work beyond the term of the 594 or SF programs, should they choose to do so. The open source nature of this project obviates the need for non-disclosure agreements. **Note:** It is intended that the new component will stand alone from Solr and PostGIS, so as to avoid licensing entanglements with these projects.

Out of Scope

Areas of work which are considered to be out of scope:

- Content criteria and geospatial can be assumed to be pre-parsed. In other words, the lexical analysis required to parse out the respective types of criteria for a text-based query is considered out of scope.
- The population of Solr and PostGIS is also considered out of scope for the project. Students can assume both will be populated with suitable sample data to support the development effort.

Project History , Name and Additional Information

This project originally started with the objective of adding geospatial search capabilities to Intellog's [Onramp](#) search engine, but became dormant in the face of other business imperatives. While the project described in this proposal is an evolution of that original concept, it is sufficiently close to warrant continuing with the original name. The history of the project is documented on the project [blog](#).



Intellog uses the names of lighthouses found on the Oregon Coast, and this project is named after the magnificent [Cape Blanco](#) lighthouse. If you're in the neighborhood, it's well worth a visit.

For further information please [contact us](#), or email Terence Gannon at terryg@intellog.com

