The open discussion version of this paper is available at: Corti P, Lewis BG, Kralidis AT, Mwenda NJ. (2016) Implementing an open source spatio-temporal search platform for Spatial Data Infrastructures.
PeerJ Preprints 4:e2238v7 https://doi.org/10.7287/peerj.preprints.2238v7

Implementing an open source spatio-temporal search platform for Spatial Data Infrastructures

Paolo Corti[1], Benjamin Lewis[1], Athanasios Tom Kralidis[2], Ntabathia Jude Mwenda[1]

[1]Harvard University, Center for Geographic Analysis, Cambridge MA USA
[2]Open Source Geospatial Foundation, Beaverton OR USA

Corresponding author:
Paolo Corti[1]

Email address: pcorti@fas.harvard.edu

ABSTRACT

A Spatial Data Infrastructure (SDI) is a framework of geospatial data, metadata, users and tools intended to provide an efficient and flexible way to use spatial information. One of the key software components of an SDI is the catalogue service which is needed to discover, query, and manage the metadata. Catalogue services in an SDI are typically based on the Open Geospatial Consortium (OGC) Catalogue Service for the Web (CSW) standard which defines common interfaces for accessing the metadata information.

A search engine is a software system capable of supporting fast and reliable search, which may use "any means necessary" to get users to the resources they need quickly and efficiently. These techniques may include features such as full text search, natural language processing, weighted results, fuzzy tolerance results, faceting, hit highlighting, recommendations, feedback mechanisms based on log mining, usage statistic gathering, and many others. In this paper we will be focusing on improving geospatial search with a search engine platform that uses Lucene, a Java-based search library, at its core.

In work funded by the National Endowment for the Humanities, the Centre for Geographic Analysis (CGA) at Harvard University is in the process of re-engineering the search component of its public domain SDI (WorldMap http://worldmap.harvard.edu) which is based on the GeoNode platform. In the process the CGA has developed Harvard Hypermap (HHypermap), a map services registry and search platform independent from WorldMap.

The goal of HHypermap is to provide a framework for building and maintaining a comprehensive registry of web map services, and because such a registry is expected to be large, the system supports the development of clients with modern search capabilities such as spatial and temporal faceting and instant previews via an open API. Behind the scenes HHypermap scalably harvests OGC and Esri service metadata from distributed servers, organizes that information, and pushes it to a search engine. The system monitors services for reliability and uses that to improve search. End users will be able to search the SDI metadata using standard interfaces provided by the internal CSW catalogue, and will benefit from the enhanced search possibilities provided by an advanced search engine. HHypermap is built on an open source software stack.

Keywords: Catalogue Service for the Web, CSW, data discovery, geoportal, geospatial data,

Please cite this paper as: Paolo Corti, Benjamin G Lewis, Athanasios Tom Kralidis, Ntabathia Jude Mwenda (2018) Implementing an open source spatio-temporal search platform for Spatial Data Infrastructures. In Marchesini I. & Pierleoni A. (Eds.) Proceedings of the 4th Open Source Geospatial Research and Education Symposium (OGRS2016). Perugia, 12-14 October 2016.
https://doi.org/10.30437/ogrs2016_paper_37

metadata, search engine, Spatial Data Infrastructure, SDI, WorldMap

SPATIAL DATA INFRASTRUCTURE AND CATALOGUE SERVICE FOR THE WEB

SDI, Interoperability, and Standards

A Spatial Data Infrastructure (SDI) is a framework of geospatial data, metadata, users and tools which provides a mechanism for publishing and updating geospatial information. An SDI provides the architectural underpinnings for the discovery, evaluation, and use of geospatial information (Global Spatial Data Infrastructure Association, 2004). SDIs are typically distributed in nature and connected by disparate computing platforms and client/server design patterns.

A critical principle of an SDI is interoperability which can be defined as the ability of a system or components in a system to provide information sharing and inter-application cooperative process control through a mutual understanding of request and response mechanisms embodied in standards (Groot and McLaughlin, 2000).

Standards (formal, de facto, community) provide three primary benefits for geospatial information: a) portability: use and reuse of information and applications, b) interoperability: multiple system information exchange and c) maintainability: long term updating and effective use of a resource (Groot and McLaughlin, 2000).
The Open Geospatial Consortium (OGC) standards baseline has traditionally provided core standards definitions to major SDI activities. Along with other standards bodies (IETF, ISO, OASIS) and de facto / community efforts (Open Source Geospatial Foundation [OSGeo], etc.), OGC standards provide broadly accepted, mature specifications, profiles, and best practices.

Catalogue Service for the Web

Ease of data discovery is a critical measure of the effectiveness of an SDI. The OGC Catalogue interface standards (Catalogue Service for the Web [CSW]) specify the interfaces and bindings, as well as a framework for defining the application profiles required to publish and access digital catalogues of metadata for geospatial data and services. (Open Geospatial Consortium, 2016).

Based on the Dublin Core metadata information model, CSW supports broad interoperability around discovering geospatial data and services spatially, non-spatially, temporally, and via keywords or free text. CSW supports application profiles which allow for information communities to constrain and/or extend the CSW specification to satisfy specific discovery requirements and to realize tighter coupling and integration of geospatial data and services. The CSW ISO Application Profile is an example of a standard for geospatial data search which follows ISO geospatial metadata standards.

Limitations of CSW

While CSW provides numerous benefits to SDI's, there are numerous opportunities to enhance the functionality of CSW and the server implementations of CSW by adding in standard search engine functionality. Here are some examples:

- Facets: CSW does not easily emit facets or facet counts as part of search results. Facets can be based on numerous classification schemes, such as named geography, date and time extent, keywords, etc. and can be used to enable interactive feedback mechanisms which help users define and refine their searches effectively.
- Hit highlighting: CSW does not provide fragment highlighting of search results.
- JSON: CSW is historically rooted in eXtensible Markup Language (XML). Modern HTTP request/response mechanisms use JavaScript Object Notation (JSON) to provide compact representation of search results enabling better performance.
- Simplified query interface: the CSW query interface is based on XML, and quickly becomes complex when crafting advanced search requests (spatial, not spatial, temporal, etc.).

In addition, from a software engineering perspective, search engine implementations such as Lucene, natively provide numerous capabilities on which to build features like text stemming, relevance tuning, dictionaries and thesauri, while also supporting high scalability and replicability due to a shardable architecture.

THE NEED FOR SEARCH ENGINES IN SPATIAL DATA INFRASTRUCTURE

Search workflow and user experience is a vital part of modern web-based applications. Numerous types of web application such as Content Management Systems (CMS), Wikis, data delivery frameworks, all benefit from improved data discovery. In the last few years, these applications have delegated the task of search optimization to specific frameworks known as search engines. Rather than implementing a custom search logic, these platforms now often add a search engine in the stack to improve search. Apache Solr and Elasticsearch, two popular open source search engine web platforms, and both based on Apache Lucene, can now be part of a typical CMS stack to support complex search criteria, faceting, result highlighting, query spellcheck, relevance tuning and more (Smiley, D. and Pugh, E., 2009). As for CMS's, SDI search can dramatically benefit from such platforms as well.

Benefits from search engine frameworks

Typically the way a search engine works can be split into two distinct phases: indexing and searching. During the indexing phase, all of the documents (metadata, in the SDI context) that must be searched are scanned, and a list of search terms (an index) is built. For each search term, the index keeps track of the identifiers of the documents that contain the search term. During the searching phase only the index is looked at, and a list of the documents containing the given search term is quickly returned to the client. This indexed approach makes a search engine extremely fast in outputting results. On top of this, a search engine provides many other useful search related features, improving dramatically the experience of users.

Crucially, search engines are good at handling the ambiguities of natural languages, thanks to stop words (words filtered out during the processing of text), stemming (ability to detect words derived from a common root), synonyms detection, and controlled vocabularies such as thesauri and taxonomies. It is possible to do phrase searches and proximity searches (search for a phrase

containing two different words separated by a specified number of words). Results can be weighted, providing a way to rank results provided to users with the more relevant ones closer to the top. In addition, it is possible to use regular expressions, wildcard search, and fuzzy search to provide results for a given term and its common variations. It is also possible to support boolean queries: a user is able to search results using terms and boolean operators such as AND, OR, NOT and hit highlighting can provide immediate search term suggestions to the user searching a text string in metadata.

An important search engine feature useful for searching a metadata catalogue is faceted search. Faceting is the arrangement of search results in categories based on indexed terms. This capability makes it possible for example, to provide an immediate indication of the number of times that common keywords are contained in different metadata documents.

A typical use case is with metadata categories, keywords and regions. Thanks to facets, the user interface of an SDI catalogue can display counts for documents by category, keyword or region (Figure 1).

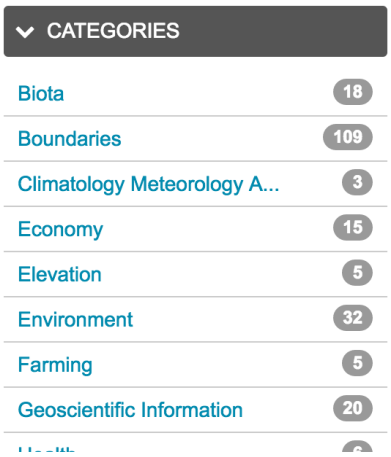

Figure 1. Facets generates counts for metadata categories

Search engines can also support temporal and spatial faceting, two features that are extremely useful for browsing large collections of geospatial metadata. Temporal faceting can display the number of metadata documents by date range as a kind of histogram. Spatial faceting can provide a spatial surface representing the distribution of layers or features across an area of interest. In Figure 2 a heatmap is generated by spatial faceting which shows the distribution of layers in the WorldMap SDI for a given geographic region.

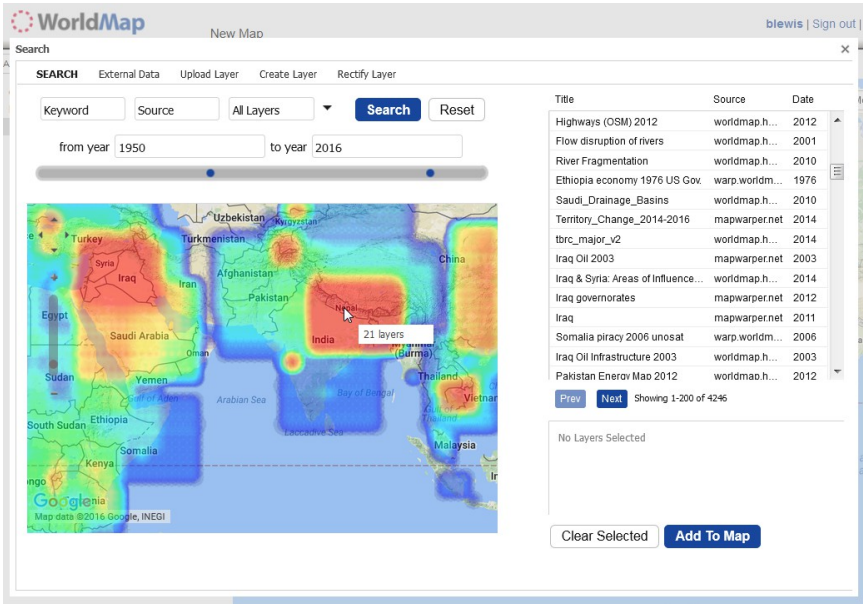

Figure 2. Spatial faceting enables heatmaps showing the distribution of layers in the space

Furthermore, from a software engineering perspective, search engines are highly scalable and replicable, thanks to their shardable architecture. Such systems are capable of providing interactive query access to collections of spatio-temporal objects containing billions of features.

HHYPERMAP: AN SDI SEARCH ENGINE BASED ON FREE AND OPEN SOURCE SOFTWARE

HHypermap is an application that manages OGC web services (such as WMS, WMTS), and Esri REST endpoints and in addition supports map service crawling, and harvesting, and uptime statistics gathering for services and layers. HHypermap is a metadata registry/catalogue/search application which works off remote and distributed data services. The aim of HHypermap is to provide a more effective search experience to WorldMap users and also for users outside WorldMap. WorldMap is an open source mapping platform developed by the CGA to lower the barrier for scholars who wish to explore, visualize, edit and publish geospatial information.

HHypermap is built exclusively on a free and open source software (FOSS) architecture (Figure 3), comprised most notably by the following components (in alphabetical order):

- Celery: asynchronous Python task queue utilizing the RabbitMQ message broker
- Django: web framework written in Python
- Lucene: search engine library made available using either of the Apache Solr or

Elasticsearch search platforms

- MapProxy: open source proxy for geospatial data. It caches, accelerates and transforms data from existing map services
- Memcached: high-performance, distributed memory object caching system, used to speed up dynamic web applications by alleviating database load
- OWSLib and arcrest: Python packages for working with OGC and Esri web services
- PostgreSQL: relational database management system (RDBMS)
- PostGIS: spatial extension for the PostreSQL database
- pycsw: CSW server implementation written in Python, providing CSW 2.0.2 and 3.0.0 support, as well as other standards-based search APIs

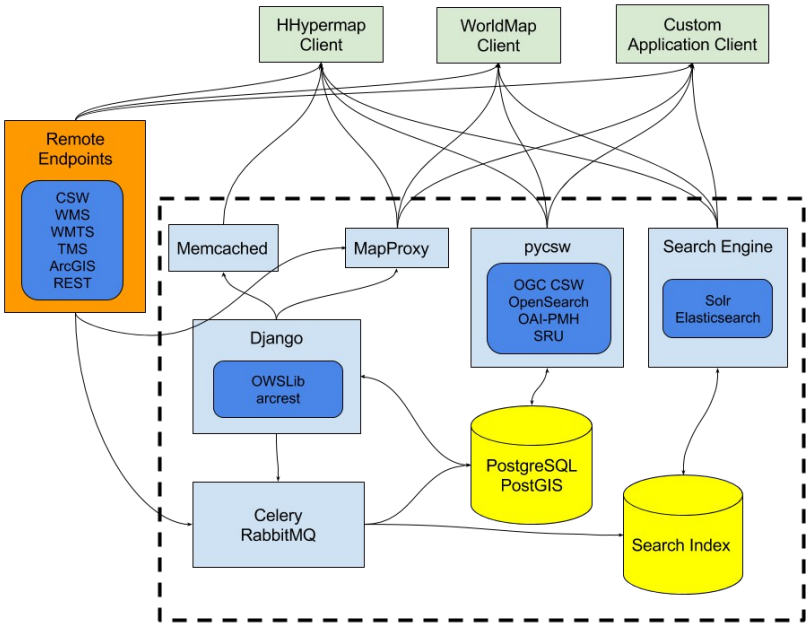

Figure 3. Hhypermap system architecture

HHypermap is developed and managed on GitHub; all source code can be found at https://github.com/cga-harvard/HHypermap, along with instructions on setup, configuration, deployment and use.

FUTURE WORK

While the CSW 3.0.0 standard provides improvements to address mass market search/discovery, the benefits of search engine implementations combined with broad interoperability of the CSW standard present opportunities to integrate the CSW standard with search engine methodologies. The authors hope that such an approach will become formalized as a CSW Application Profile or

Best Practice in order to achieve maximum benefit and adoption in SDI activities. This will allow CSW implementations to make better use of search engine methodologies for improving the user search experience in SDI workflows.

CONCLUSION

Harvard Hypermap aims to provide a FOSS solution using modern approaches to realize a highly scalable, flexible and robust geospatial registry and catalogue/search platform while achieving broad interoperability via open standards.

REFERENCES

Bone, C., Ager, A., Bunzel, K. and Tierney, L., 2016. A geospatial search engine for discovering multi-format geospatial data across the web. *International Journal of Digital Earth*, *9*(1), pp.47-62.

Chen, N., Chen, Z., Hu, C. and Di, L., 2011. A capability matching and ontology reasoning method for high precision OGC web service discovery. *International Journal of Digital Earth*, *4*(6), pp.449-470.

Infrastructures, D.S.D., 2004. the SDI Cookbook. *GSDI/Nebert*

Goodchild, M.F., Fu, P. and Rich, P., 2007. Sharing geographic information: an assessment of the Geospatial One-Stop. *Annals of the Association of American Geographers*, *97*(2), pp.250-266.
Groot, Richard, and John D. McLaughlin, eds. *Geospatial data infrastructure: concepts, cases, and good practice*. Oxford: Oxford university press, 2000.

Guan, W.W., Bol, P.K., Lewis, B.G., Bertrand, M., Berman, M.L. and Blossom, J.C., 2012. *WorldMap–a geospatial framework for collaborative research. Annals of GIS, 18(2)*, pp.121-134.

Kralidis, A.T., 2009. Geospatial web services: The evolution of geospatial data infrastructure. In *The Geospatial Web* (pp. 223-228). Springer London.

Li, W., Yang, C. and Yang, C., 2010. An active crawler for discovering geospatial web services and their distribution pattern–A case study of OGC Web Map Service. *International Journal of Geographical Information Science*,*24*(8), pp.1127-1147.

Masó, J., Pons, X. and Zabala, A., 2012. Tuning the second-generation SDI: theoretical aspects and real use cases. *International Journal of Geographical Information Science*, *26*(6), pp.983-1014.

Open Geospatial Consortium, 2016, Catalog Service | OGC, <http://www.opengeospatial.org/standards/cat>.

Rajabifard, A., Kalantari, M. and Binns, A., 2009. SDI and metadata entry and updating tools. *SDI convergence*, *121*.

Smiley, D. and Pugh, E., 2009. *Solr 1.4 Enterprise Search Server*. Packt Publishing Ltd.