SERVEI DE SISTEMES D'INFORMACIÓ GEOGRÀFICA I TELEDETECCIÓ
Universitat de Girona

VI JORNADAS DE SIG LIBRE

Construyendo un sistema de indexación y búsqueda de recursos georreferenciados

A. Beltran Fonollosa, L. Díaz Sánchez y J. Huerta Guijarro

Institute of New Imaging Technologies (INIT). Universidad Jaume I de Castellón, Avda. de Vicente Sos Baynat s/n, E-12071 Castellón, {arturo.beltran, diazl, huerta}@uji.es

RESUMEN

La cantidad y variedad de recursos georreferenciados disponibles en la web crece día a día. Este hecho demuestra el interés de los usuarios y el papel fundamental que la información con contexto geográfico juega en la sociedad. Actualmente existen numerosos servicios Web especializados en tipos concretos de recursos como imágenes, video o texto que nos permiten realizar búsquedas en base a una localización. Por otra parte, de manera más formal, en el contexto de los Sistemas de Información Geográfica (SIG) se han realizado grandes esfuerzos en generar grandes catálogos de metadatos. Sin embargo, debido entre otros, a que la generación y publicación de metadatos es un proceso manual, existe una escasez de ellos y aún resulta complicado encontrar contenidos georreferenciados relevantes de una forma integrada y sencilla.

Podemos tomar como referencia el mundo web, donde inicialmente era muy complicado encontrar los contenidos que eran relevantes. La revolución llego con los buscadores, empresas como Yahoo! o Google se dieron cuenta de las deficiencias del sistema y empezaron a recopilar ellos mismos información de cada recurso cuyos creadores dejaban accesible. Esta labor es realizada por los conocidos robots o crawlers, que se dedican a recorrer sistemáticamente los recursos disponibles con el fin de obtener de ellos el máximo de información (metadatos) que su tecnología les permite. En base a estos metadatos se podrán indexar los recursos de una forma más exacta y eficiente proporcionando resultados más relevantes y exactos a las búsquedas realizadas por los usuarios. Tal ha sido el éxito de estos buscadores que hoy es inimaginable la búsqueda de información y la navegación por la red sin acceder a alguno de estos servicios.

Por ello, en este trabajo se presenta una primera aproximación para desarrollar un sistema de indexación y búsqueda de recursos georreferenciados. Mediante este sistema se pretende mejorar el descubrimiento y consecuentemente la accesibilidad a la información, nuestro objetivo inicial.

Palabras clave: *Descubrimiento, Indexación, Metadatos, Recursos Georreferenciados, Crawler.*

1.- INTRODUCCIÓN

La cantidad y variedad de recursos georreferenciados disponibles en la web crece día a día. Este hecho demuestra el interés de los usuarios y el papel fundamental que la información con contexto geográfico juega en la sociedad. Actualmente existen numerosos servicios Web especializados en tipos concretos de recursos como imágenes, video o texto que nos permiten realizar búsquedas en base a una localización. Por otra parte, de manera más formal, en el contexto de los Sistemas de Información Geográfica (SIG) se han realizado grandes esfuerzos en generar grandes catálogos de metadatos. Sin embargo, debido entre otros, a que la generación y publicación de metadatos es un proceso manual, existe una escasez de ellos y aún resulta complicado encontrar contenidos georreferenciados relevantes de una forma integrada y sencilla.

En el trabajo presentado en las anteriores Jornadas de SIG Libre [1], se puede comprender de forma fácil que en este escenario heterogéneo las descripciones de los recursos resultan ser la pieza clave de cualquier sistema de información [2]. Mediante los metadatos se pretende describir los recursos en base a sus propiedades, características y el contexto en el que el recurso toma sentido. Gracias a la indexación y la catalogación de los recursos de acuerdo a sus características (tipo de datos, contenido, origen, calidad, fecha de creación, localización, etc.) y su contexto, se posibilita y facilita su posterior descubrimiento [3].

La generación y/o creación de metadatos ha sido identificada como una tarea tediosa y poco gratificante, siendo necesario dedicar gran cantidad de tiempo y recursos, tanto económicos como humanos [4][5]. Además de resultar una tarea pesada, la compilación manual de metadatos, supone una fuente de errores por parte del creador [6]. Resulta esencial impulsar la investigación en todos los campos relacionados con la generación y gestión de metadatos dado el papel fundamental que estos juegan en cualquier sistema de información. Estos metadatos permitirían indexar o catalogar los recursos de una forma más exacta en los sistemas de información y, en consecuencia, tendremos la capacidad de proporcionar resultados más relevantes y exactos a las búsquedas realizadas por los usuarios. Con todo esto, conseguiremos mejorar el descubrimiento y consecuentemente la accesibilidad a la información, nuestro objetivo inicial.

Podemos tomar como referencia el mundo web, donde inicialmente era muy complicado encontrar los contenidos que eran relevantes. En consecuencia, aparecieron los primeros directorios, similares a nuestros actuales catálogos, donde los creadores de contenidos podían clasificar sus recursos de acuerdo a ciertos criterios como su temática, es decir, creando e introduciendo manualmente sus metadatos. Esto facilitó la accesibilidad a la información al usuario, pues este podía encontrar más fácilmente recursos acordes a su interés. Pero este sistema no resultaba efectivo, el trabajo de dar de alta la página en un gran número de directorios no era gratificante para los creadores de contenidos. Además el sistema era susceptible de engaño al clasificar los contenidos de acuerdo a los deseos de sus creadores.

La revolución llego con los buscadores, empresas como Yahoo![1] o Google[2] se dieron cuenta de las deficiencias del sistema y empezaron a recopilar ellos mismos información de cada recurso que sus creadores dejaban accesible. Esta labor es realizada por los conocidos robots o *crawlers*, que se dedican a recorrer sistemáticamente los recursos disponibles con el fin de obtener de ellos el máximo de información (metadatos) que su tecnología les permite. En base a estos metadatos se podrán indexar los recursos de una forma más exacta y eficiente proporcionando resultados más relevantes y exactos a las búsquedas realizadas por los usuarios. Tal

[1] http://www.yahoo.com
[2] http://www.google.com

ha sido el éxito de estos buscadores que hoy es inimaginable la búsqueda de información y la navegación por la red sin acceder a alguno de estos servicios.

Comprobada la validez de la metodología *"Google"*, se pueden imaginar los beneficios que aportaría al contexto de la información geográfica. Por lo tanto, se ha trabajando en la línea de evolución que ha seguido el mundo web, consiguiendo que los metadatos, a pesar de su papel fundamental, sean totalmente transparentes al usuario.

Con el fin de poner en práctica los conceptos mencionados, se planteó la creación de un sistema de generación y publicación masiva de metadatos, denominado *GeoCrawler*. Se pretende que este sistema, de forma sinérgica, trate distintos tipos de recursos, diferentes métodos de generación de metadatos y diferentes estrategias de publicación. Con la idea de posibilitar y facilitar el descubrimiento de los recursos, el objetivo del sistema es recopilar, describir, indexar o catalogar y finalmente publicar todos los recursos que se consideren de interés disponibles en el contexto en el que se ejecuta.

El resto del artículo se estructura en cinco secciones: En la sección 2 se introducen brevemente los antecedentes de este proyecto y se justifica la elección de diferentes herramientas relacionadas, en la sección 3 se explican los requisitos necesarios para alcanzar los objetivos y se presenta la arquitectura básica del sistema de indexación y búsqueda, en la sección 4 se dan algunos detalles de implementación que nos permiten entender cómo encajan las diferentes piezas del sistema, en la sección 5 se describen las bondades del sistema obtenido como resultado y, finalmente, en la sección 6 se recogen las conclusiones.

2.- TRABAJO RELACIONADO

2.1.- Generación de metadatos

Intentando abordar el problema de la creación de metadatos en [7], tras un extenso estado del arte, se plantea un camino común para la generación automática de metadatos. El trabajo empieza con la extracción de metadatos y, posteriormente, en base a la información obtenida se pueden aplicar diferentes métodos de generación de metadatos (cálculo, deducción, inferencia, etc.) para mejorar la descripción del recurso. Con esa metodología se pretende poder describir de forma completa y veraz los recursos para posteriormente poder ser publicados y conseguir así facilitar a los usuarios el acceso a los mismos.

Las tareas de extracción de metadatos permiten obtener gran cantidad de información del recurso en sí mismo, de su contexto y, obviamente, de su contenido. Pero esto no siempre es fácil, dado que la extracción automática de metadatos implica el conocimiento de las estructuras internas de los formatos de almacenamiento de datos utilizados por los recursos geográficos [5]. Este proceso normalmente lleva a cabo una correspondencia entre las características extraídas de cada formato y los distintos elementos de metadatos descritos por alguno o algunos de los estándares existentes (Dublin Core, ISO19115, etc.). Sin embargo, el gran número de formatos de datos existentes para los recursos geográficos hace muy difícil que una sola aplicación pueda manejar todos ellos. Un enfoque alternativo es el desarrollo de soluciones integradas y flexibles basadas en la reutilización de librerías, herramientas o componentes que son capaces de leer múltiples formatos para extraer la información de metadatos.

Si aparte de describir los recursos geográficos, se pretende generalizar el proceso de extracción de metadatos para cualquier tipo de recurso georreferenciados, entonces el problema se agrava, dado que el número de posibles formatos con los que se va a tener que tratar aumenta considerablemente. Por esa razón, se trabajó

en una plataforma que permite acceder a recursos heterogéneos y obtener información de una manera homogénea.

La necesidad de acceder y de obtener información de tantos formatos como sea posible motivó un estudio que analiza y evalúa varias plataformas comunes que proporcionan acceso a información geográfica, así como varias soluciones de código abierto para la extracción de metadatos [1].

Con objetivo de obtener descripciones de recursos basadas en la extracción de metadatos, se evaluaron varias herramientas de extracción de metadatos. Dado que se pretende generalizar el proceso de extracción de metadatos para cualquier tipo de recurso, se consideró la herramienta Apache Tika [8] como una solución que se adecua a los requerimientos, pese a no incluir soporte para formatos geoespaciales.

Como paso previo a la extracción de los metadatos, la nueva solución tenía que ser capaz de acceder e interpretar los formatos de datos geográficos por lo que se analizaron diversas plataformas de acceso a datos. El proyecto OSGeo FDO [9] posibilita el acceso a diversas fuentes de datos geoespaciales a través de un mecanismo común. Además, soporta una gran variedad de fuentes de datos, incluyendo formatos de archivos, bases de datos y servicios geoespaciales. Por lo que se consideró que ofrece la funcionalidad deseada.

En resumen, el enfoque que se adoptó en [1] para acceder, interpretar y extraer los metadatos de los recursos geográficos y no geográficos combina los beneficios de dos herramientas independientes pero complementarias: Apache Tika y OSGeo FDO. El resultado fue un prototipo funcional que permite la extracción de metadatos de una amplia gama de formatos de recursos multimedia. En concreto, esta herramienta soporta los tipos de recursos inicialmente soportados por OSGeo FDO (más de 150 formatos de IG)[3] y los tipos de recursos que soporta Apache Tika (más de 50 formatos multimedia)[4]. Por lo que la solución integrada es compatible con más de 200 formatos de recursos multimedia en su conjunto.

2.2.- *Crawlers* y Herramientas de indexación y búsqueda

Como se ha argumentado en la sección de introducción, se pretende seguir trabajando en la línea de evolución que ha seguido el mundo web, consiguiendo que los metadatos, a pesar de su papel fundamental, sean totalmente transparentes al usuario. Y, a su vez, permitan recopilar, describir, indexar y publicar de una forma eficiente los recursos disponibles. Facilitando al usuario, en el mayor grado posible, la accesibilidad a ellos.

Combinando ciertas herramientas se podría conseguir un sistema de indexación y búsqueda de recursos geoespaciales que facilite al usuario el descubrimiento y la accesibilidad a esos recursos. Por este motivo, se han buscado soluciones que ofrezcan un motor de indexación y búsqueda sobre las descripciones de los recursos que se obtendrán mediante la solución que integra Apache Tika y OSGeo FDO. Y, por otra parte, se han probado diferentes implementaciones de *crawlers* que permitan recopilar los recursos.

En lo referente al motor de indexación y búsqueda, se ha realizado un análisis de distintos tipos de indexación de registros de metadatos, así como la integración de estos índices dentro de un sistema de recuperación de información. Concretamente, se analizan diferentes alternativas para la indexación espacial de metadatos y también para su indexación textual.

De forma muy resumida, como estrategias de indexación espacial, se estudian los índices *Quadtree* y los índices *R-tree*. Quedando estos últimos como vencedores por su tiempo de respuesta más rápido, su soporte para consultas de tipo *"contains"* y por

[3] http://fdo.osgeo.org/OSProviderOverviews.html
[4] http://tika.apache.org/0.8/formats.html

su indexación de geometrías complejas con su consiguiente reducción de falsos positivos.

Respecto a la indexación textual, se investiga la creación de este tipo de índices para acelerar las búsquedas y la devolución de metadatos de información geográfica. Con este objetivo se estudian los índices *XQueryIndex*, *MG4JIndex*, *Apache Lucene* y la indexación en memoria (*indexerMemory*). Concluyendo que la opción que mejor se comporta en todas las situaciones es la del proyecto Apache Lucene[5], ya que los tiempos de respuesta a consultas son siempre similares, independientemente de la complejidad de las mismas y el número de metadatos indexados.

Dado que en este caso se quieren indexar todos los metadatos que se puedan generar, se elegirá como base la indexación textual. Por lo que, queda justificada la elección inicial de soluciones basadas en el motor de búsqueda e indexación del proyecto Apache Lucene.

Por otra parte, dada la naturaleza espacial de los datos que se manejan, se considera interesante el desarrollo de una plataforma que permita combinar ambos tipos de índices (espaciales y textuales), de forma que los resultados de las búsquedas se vean beneficiados por las bondades de ambos tipos de índices. En este sentido, en base al algoritmo utilizado por la aplicación de catálogo GeoNetwork[6], se ha trabajado en un algoritmo de resolución de consultas espaciales que se puede incorporar al motor de indexación Apache Lucene, permitiendo de este modo la resolución conjunta de operaciones espaciales y textuales.

2.2.1.- Apache Solr

Teniendo en cuenta los requerimientos descritos anteriormente, la solución que más se ajusta a las necesidades es el proyecto de la comunidad Apache Solr[7] basado en el motor de búsqueda e indexación Lucene.

Solr es la plataforma de búsqueda de código abierto del proyecto Apache Lucene. Sus características principales incluyen potentes búsquedas sobre texto completo, marcado de coincidencias, búsqueda por facetas, *clustering* dinámico, integración de bases de datos, manejo de documentos avanzados (por ejemplo, Word, PDF) y búsqueda geoespacial. Además, Solr es altamente escalable, proporcionando búsquedas distribuidas y replicación de índices. Por ello, proporciona las funciones de navegación y búsqueda de muchos de los sitios más grandes del mundo de Internet.

Solr está escrito en Java y se ejecuta como servidor de búsqueda independiente dentro de un contenedor de *servlets* como Tomcat[8]. Solr utiliza la librería de Lucene como núcleo para la indexación y búsqueda de texto completo, y ofrece interfaces REST[9] como HTTP/XML y JSON que hacen que sea fácil de usar desde virtualmente cualquier lenguaje de programación. La potente configuración externa de Solr permite que pueda adaptarse a casi cualquier tipo de aplicación, además cuenta con una arquitectura de *plugins* que nos permite extender su funcionalidad cuando se requiere una personalización más avanzada.

2.2.2.- Apache Nutch

Respecto al *crawler*, teniendo en cuenta los requerimientos descritos anteriormente y tras analizar varias herramientas de tipo *crawler*, incluyendo una de desarrollo propio para explorar el contenido de una máquina local. Se considera que

[5] http://lucene.apache.org
[6] http://geonetwork-opensource.org
[7] http://lucene.apache.org/solr
[8] http://tomcat.apache.org
[9] http://es.wikipedia.org/wiki/Representational_State_Transfer

la solución que más se ajunta a las necesidades es el proyecto Nutch[10] de la comunidad Apache.

Nutch es una implementación de código abierto en Java de un *crawler*, proporcionando todas las herramientas que necesita para ejecutarlo. Entre sus bondades pueden destacarse la transparencia y el entendimiento del sistema, dado que al ser de código abierto cualquiera puede ver cómo funcionan sus algoritmos. Otro de sus puntos fuertes es la extensibilidad, Nutch es muy flexible y permite que los desarrolladores puedan añadir funciones de filtrado de recursos, de indexación o de procesamiento para nuevos tipos de recursos.

El *crawler* obtiene los recursos y construye un índice invertido, que posteriormente un motor de búsqueda puede utilizar para responder las consultas de los usuarios. La principal ventaja de Nutch es que está basado en lo referente a la generación de índices está basado en las librerías del proyecto Lucene que se ha comentado anteriormente. Por lo que, con la configuración adecuada, puede generar índices compatibles con el motor de búsqueda Lucene y en consecuencia con Solr.

Nutch habitualmente puede funcionar a una de estas tres escalas: sistema de archivos local, intranet, o en la web entera. Las tres tienen características diferentes. Por ejemplo, rastrear un sistema de ficheros local es fiable en comparación con los otros dos, ya que los errores de red no se producen y las copias caché del contenido de la página no son necesarias. En contraste, el rastreo de la web entera se encuentra en el otro extremo.

3.- GEOCRAWLER: DESCRIPCIÓN Y ARQUITECTURA

Actualmente, resulta muy complicado conseguir un sistema totalmente autónomo, pues siempre será necesaria la participación del usuario para introducir o por lo menos validar los campos de metadatos menos intuitivos. Hay que tener en cuenta que algunos campos de metadatos son mucho más difíciles de averiguar o inferir que otros como por ejemplo el resumen o el título. La idea consiste en comenzar por rellenar los campos básicos de descubrimiento de forma que se puedan ejecutar búsquedas mínimas con éxito. Posteriormente se podrá evaluar la necesidad y los beneficios de completar rigurosamente el metadato de acuerdo a un estándar. Es preferible obtener los metadatos básicos e indexar los recursos en base a ellos que atascarse intentando generar exhaustivamente uno de ellos.

Como se ha comentado anteriormente, tras conseguir una descripción detallada de los recursos gracias a la integración de los proyectos OSGeo FDO y Apache Tika, es deseable, entre otras opciones de publicación, indexar los recursos y ofrecer un mecanismo que permita realizar búsquedas en base a sus características. Por lo tanto, con el objetivo de conseguir un sistema de indexación y búsqueda de recursos geoespaciales, siguiendo la metodología del mundo web, se pretende combinar un *crawler* que recopile los recursos disponibles con una herramienta de indexación y búsqueda de recursos.

Con los elementos descritos en la sección anterior es posible implementar un sistema de indexación y búsqueda, a continuación se detalla como encajan estas piezas. En primer lugar, será necesario un *crawler* que recopile los recursos disponibles. Se va a usar el *crawler* del proyecto Nutch con opciones para explorar redes extensas y generar índices compatibles con el motor de búsqueda Lucene. A partir de la lista de recursos disponibles, mediante la plataforma de extracción de metadatos obtenida tras la integración de los proyectos OSGeo FDO y Apache Tika se generarán las descripciones de los múltiples recursos. Con estas descripciones, Nutch será capaz de generar los índices de los recursos mediante las librerías del

[10] http://nutch.apache.org

proyecto Lucene, que posteriormente podrán ser utilizados por plataformas de búsqueda para sistemas más avanzados como Solr. A partir de este punto, cualquier tipo de usuario desde cualquier tipo de dispositivo debería ser capaz de realizar búsquedas sobre los recursos indexados. En la Fig. 1 se puede ver el planteamiento inicial de este sistema.

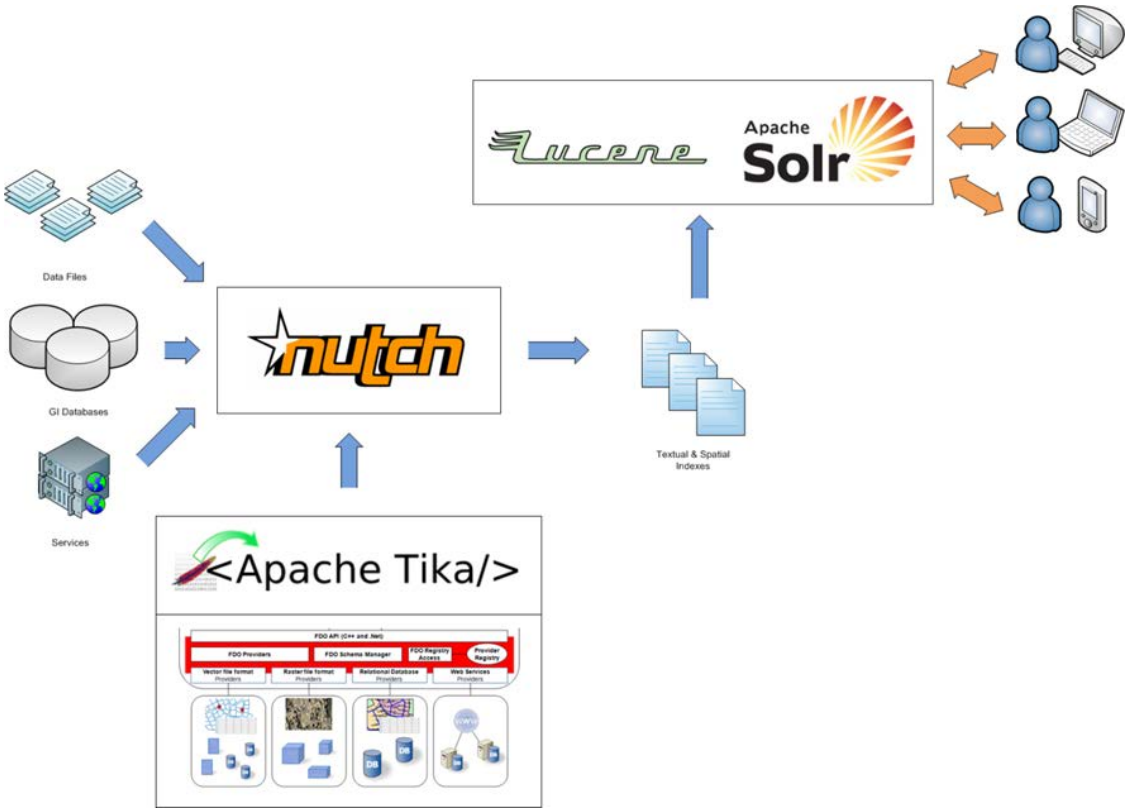

Figura 1: *Visión general del Sistema de indexación y búsqueda de recursos.*

4.- GEOCRAWLER: DETALLES DE IMPLEMENTACIÓN

Nutch no dispone de soporte para formatos geoespaciales. Viendo la arquitectura de este proyecto, la primera idea sería acoplar la solución que integra Apache Tika y OSGeo FDO en la parte *crawler* para generar las descripciones de los recursos con formatos geoespaciales y posteriormente poder indexarlos.

Para ello, se implementó un nuevo *plugin*, que, aparte de los ficheros de configuración y compilación correspondientes, podemos resumir en dos elementos. En primer lugar se implementó un nuevo analizador o *parser* que será invocado cuando se detecte que un recurso es de uno de los tipos soportados por la plataforma de extracción de metadatos en la que se va a basar. De este modo, de forma resumida, el *parser* a través de la funcionalidad de la solución que integra Apache Tika y OSGeo FDO generará y asociará al nuevo recurso encontrado todos los metadatos que es capaz de generar. En segundo lugar se implementó la parte de indexación (*IndexingFilter*), que permite generar y filtrar los contenidos que se van a indexar.

Para entender mejor este elemento, será necesario conocer un poco más acerca del funcionamiento de Nutch y acerca del motor de índices Apache Lucene. Lucene basa su funcionamiento en los elementos nombrados como Documentos (*Document*),

que se componen de uno o varios Campos (*Fields*). Un Documento puede ser visto como un registro del índice y los Campos son los elementos básicos de indexación. Los Campos siguen la filosofía de propiedades en pares (nombre, valor). Aunque estos campos no pueden ser indexados por si solos (deben ser agrupados en Documentos), las consultas se realizarán en referencia a ellos. Por lo tanto, a alto nivel, un Documento se correspondería con el concepto del registro de metadatos y los Campos con los elementos de metadatos de este registro.

Nutch para generar los índices asocia un Documento de Lucene a cada uno de los recursos encontrados. El nuevo elemento del *plugin*, en base a los metadatos que se han asociado al recurso mediante el *parser* recopilará, filtrará y formateará de forma adecuada la información que se va a incluir en el índice, es decir añadirá los Campos de forma apropiada al Documento asociado al recurso.

En este punto, dada la naturaleza espacial de los recursos, se desean combinar índices textuales y espaciales, aportando estos últimos un gran valor añadido a las búsquedas al poder restringir los resultados en un contexto geográfico. Para ello, se va a seguir una estrategia de indexación espacial que se pueda utilizar de forma conjunta con los índices textuales de Lucene. Básicamente, en la fase de indexación se deben obtener las variables X_{max}, Y_{max}, X_{min}, Y_{min} correspondientes a la caja envolvente que contiene al recurso y sobre las que posteriormente se realizarán las consultas, lo que quiere decir que para cada recurso el *parser* deberá extraer esa información y el filtro incluir esos cuatro campos en los documentos.

A parte de la implementación del *plugin*, hay algunos archivos de configuración de Nutch que se deben tener en cuenta.

- *nutch-default.xml* y *nutch-site.xml*: en estos dos archivos en formato XML se especifican todos los parámetros de configuración de Nutch y sus *plugins*.
- *regex-urlfilter.txt*: este archivo contiene una serie de expresiones regulares que permiten configurar y filtrar las URLs que el *crawler* va a inspeccionar.
- *solrindex-mapping.xml*: en este archivo en formato XML se especifican las relaciones entre los campos creados por Nutch y sus *plugins* con los campos definidos y esperados por Solr.

Tras desarrollar el *plugin* y configurar Nutch, solo faltará configurar de forma apropiada Solr. Para poner a funcionar la plataforma de búsqueda la mayor parte de la configuración se va a realizar en su esquema (*schema.xml*).

En el archivo *schema.xml*, en primer lugar, se realiza una extensa definición de tipos de datos. En cada definición de un tipo de datos se especifica un nombre único que será usado en las definiciones de los campos y posteriormente varios atributos que determinarán el comportamiento real del tipo. Entre estos atributos, debemos destacar el atributo *class* que indicará en cual de los tipos de datos implementados y soportados por Solr se va a basar el tipo de datos que se está definiendo. Posteriormente otros atributos nos permitirán configurar el comportamiento del tipo de datos en aspectos como cuál será el criterio de la ordenación en los resultados si el recurso no tiene valor en el campo de este tipo de datos u otros aspectos específicos de ciertos tipos de datos como la precisión de los datos numéricos. Además para los tipos de datos basados en texto es posible especificar de forma personalizada los analizadores que se van a utilizar, la forma de particionar el texto y los filtros que se van a aplicar. Todo esto permite, en base a los tipos de datos básicos (texto, numéricos, fecha…), especificar de forma muy concreta cómo van a ser analizados e indexados los valores de los campos y permite ajustar de una forma muy precisa el comportamiento y las prestaciones, tanto en rendimiento como en precisión de búsqueda posterior.

Posteriormente a la definición de tipos de datos se realiza la definición de los campos que se van a incluir. Cada campo es identificado por un nombre único, y mediante sus atributos se especifica su tipo de datos y otros aspectos como si va a ser almacenado, indexado o es un campo requerido.

Finalmente, se especifica cual es el campo identificador y por lo tanto único de los documentos. Por otra parte, se indicarán otros aspectos como el campo de búsqueda por defecto, para cuando en las búsquedas no se especifica un campo concreto, la operación por defecto y además se puede indicar cómo se forman algunos campos que se componen de otros. Esto último resulta muy útil para, por ejemplo, acumular en un solo campo todos los valores de los campos textuales más interesantes y definir este último como campo de búsqueda por defecto.

Con todo esto sistema de indexación y búsqueda ya es funcional. Solo queda por conocer cuáles son los campos que se han definido en el índice. La Tabla 1 lista dichos campos junto con su tipo de datos (básico) y su descripción. Los campos cuyo nombre empieza por "*gc_*" son los que proporciona el *plugin* que se ha implementado.

Campo	Tipo	Descripción
id	String	Identificador único del recurso (coincide con su URL)
host	URL	URL del host del recurso
site	String	Site del recurso
tstamp	Date	Fecha de indexación
anchor	String	Enlaces que contiene el recurso
type	String	*MIME Type* del recurso
contentLength	Long	Tamaño del archivo (si lo es)
date	Date	Fecha de creación del recurso
gc_url	URL	URL del recurso
gc_name	String	Nombre del recurso
gc_path	URL	Ruta de acceso del recurso
gc_type	String	Tipo de recurso
gc_lastModified	Date	Fecha de modificación
gc_length	Long	Tamaño
gc_description	Text	Descripción del recurso
gc_title	Text	Título del recurso
gc_keywords	Text	Palabras clave
gc_publisher	Text	Información acerca del autor del recurso
gc_schema	URL	Esquema según el tipo del recurso
gc_source	URL	Origen del recurso
gc_resourceName	String	Nombre del principal esquema interno del recurso
gc_resourceDescription	Text	Descripción del esquema interno del recurso
gc_resourceTitle	Text	Título del esquema interno del recurso
gc_resourceMinX	Double	Coordenada X_{min} del BBOX
gc_resourceMinY	Double	Coordenada Y_{min} del BBOX
gc_resourceMaxX	Double	Coordenada X_{max} del BBOX
gc_resourceMaxY	Double	Coordenada Y_{max} del BBOX
gc_resourceCRS	String	Nombre del sistema de coordenadas del recurso
gc_resourceGeometry	String	Tipo de geometría del recurso
gc_resourceNumRecords	Int	Número de registros del recurso
gc_resourceRestrictions	String	Restricciones del recurso
gc_resourceHints	String	Consejos del recurso
gc_atributeNames	String	Recopilación de los nombres de las *features*
gc_completeXML	String	XML completo generado por la plataforma de extracción de metadatos

Tabla 1: *Principales campos indexados del Sistema de indexación y búsqueda de recursos.*

5.- GEOCRAWLER: RESULTADOS

El resultado de este trabajo es un sistema de indexación y búsqueda de recursos geoespaciales que facilita al usuario el descubrimiento y la accesibilidad a esos recursos. Esto se ha conseguido, por una parte, tras la implementación de un nuevo *plugin* para el *crawler* del proyecto Nutch basado en la plataforma de extracción de metadatos obtenida tras la integración de los proyectos OSGeo FDO y Apache Tika para generar las descripciones de recursos con formatos geoespaciales. Y por otra parte, se ha implementado y se han realizado todas las modificaciones necesarias para integrar un índice espacial dentro de los índices textuales del proyecto Lucene. Estos índices serán creados desde Nutch y plenamente accesibles desde la plataforma de búsquedas Solr dado que ambas solucionas se basan en las librerías del proyecto Lucene y gracias a la configuración efectuada.

Este sistema, dada la naturaleza espacial de los recursos, al combinar índices textuales y espaciales aportará un gran valor añadido a las búsquedas de recursos geográficos al poder restringir los resultados en un contexto geográfico.

De forma resumida, el sistema es capaz de recopilar todos los recursos interesantes, (archivos, bases de datos y servicios) existentes en el ámbito en el que se ejecute (máquina local, intranet, red abierta). Y tras obtener las descripciones del módulo que integra los proyectos Apache Tika y OSGeo FDO es capaz de generar los índices textuales y espaciales en base a ellas. Gracias a estos índices, mediante las interfaces de usuario y los algoritmos de búsqueda del proyecto Solr, las consultas ejecutadas por los usuarios podrán filtrar los resultados de acuerdo a las características específicas que tiene la información geográfica y especialmente por su localización.

Gracias al proyecto Solr se ha obtenido una potente y eficiente plataforma de búsqueda para el sistema con capacidades avanzadas. Las principales ventajas es que se trata de un sistema escalable, flexible y adaptable mediante la configuración en archivos XML y extensible gracias a su arquitectura basada en *plugins*, por lo que puede ser adaptado a cualquier caso de uso de forma específica. Otro de los puntos fuertes es que sus interfaces se basan en estándares abiertos y podemos realizar consultas mediante interfaces tipo REST o HTTP y recibir las respuestas en formatos como XML o JSON que hacen que sea fácil de usar desde virtualmente cualquier lenguaje de programación.

De este modo será relativamente sencillo ofrecer interfaces de usuario más amigables y visuales basadas en la plataforma de búsqueda del sistema. Es especialmente interesante para consultas de carácter geoespacial ya que se podría obtener la caja envolvente desde un mapa como el que ofrece el proyecto OpenLayers[11] o desde un globo virtual como el que ofrece el proyecto Nasa World Wind[12]. Aunque todo esto ya queda fuera del alcance de este trabajo.

Solr ofrece una pequeña interfaz de administración que ha sido muy útil para realizar todas las pruebas de este proyecto. Esta interfaz web permite realizar consultas y depurar los resultados mediante opciones como la explicación de la puntuación del documento (ver Fig. 2 izquierda). Además ofrece un visor interactivo del esquema (*schema.xml*) cargado que además de toda la información sobre los tipos de datos y los campos definidos incluye varias estadísticas de indexación (ver Fig. 2 derecha). Finalmente la interfaz, a través de varias páginas ofrece gran variedad de información y estadísticas del esquema, la configuración, indexación, utilización, actualización y búsquedas.

[11] http://openlayers.org
[12] http://worldwind.arc.nasa.gov/java

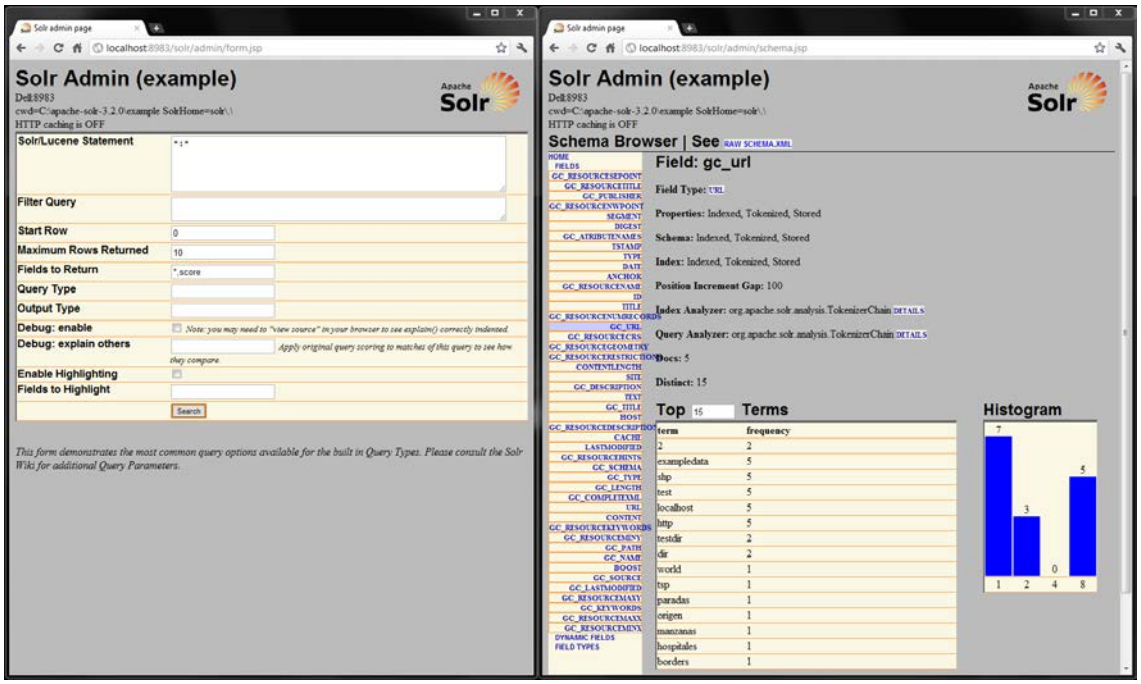

Figura 2: *Interfaz de administración de Solr.*

Finalmente, a modo de resumen, destacar que el resultado de este trabajo es un sistema de indexación y búsqueda de recursos geoespaciales en el cual, de forma paralela a la línea de evolución que ha seguido el mundo web, los metadatos, a pesar de su papel fundamental, serán totalmente transparentes al usuario permitiendo recopilar, describir, indexar/catalogar y publicar de una forma eficiente los recursos disponibles. De modo que la información esté disponible globalmente y pueda ser encontrada fácilmente por el mayor número de personas como sea posible.

6.- CONCLUSIONES

Hoy en día, pese a la creciente cantidad y variedad de recursos georreferenciados disponibles en la web, todavía resulta muy complicado encontrar contenidos georreferenciados relevantes de una forma integrada y sencilla pese a los esfuerzos realizados en generar grandes catálogos de metadatos. Viendo el éxito alcanzado por la metodología *"Google"* para los recursos del mundo web, se pueden imaginar los beneficios que estas técnicas aportarían en el contexto de la información geográfica. En consecuencia, se ha trabajado de forma paralela a la línea de evolución que ha seguido el mundo web, consiguiendo que los metadatos, a pesar de su papel fundamental, sean tan transparentes al usuario como sea posible, y a su vez permitan recopilar, describir, indexar/catalogar y publicar de una forma eficiente los recursos disponibles. En este sentido, se ha investigado sobre la recopilación de recursos mediante aplicaciones de tipo *crawler*, la descripción de los recursos mediante la generación automática de metadatos, la indexación que permite organizar los recursos y finalmente sobre estrategias de publicación. El resultado es un sistema de indexación y búsqueda de recursos geoespaciales. Este sistema se ha implementado sobre el *crawler* del proyecto Apache Nutch, consiguiendo las descripciones de los recursos mediante la plataforma de extracción de metadatos obtenida tras la integración de los proyectos OSGeo FDO y Apache Tika, indexando los metadatos gracias a las librerías del proyecto Lucene y finalmente ofreciendo una plataforma de búsquedas basada en Solr. Esta solución ofrece una potente interfaz de búsqueda,

que puede ser fácilmente adaptable a interfaces de usuario más amigables mediante visores de mapas o globos virtuales. De modo que la información esté disponible globalmente y pueda ser encontrada fácilmente por el mayor número de personas como sea posible. Cumpliendo así con el objetivo final de facilitar al usuario el acceso a los recursos.

Resulta esencial impulsar la investigación en todos los campos relacionados con la generación y gestión de metadatos dado el papel fundamental que estos juegan en cualquier sistema de información. Estos metadatos permitirían indexar o catalogar los recursos de una forma más exacta en los sistemas de información y, en consecuencia, tendremos la capacidad de proporcionar resultados más relevantes y exactos a las búsquedas realizadas por los usuarios.

AGRADECIMIENTOS

Este trabajo ha sido parcialmente financiado por una beca predoctoral de la Universitat Jaume I (PREDOC/2008/06), por una ayuda de movilidad de la Fundació Caixa Castelló-Bancaixa (E-2011-12) y por el proyecto “España Virtual” (ref. CENIT 2008-1030) a través del Instituto Geográfico Nacional (IGN) de España.

REFERENCIAS

[1] A. Beltran, C. Granell, J. Huerta (2011) *Descripción de recursos multimedia georreferenciados*. Actas de las V Jornadas de SIG Libre (SIG Libre 2011). Girona, Spain, Mar 2011. ISBN: 978-84-694-1624-2.

[2] J. Nogueras-Iso, FJ. Zarazaga-Soria, PR. Muro-Medrano (2005) *Geographic Information Metadata for Spatial Data Infrastructures: Resources, Interoperability and Information Retrieval*. Springer 2005, XXII, 264 p. ISBN 978-3-540-24464-6

[3] PC. Smits and A. Friiss-Christensen (2007) *Resource Discovery in a European Spatial Data Infrastructure*. IEEE Transactions on Knowledge and Data Engineering 19(1) pp. 85-95.

[4] R. Tolosana-Calasanz, JA. Alvarez-Robles, J. Lacasta, J. Nogueras-Iso, PR. Muro-Medrano, y FJ. Zarazaga-Soria. (2006) *On the Problem of Identifying the Quality of Geographic Metadata. Research and Advanced Technology for Digital Libraries*. Capítulo en libro Research and Advanced Technology for Digital Libraries, ECDL 2006, pages 232–243. Number 4172 in Lecture Notes in Computer Science. Springer-Verlag, 2006.

[5] MA. Manso, J. Nogueras-Iso, MA. Bernabé, y FJ. Zarazaga-Soria (2004) *Automatic metadata extraction from geographic information*. In Proc. of the 7th AGILE conference on Geographic Information Science, pages 379–385, Heraklion, Greece, 29 April -1 May 2004.

[6] MA. Manso Callejo y MA. Bernabé Poveda (2009) *Metadatos implícitos de la información geográfica: caracterización del coste temporal y de los tipos y tasas de errores en la compilación manual*. GeoFocus, (9):317–336, 2009.

[7] MA. Manso y A. Beltran (2012) *Automatic Metadata Generation for Geospatial Resource Discovery*. Capítulo de libro en "Discovery of Geospatial Resources: Methodologies, Technologies, and Emergent Applications." IGI Global, 2012. Web. 7 Mar. 2012. doi:10.4018/978-1-4666-0945-7

[8] Apache Software Foundation (2010) *Apache Tika: a content analysis toolkit*. http://tika.apache.org, 2010.

[9] Open Source Geospatial Foundation (2010) *Feature Data Objects (FDO) Data Access Technology*. http://fdo.osgeo.org, 2010.