Implementing an Open Source Spatiotemporal Search Platform for Spatial Data Infrastructures

OGRS 2016, Perugia, Italy - 10/13/2016

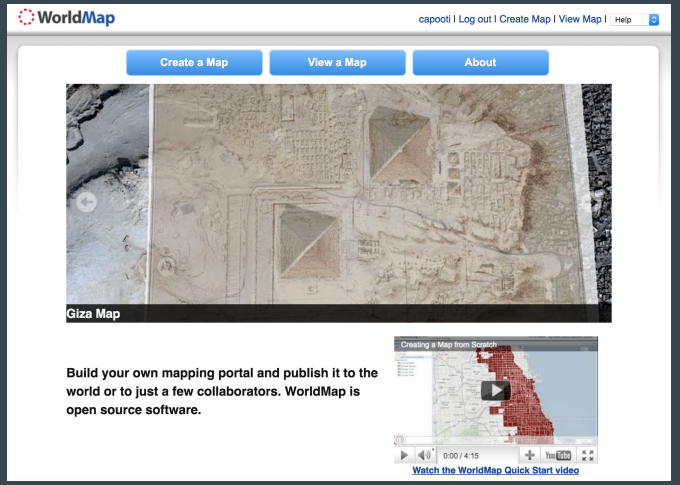
Paolo Corti [1], Benjamin Lewis [1], Athanasios Tom Kralidis [2], Ntabathia Jude Mwenda [1]

[1] Harvard Center for Geographic Analysis

[2] Open Source Geospatial Foundation

HHypermap (Harvard Hypermap)

- With Funding from the National Endowment for the Humanities, the Harvard Centre for Geographic Analysis (CGA) developed **HHypermap**, a map services registry and search platform
- HHypermap was developed in the process of re-engineering the search component of CGA's public domain SDI (**WorldMap** http://worldmap.harvard.edu), based on GeoNode
- It is built on an open source software stack



A brief history

WorldMap was developed on GeoNode 1.2 and released in 2012 as a public space for scholars and the public to upload and share spatial data.

Within a year WorldMap had 12,000 datasets and 8000 users.

Finding data became difficult and demand grew for being able to bring in and save map service layers from other servers.

The CGA proposed to NEH to build a system for building and maintaining a comprehensive registry of WMS and Esri REST Map services that would plug into Worldmap.

Being able to search for data by time and space as well as by keyword was a priority given the user base.

Note on uptake

Since the release of HHypermap on GitHub in April, a **U.S. federal agency** has adopted it and **Boundless** is using it within its flagship platform **Boundless Exchange**.

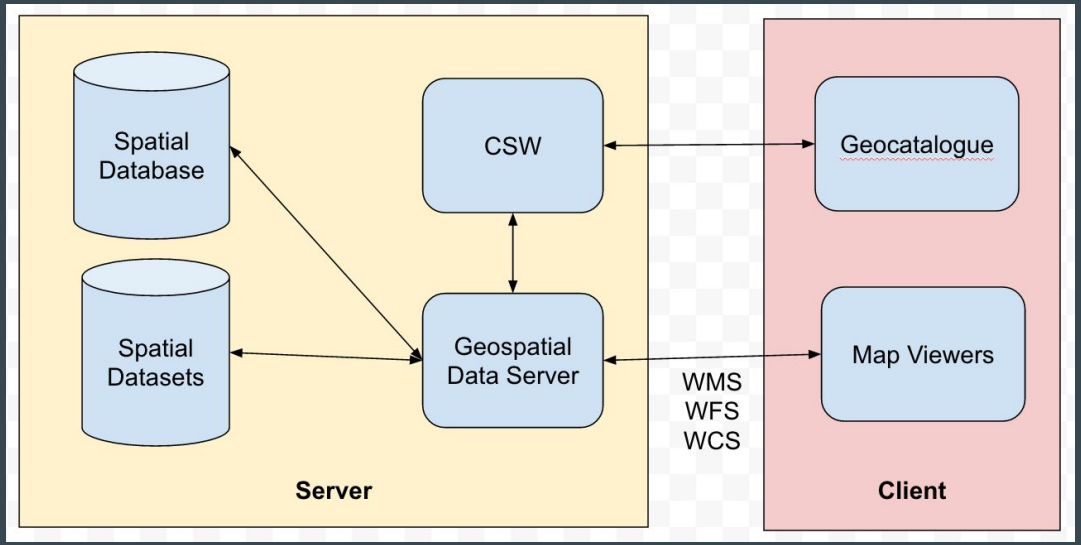
The geo-visualization capabilities we developed (2D faceting in Lucene) are highly scalable and being used to build a "**Billion Object Platform**" to enable interactive exploration of a billion spatio-temporal objects.

Glossary

- Spatial Data Infrastructure (SDI)
- Catalogue Service for the Web (CSW)
- Search Engine

Spatial Data Infrastructure (SDI)

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



Catalogue Service for the Web (CSW)

- 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
- Notable implementations: pycsw, GeoNetwork

Search Engine

- A **search engine** is a software system capable of supporting fast and reliable search
- It provides features such as full text search, natural language processing, weighted results, fuzzy tolerance results, faceting, hit highlighting
- Highly scalable and replicable architecture
- Notable implementations: Solr and ElasticSearch, both based on Apache Lucene

Goals of HHypermap

- Provide a framework for building and maintaining a comprehensive registry of web map services
- Support 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
- Monitor services and layers for reliability and use to improve results ranking
- End users can search the SDI metadata using standard interfaces provided by the internal CSW catalogue, and benefit from advanced search capabilities provided by a more full featured, RESTful API

Catalogue Service for the Web

- The OGC Catalogue Service for the Web (CSW) standard 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
- 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 freetext
- 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

Limitations of CSW

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

- Faceted search
- JSON representation (vs XML in CSW)
- Simplified query interface (CSW, being based on XML, can quickly become complex)
- Text stemming (ability to detect words derived from a common root)
- Highly scalable and replicable architecture

The Need for Search Engines in Spatial Data Infrastructure

- Numerous types of web application such as 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
- As for CMS, SDI search can dramatically benefit if paired with these platforms

How a search engine works

Two distinct phases:

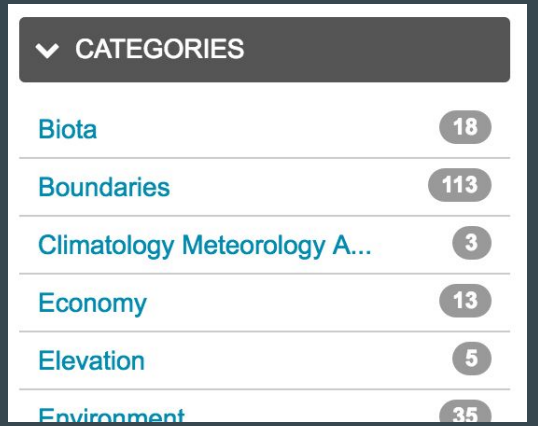
- **Indexing:** all of the documents (metadata, in the SDI context) that must be searched are scanned, and a list of search terms (an index) is built. For each search term, the index keeps track only of the identifiers of the documents that contain the search term
- **Searching:** only the index is looked 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

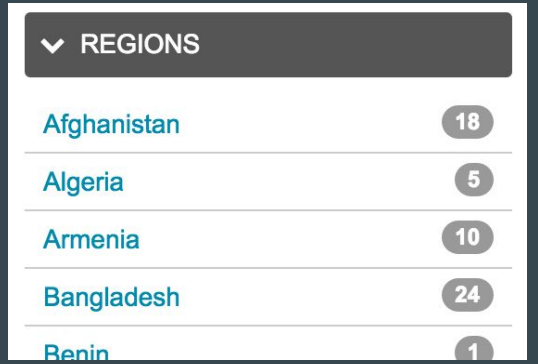
Benefits from search engine frameworks

- Very fast, thanks to the indexing mechanism
- Handling the ambiguities of natural languages, with 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
- Phrase searches and proximity searches (search for a phrase containing two different words separated by a specified number of words)
- Weighted results
- Handling regular expressions, wildcard search, and fuzzy search to provide results for a given term and its common variations
- Support for boolean queries (AND, OR, NOT)
- Hit highlighting
- Highly scalable and replicable

Faceted search

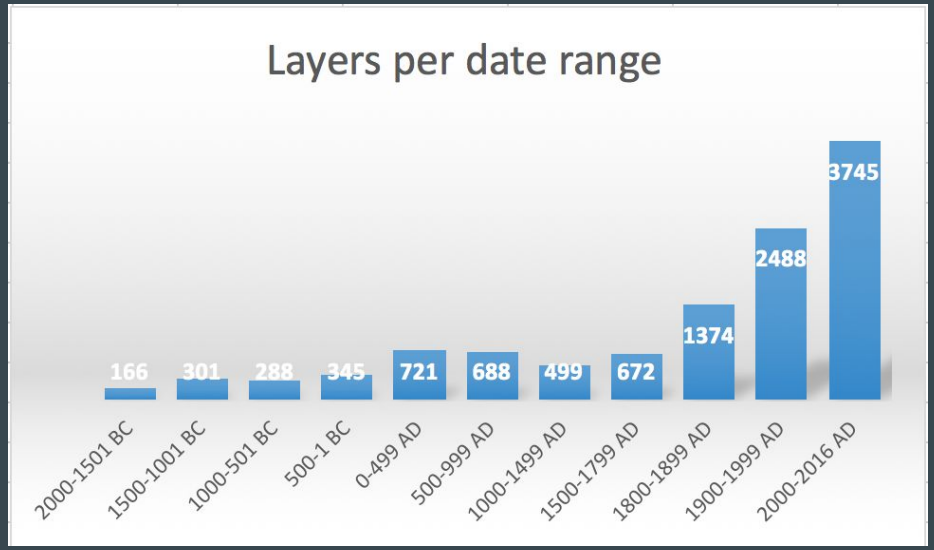
- Faceting is the arrangement of search results in categories based on indexed terms
- This capability makes it possible for example, to provide an immediate indication of the number of times that common keywords are contained in different metadata documents
- A typical use case for SDI is with metadata categories, keywords and regions
- Faceting without a search engine is generally computationally expensive in relational normalized structures (lots of query in a RDBMS)





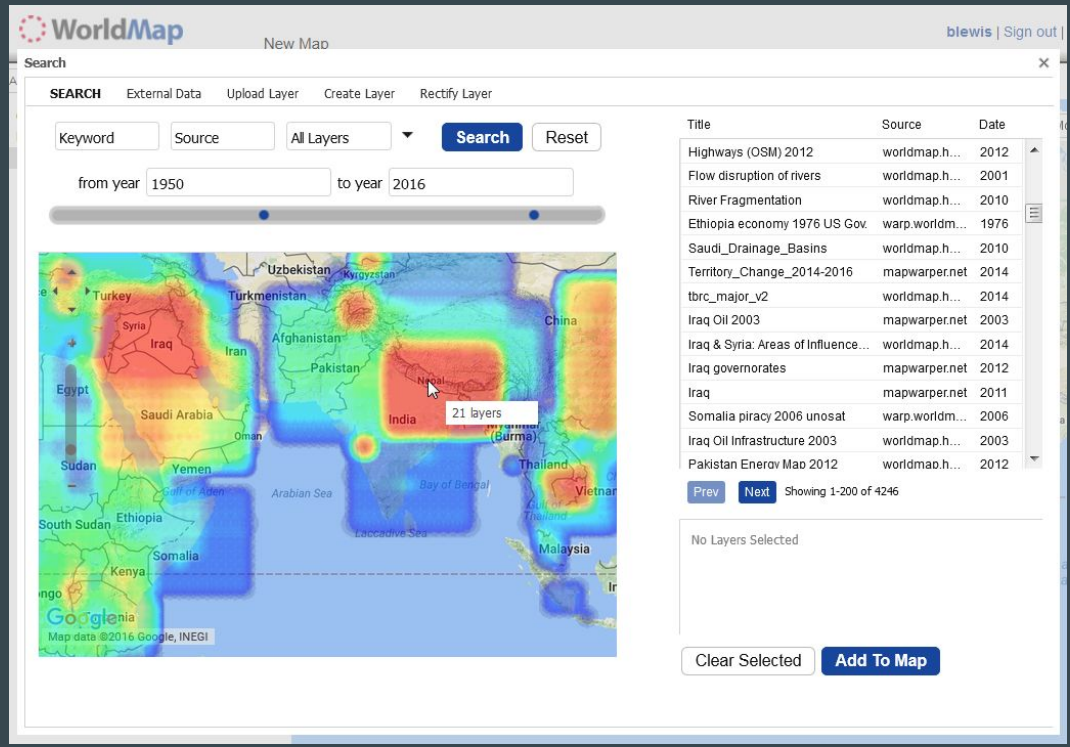
Temporal faceting

- 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 in a SDI by date range as a kind of histogram



Spatial faceting

- Spatial faceting can provide a spatial surface representing the distribution of layers or features across an area of interest
- In this example a heatmap is generated by spatial faceting to show the distribution of layers in the WorldMap SDI for a given geographic region



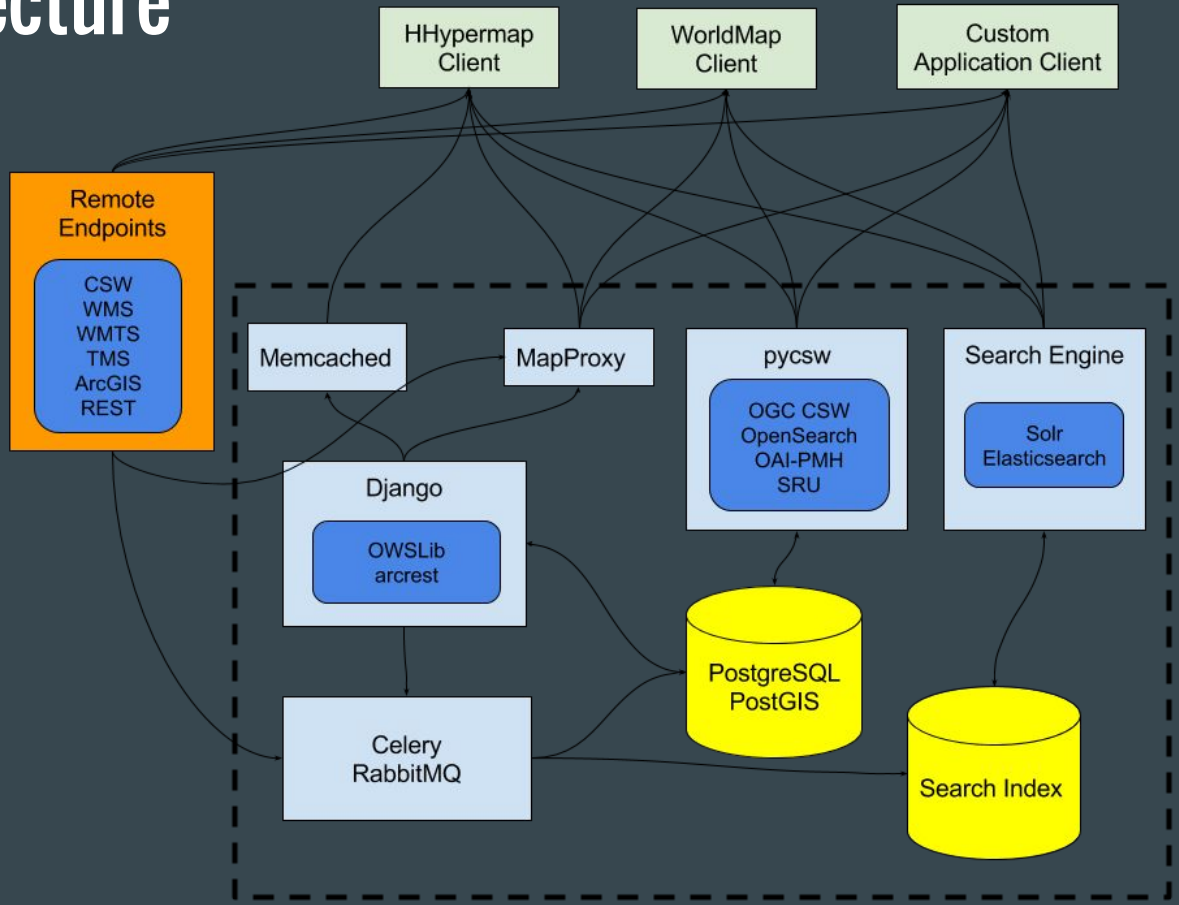
HHypermap: an SDI search engine based on Free and Open Source Software

- HHypermap is an application that manages OGC web services (such as WMS, WMTS), and Esri REST endpoints
- map service crawling, and harvesting, and uptime statistics gathering for remote services and layers
- 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, based on GeoNode, developed by the CGA to lower the barrier for scholars who wish to explore, visualize, edit and publish geospatial information

HHypermap Architecture

Built on Open Source software:

Celery, RabbitMQ, Django, Lucene (Solr, Elasticsearch), MapProxy, Memcached, OWSLib, PostgreSQL, PostGIS, pycsw



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

HHypermap 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

- Harvard University CGA: http://gis.harvard.edu/
- WorldMap: http://worldmap.harvard.edu/
- Harvard Hypermap public registry: http://hh.worldmap.harvard.edu/
- HHypermap code repository: https://github.com/cga-harvard/HHypermap