LBD LOCAL: Un Sistema para la Recuperación de Documentos con Referencias Geográficas

Miguel R. Luaces[(1)], Jose R. Paramá[(1)], Oscar Pedreira[(1)] y Diego Seco[(1)]

[(1)]Laboratorio de Bases de Datos, Universidad de A Coruña, Campus de Elviña, 15071 A Coruña, España, {luaces, parama, opedreira, dseco}@udc.es.

RESUMEN

Tanto los Sistemas de Información Geográfica *como la* Recuperación de Información *han sido campos de investigación muy importantes en las últimas décadas. Recientemente, un nuevo campo de investigación llamado* Recuperación de Información Geográfica *ha surgido fruto de la confluencia de estos dos campos. El objetivo principal de este campo es definir estructuras de indexación y técnicas para almacenar y recuperar documentos de manera eficiente empleando tanto las referencias textuales como las referencias geográficas contenidas en el texto.*

En este artículo presentamos la arquitectura de un sistema para recuperación de información geográfica y definimos el flujo de trabajo para la extracción de las referencias geográficas de los documentos. Presentamos además una nueva estructura de indexación que combina un índice invertido, un índice espacial y una ontología. Esta estructura mejora las capacidades de consulta de otras propuestas.

Palabras clave: *Recuperación de Información Geográfica (GIR), estructura de indexación, referencias geográficas.*

ABSTRACT

Both Geographic Information Systems and Information Retrieval have been very active research fields in the last decades. Lately, a new research field called Geographic Information Retrieval has appeared from the intersection of these two fields. The main goal of this field is to define index structures and techniques to efficiently store and retrieve documents using both the text and the geographic references contained within the text.

We present in this paper the architecture of a system for geographic information retrieval. It defines a workflow for the extraction of the geographic references in the document. In addition, a new index structure is defined that combines an inverted index, a spatial index, and an ontology. This structure improves the query capabilities of other proposals.

Keywords: *Geographic Information Retrieval (GIR), index structure, geographic references.*

INTRODUCCIÓN

Aunque el campo de investigación de *Recuperación de Información* [1] ha estado activo las últimas décadas, la creciente importancia de Internet y de la *World Wide Web* ha hecho de él uno de los campos de investigación más importantes hoy en día. Se han propuesto muchas estructuras de indexación, técnicas de compresión y algoritmos de recuperación diferentes en los últimos años. Estas propuestas se han empleado generalmente en la implementación de bases de datos documentales, bibliotecas digitales y motores de búsqueda en el web.

Otro campo que ha recibido mucha atención en los últimos años es el de los *Sistemas de Información Geográfica* [2]. Las mejoras recientes en el hardware han hecho posible que la implementación de este tipo de sistemas sea abordable por muchas organizaciones. Además, se ha llevado a cabo un esfuerzo colaborativo por dos organismos internacionales (*ISO* [3] y el *Open Geospatial Consortium* [4]) para definir estándares y especificaciones para la interoperabilidad de los sistemas. Este esfuerzo ha hecho posible que muchas organizaciones públicas estén trabajando en la construcción de *infraestructuras de datos espaciales* [5] que les permitirán compartir su información geográfica.

Estos dos campos de investigación han avanzado independientemente en las últimas décadas. Sin embargo, muchos de los documentos almacenados en bibliotecas digitales y bases de datos documentales incluyen referencias geográficas en sus textos. Por ejemplo, las noticias de prensa hacen referencia al lugar donde tuvo lugar el evento y, a menudo, al lugar donde ha sido escrito el documento. Además, la información contenida en una infraestructura de datos espaciales contiene a menudo documentos con información geográfica tales como las licencias de construcción o información sobre planeamiento urbanístico. Finalmente, las referencias geográficas se pueden extraer también de páginas web usando la información del texto que contienen, la localización del servidor web y muchos otros elementos de información.

Sin embargo, aunque sea muy común que la información textual y la geográfica sea almacenada conjuntamente en los sistemas de información, las referencias geográficas de los documentos son usadas pocas veces en los sistemas de recuperación de información. Pocas estructuras de indexación o algoritmos de recuperación tienen en cuenta la naturaleza espacial de las referencias geográficas embebidas en los documentos. Las técnicas puramente textuales se centran sólo en aspectos del lenguaje de los documentos y las técnicas puramente espaciales se centran sólo en los aspectos geográficos de los documentos. Ninguna de estas técnicas es adecuada para una aproximación combinada a la recuperación de información porque ignoran completamente el otro tipo de información. Como resultado, hay una falta de arquitecturas de sistemas, estructuras de indexación y lenguajes de consulta que combinen ambos tipos de información.

Algunas propuestas que han aparecido recientemente [6, 7] definen nuevas estructuras de indexación que tienen en cuenta tanto los aspectos textuales como los geográficos de un documento. Sin embargo, las aproximaciones descritas en estos trabajos no tienen en cuenta algunas particularidades específicas del espacio geográfico. En particular, conceptos como la naturaleza jerárquica del espacio geográfico y las relaciones topológicas entre los objetos deben ser consideradas para representar completamente las relaciones entre los documentos y para permitir que se puedan realizar nuevos e interesantes tipos de consulta a estos sistemas.

En este artículo presentamos una arquitectura de un sistema de recuperación de información y una estructura de indexación que tienen en cuenta estas cuestiones. Primero, en la sección *Trabajo Relacionado*, se describen algunos conceptos básicos y trabajo relacionado. A continuación, en la sección *Arquitectura del Sistema*,

presentamos la arquitectura general del sistema y describimos sus componentes. Además, en la sección *Tipos de Consultas Soportadas*, describimos algunos tipos de consulta que pueden ser contestadas con este sistema y esbozamos los algoritmos que se pueden emplear para resolver estas consultas. Finalmente, la sección *Conclusiones* presenta algunas conclusiones y futuras líneas de trabajo.

TRABAJO RELACIONADO

Los *índices invertidos* son considerados como la técnica de indexación de texto clásica. Un índice invertido asocia a cada palabra en el texto (organizado como un *vocabulario*) la lista de punteros a las posiciones donde la palabra aparece en los documentos. El conjunto de todas las listas se llama *ocurrencias* [1]. El principal inconveniente de esta técnica es que ignora por completo las referencias geográficas. Los nombres de lugar son considerados simplemente como palabras. Si el usuario realiza una consulta del tipo *hoteles en España*, el nombre de lugar *España* es considerado una palabra y sólo se recuperan aquellos documentos que contengan esa palabra. Sin embargo, un documento que contenga nombres de ciudades de España pero no la palabra exacta *España* no es recuperado porque no se ajusta a la consulta textual.

En cuanto a la indexación de información geográfica, se han propuesto una gran variedad de estructuras de indexación espacial a lo largo de los años. En [8] se puede encontrar un buen resumen de esas estructuras. El objetivo principal de las estructuras de indexación espacial es mejorar el tiempo de acceso a las colecciones de objetos con datos geográficos. Una de las estructuras de indexación espacial más populares y un ejemplo paradigmático es el R-Tree [9]. Un inconveniente de estas estructuras es que no tienen en cuenta la jerarquía del espacio. Los nodos internos en la estructura carecen de significado en el mundo real, sólo tienen significado para la estructura de indexación. Por ejemplo, supongamos que queremos construir un índice para una colección de países, provincias y ciudades. Estos objetos están estructurados en una relación topológica de contenido, esto es, una ciudad está contenida en una provincia que a su vez lo está en un país. Si nosotros construimos un R-Tree con estos objetos geográficos la jerarquía de contenidos no se mantendrá. Los nodos internos del R-Tree no representan provincias o países y, por tanto, el índice no mantiene la jerarquía del espacio. No se puede asociar información al nodo de una provincia y que las ciudades que contiene hereden esa información porque no existe ninguna relación entre una provincia y sus ciudades en la estructura del R-Tree.

Se han realizado algunos trabajos para tratar de combinar ambos tipos de índices. Los artículos sobre el proyecto SPIRIT (*Spatially-Aware Information Retrieval on the Internet*) [10, 11, 12, 13, 14] son un muy buen punto de partida para comenzar. En [13], los autores concluyen que manteniendo separado el índice espacial del índice textual, en lugar de combinarlos en un único índice, se consigue un menor coste de almacenamiento aunque, por contra, podría implicar mayores tiempos de respuesta. Más recientes son los artículos [6, 7] que resumen este trabajo y proponen mejoras al sistema y a los algoritmos empleados en el mismo. En su trabajo proponen dos algoritmos como base: *Text-First* y *Geo-First*. Ambos algoritmos emplean la misma estrategia, primero se emplea un índice para filtrar los documentos (el índice textual en el Text-First y el índice espacial en el Geo-First). El conjunto de documentos resultante es ordenado por sus identificadores y posteriormente filtrado usando el otro índice (el índice espacial en Text-First y el índice textual en Geo-First). Sin embargo, ninguna de estas aproximaciones tiene en cuenta las relaciones entre los objetos geográficos que están indexando.

Una estructura que puede describir adecuadamente las características específicas del espacio geográfico es una *ontología*, la cual se define como una especificación explícita y formal de una conceptualización compartida [15]. Una ontología

proporciona un vocabulario de clases y relaciones para describir un ámbito determinado. En [16], se propone un método para el mantenimiento efectivo de ontologías con muchos datos espaciales usando un índice espacial para mejorar la eficiencia de las consultas espaciales. Además, en [11, 14] los autores describen cómo se emplean ontologías en tareas de expansión de los términos de las consultas (*query expansion*), en la elaboración de rankings de relevancia y en la anotación de recursos web en el proyecto SPIRIT. Sin embargo, hasta donde nosotros sabemos, nadie ha tratado de combinar ontologías con otros tipos de índices para obtener una estructura híbrida.

ARQUITECTURA DEL SISTEMA

La Figura 1 muestra nuestra propuesta para la arquitectura de un sistema de recuperación de información geográfica. La arquitectura se puede dividir en tres capas independientes: el flujo de trabajo para la construcción del índice, los servicios de procesado y las interfaces de usuario. La parte inferior de la figura muestra el flujo de trabajo para la construcción del índice, el cual, a su vez consta de tres módulos: el módulo de abstracción de documentos, el módulo de construcción del índice y la propia estructura de indexación. En las secciones que siguen se describen en mayor detalle estos componentes.

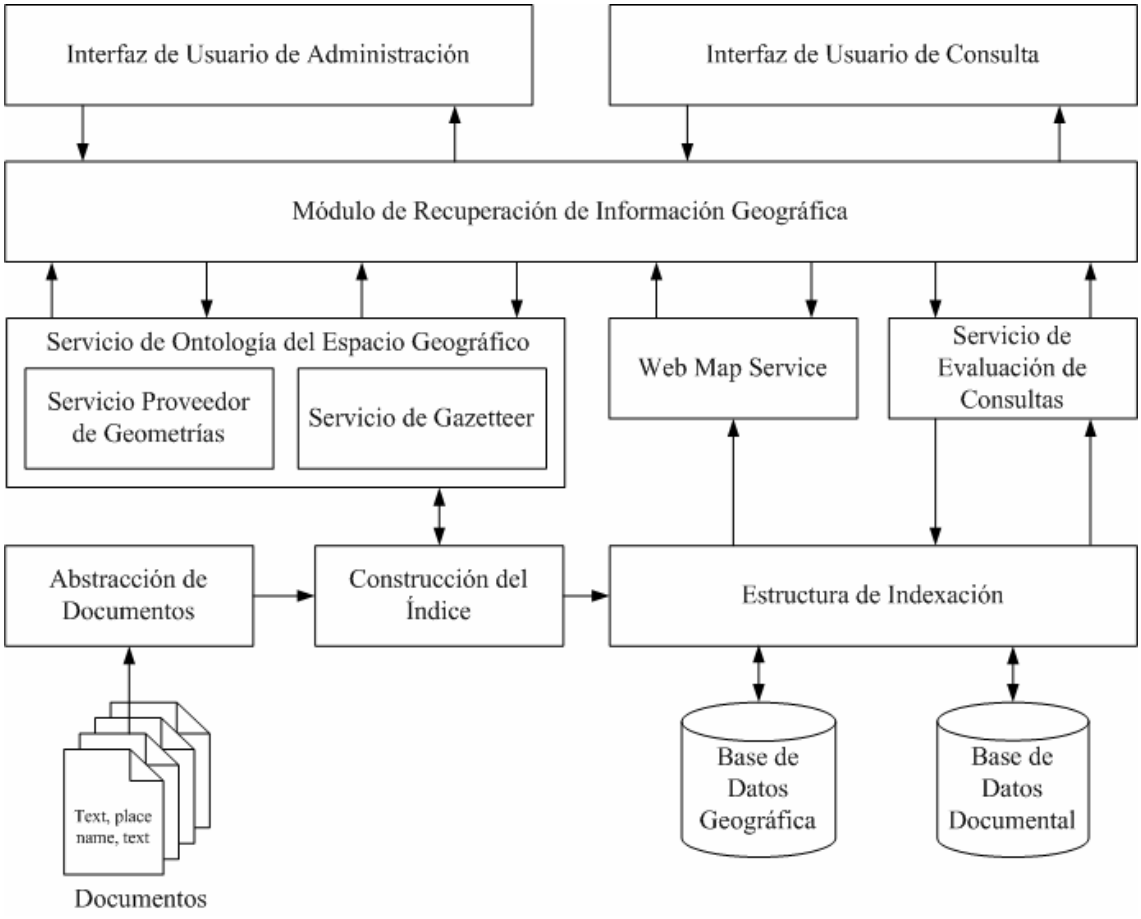

Figura 1: *Arquitectura del Sistema*

En el medio de la figura se muestran los servicios de procesado. En el lado izquierdo se muestra el *Servicio de Ontología del Espacio Geográfico* que se emplea en la construcción del índice espacial. Este servicio se describe en la sección

Indexación Espacial donde se explica el proceso de construcción de la estructura de indexación. En el lado derecho se muestran dos servicios empleados para resolver consultas. El situado más a la derecha es el *Servicio de Evaluación de Consultas*, el cual recibe consultas y usa la estructura de indexación para resolverlas. La sección *Tipos de Consultas Soportadas* describe los tipos de consultas que puede resolver este servicio, así como los algoritmos empleados para resolver esas consultas. El otro servicio es un *Web Map Service* siguiendo las especificaciones del OGC [17] que se emplea para crear representaciones cartográficas de los resultados de las consultas. Este servicio no se describe en el presente artículo. Encima de estos servicios se sitúa el *Módulo de Recuperación de Información Geográfica* que se encarga de coordinar las tareas realizadas por cada servicio para responder a las consultas de los usuarios.

La capa superior de la arquitectura muestra las dos interfaces de usuario existentes en la arquitectura: la *Interfaz de Usuario de Administración* y la *Interfaz de Usuario de Consulta*. Estas interfaces de usuario se describen en la sección *Interfaces de Usuario*.

La arquitectura propuesta es altamente modular con el objetivo de poder definir y emplear distintos componentes reusables. Para la implementación del sistema LBDLOCAL, desarrollado en base a esta arquitectura, hemos empleado algunos componentes en software libre disponibles para ciertas partes de la arquitectura y hemos desarrollado otros para donde no los había o para mejorar su funcionalidad, rendimiento, etc.

Abstracción de documentos

Dado que el sistema debe ser genérico, debe soportar la indexación de distintos tipos de documentos. Estos documentos serán diferentes no sólo en cuanto a que serán almacenados empleando formatos de archivo diferentes (texto plano, XML, etc.), sino también en cuanto a que tendrán esquemas de contenido diferentes. Un esquema de contenido podría tener un conjunto de atributos que tienen que ser almacenados en el índice (tales como *identificador del documento*, *autor*, *texto del documento*), mientras que otro esquema podría tener un conjunto de atributos diferente (tales como *identificador del documento*, *resumen*, *texto*, *autor* y *fuente*).

Para resolver este problema, hemos definido una abstracción para documentos que representa un *documento* como un conjunto de *campos*, cada uno con un valor extraído del texto del documento. Cada campo puede ser *almacenado*, *indexado* o ambas cosas. Si un campo es almacenado, sus contenidos se almacenan en la estructura de indexación y pueden ser recuperados mediante una consulta. Si es indexado, se emplea en la construcción de la estructura de indexación. Además, un campo puede ser indexado en el índice textual, en el índice espacial o en ambos índices. La definición de un documento como un conjunto de campos es similar a la empleada por el motor de búsqueda textual *Lucene* [18]. La principal diferencia reside en que hemos añadido la posibilidad de indexación espacial.

Para soportar diferentes tipos de documentos y diferentes formatos de archivo, el sistema expone la abstracción de documentos como una interfaz de programación que puede ser extendida mediante implementaciones particulares para diferentes configuraciones de formatos de archivo y esquemas de documento. Para soportar una nueva configuración, el desarrollador tan sólo tendrá que implementar la interfaz *DocumenFactory* que define las operaciones que deben ser implementadas para crear *documentos*.

Como ejemplo, para la validación del sistema, hemos indexado documentos pertenecientes a la colección del *Financial Times* [19]. Esta colección de documentos está etiquetada empleando SGML (*Standard Generalizad Markup Language*). Cada documento tiene una etiqueta <DOCNO> que contiene la cadena de caracteres de

identificación TREC y una etiqueta <TEXT> que incluye el contenido principal del documento. Para dar soporte a esta colección de documentos, hemos definido una factoría, *TRECFTDocumentFactory,* que construye documentos con dos campos. El primer campo, para el contenido etiquetado como DOCNO, es almacenado pero no indexado. El segundo campo, para el contenido de la etiqueta TEXT, no se almacena pero se indexa tanto en el índice textual como en el espacial.

Indexación textual

Como ya se ha dicho, la estructura de indexación que constituye el núcleo del sistema contiene tanto un índice textual como un índice espacial. Para implementar un índice textual empleamos *Lucene* [18]. Lucene es una librería para motores de búsqueda textuales de alto rendimiento escrito completamente en Java. Es un proyecto de código abierto que forma parte del proyecto Apache. Lucene emplea una representación objetual de los documento indexables. Un *Documento* de Lucene contiene varios *Campos*. Un campo es un par (nombre, valor) e información sobre si es almacenado y/o indexado. Los valores de los campos se establecen empleando analizadores. Estos analizadores implementan varias técnicas de recuperación de información clásicas (por ejemplo, *borrado de stopwords, stemmers, etc.*) para reducir el número de palabras indexadas y mejorar el rendimiento del índice. El analizador más sofisticado construido en el núcleo de Lucene es el *StandardAnalyzer*. Este analizador contiene reglas para direcciones de correo electrónico, acrónimos, nombres de máquina, números en punto flotante. Además convierte el valor a letras minúsculas y borra *stopwords*.

En esta etapa del proceso del flujo de trabajo, el sistema construye un índice de Lucene. Cada uno de los documentos construidos en la etapa anterior se inserta en el índice textual. El identificador de documento es almacenado pero no indexado en el índice textual y cada campo marcado como indexable en el índice textual o en ambos índices se indexa tokenizado en el índice de Lucene.

Indexación espacial

Después de construir el índice textual se debe construir el índice espacial. La indexación espacial es la tarea más compleja y comprende dos etapas. En la primera etapa el sistema extrae los nombres de lugar de los campos de los documentos marcados como indexables espacialmente y los geo-referencia. En este contexto, geo-referenciar un nombre de lugar supone no sólo obtener sus coordenadas en un sistema de coordenadas particular, sino también obtener toda la información necesaria para incluir el lugar en un índice espacial.

En la segunda etapa, se construye el índice espacial con las localizaciones geo-referenciadas en la etapa anterior junto con las referencias a los documentos que las contienen. En las siguientes subsecciones se describen estas dos etapas con más detalle.

Obtención de geo-referencias

Para extraer las referencias geográficas de los campos de los documentos marcados como indexables espacialmente se deben llevar a cabo dos tareas: analizar esos campos para obtener posibles nombres de lugar y geo-referenciar esos nombres de lugar candidatos en caso de identificarse como verdaderos topónimos.

Para la primera tarea se procesan todos esos campos mediante una técnica de *Análisis Lingüístico* conocida como *Named-Entity Recognition*. Esta técnica permite encontrar en textos menciones de categorías predefinidas tales como nombres de personas, organizaciones, lugares, etc. Nuestro módulo de *Descubrimiento de Nombres de Lugar* emplea la herramienta de lenguaje natural *LingPipe* [20] que

implementa esta técnica. LingPipe involucra el entrenamiento supervisado de un modelo estadístico para reconocer entidades. Los datos de entrenamiento deben ser etiquetados con todas las entidades de interés y sus correspondientes tipos. En la validación del sistema, con la colección del Financial Times, empleamos LingPipe entrenado con el corpus MUC6 (http://www.ldc.upenn.edu) etiquetado con lugares, personas y organizaciones. Por lo tanto, el módulo realiza un post-procesado para filtrar las entidades resultantes seleccionando sólo los lugares y descartando personas y nombres de organización.

Después de descubrir una colección de nombres de lugar candidatos, el sistema debe distinguir los falsos candidatos y geo-referenciar los verdaderos. Para este propósito hemos desarrollado un *Servicio de Ontología del Espacio Geográfico* construido empleando un *Gazetteer* y un *Proveedor de Geometrías*. La Figura 2 muestra un diagrama de clases de este componente.

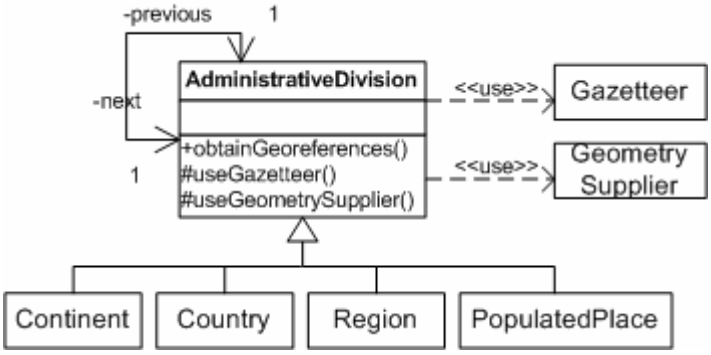

Figura 2: *Servicio de Ontología del Espacio Geográfico*

Un *Gazetteer* es un diccionario geográfico que contiene, además de nombres de lugar, otros nombres alternativos, poblaciones, localizaciones de lugares y otra información relacionada con el lugar. En la implementación de prueba empleamos *Geonames* [21] que proporciona una base de datos geográfica disponible bajo licencia *Creative Commons*. Sin embargo, Geonames (y los Gazetteers en general) no proporcionan geometrías distintas de un simple punto representativo y para nuestro índice espacial necesitamos la geometría real del lugar (por ejemplo, el borde de los países). Por ese motivo definimos un servicio *Proveedor de Geometrías* para obtener las geometrías de esos nombres de lugar. Como base de este servicio empleamos la cartografía de *Vector Map* (VMap) [22]. VMap es una actualización y versión mejorada de la cartografía proporcionada por la *National Imagery and Mapping Agency's Digital Chart of the World*. Aunque la implementación de prueba emplea Geonames y VMap, el sistema ha sido diseñado de manera que esos componentes sean fácilmente intercambiables. Todos los accesos a esos componentes se realizan mediante interfaces genéricas que pueden ser fácilmente implementadas por otros componentes.

El núcleo del servicio es una jerarquía de divisiones administrativas que incluye cuatro niveles: Continente, País, Región y Lugar Poblado, aunque resulta sencillo añadir más niveles. El algoritmo para geo-referenciar nombres de lugar involucra un recorrido descendente de la jerarquía para buscar localizaciones con el nombre consultado en cada nivel (para ello cada nivel emplea el Gazetteer con una configuración específica), y un recorrido ascendente para, una vez encontrada una localización, construir toda la ruta hasta la raíz (para lo que se emplean tanto el Gazetteer como el Proveedor de Geometrías). Por ejemplo, si el nombre de lugar consultado es *Londres*, en el recorrido descendente se obtienen varios lugares con este nombre a nivel *Lugar Poblado*. En el recorrido ascendente, se construyen varias rutas completas entre las que se encontrarán "*Londres, Inglaterra, Reino Unido, Europa"* y "*Londres, Ontario, Canadá, América del Norte"*.

Construcción del índice espacial

El componente principal de la estructura de indexación es un árbol compuesto por nodos que representan nombres de lugar. Estos nodos están conectados por medio de relaciones de inclusión (por ejemplo, Galicia está incluido en España). La estructura de árbol se construye empleando las rutas obtenidas en el proceso descrito en la sección anterior. En cada nodo almacenamos: (i) la palabra clave (un nombre de lugar), (ii) el *bounding box* de la geometría que representa ese lugar, (iii) una lista con los identificadores de los documentos que incluyen referencias geográficas a ese lugar y (iv) una lista de nodos hijo que están incluidos geográficamente en ese nodo. Si la lista de nodos hijo es muy grande, es muy ineficiente emplear un acceso secuencial. Por esta razón, si el número de nodos hijo excede un umbral, se emplea un R-Tree para mejorar el rendimiento del acceso a esos nodos hijo.

En el índice se emplean dos estructuras auxiliares. Primero, una *tabla hash de nombres de lugar* que almacena para cada nombre de lugar su posición en la estructura de indexación. Esto proporciona un acceso directo a un nodo determinado mediante una palabra clave obtenida con el Servicio de Gazetteer si la palabra procesada es un nombre de lugar. La segunda estructura auxiliar es un índice textual, con todas las palabras en los documentos, que se emplea para resolver consultas textuales (este índice está descrito en la sección *Indexación textual*).

Mantener separados los índices para el ámbito textual y para el geográfico tiene muchas ventajas. En primer lugar, todas las consultas textuales pueden ser procesadas de manera eficiente por el índice textual y todas las consultas espaciales pueden ser procesadas eficientemente por el índice espacial. Además, el sistema puede soportar consultas que combinen aspectos textuales y espaciales. También, se pueden gestionar de manera independiente actualizaciones en cada índice lo que facilita que se puedan añadir y borrar datos. Finalmente, se pueden aplicar optimizaciones a cada estructura de indexación de manera individual.

Por el contrario, la estructura presenta dos inconvenientes principales. En primer lugar, el árbol que soporta la estructura es posiblemente no balanceado penalizando la eficiencia del sistema. En segundo lugar, los sistemas ontológicos tienen una estructura fija y, por tanto, nuestra estructura es estática y debe ser construida *ad-hoc*.

Interfaces de usuario

El sistema tiene dos interfaces de usuario diferentes: una interfaz de usuario de administración y una interfaz de usuario de consulta. La primera de ellas ha sido desarrollada como una aplicación *stand-alone* y se emplea para gestionar la colección de documentos. Las funcionalidades principales son: crear índices, añadir documentos a índices, cargar y almacenar índices, etc. La pantalla principal de esta interfaz muestra información útil acerca del índice cargado (número de documentos indexados, campos de cada uno de esos documentos, número de nombres de lugar en el índice, etc.).

La Figura 3 muestra una captura de pantalla de la interfaz de usuario de consulta. Esta interfaz fue desarrollada como una aplicación web empleando el *Google Maps API* [23]. Este API proporciona gran cantidad de utilidades para manipular mapas y añadir contenido a esos mapas.

En la siguiente sección describiremos los tipos de consultas que se pueden resolver con este sistema. Estas consultas tienen dos ámbitos diferentes: un ámbito textual y otro espacial. La interfaz de usuario de consulta le permite al usuario indicar ambos ámbitos. En concreto, el ámbito espacial se puede introducir de tres maneras mutuamente excluyentes:

- *Tecleando el nombre de lugar*. En este caso, el usuario introduce el nombre de lugar en un campo de texto. Este es el método más ineficiente ya que el

sistema tiene que obtener todas las geo-referencias asociadas con el nombre de lugar tecleado y éste es un proceso costoso en tiempo.

- *Seleccionando el nombre de lugar en un árbol*. En este caso, el usuario selecciona sucesivamente un continente, un país contenido en ese continente, una región en ese país y un lugar poblado en esa región. Si el usuario quiere especificar un nombre de lugar de nivel más alto que lugar poblado no tiene que rellenar todos los niveles. La operación es muy sencilla e intuitiva porque la interfaz está implementada con un componente desarrollado a medida empleando la tecnología AJAX que permite recuperar en un segundo plano los nombres de lugar del siguiente nivel. Cuando el usuario selecciona un lugar en el componente, el mapa de la parte derecha enfoca automáticamente el lugar seleccionado.

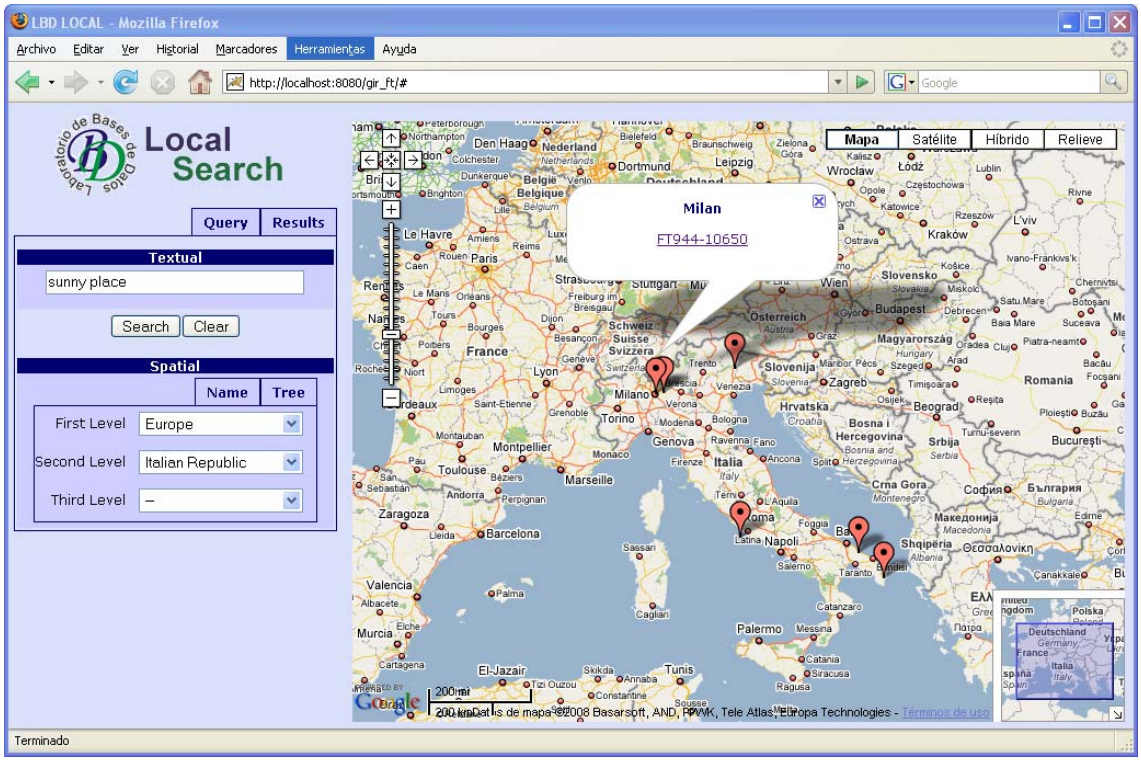

Figura 3: *Interfaz de Usuario de Consulta*

- *Visualizando el contexto espacial de interés en el mapa*. El usuario puede navegar empleando el mapa de la parte derecha para seleccionar el contexto espacial de interés. El sistema usará el *bounding box* de ese mapa como ventana de consulta si el usuario no tecleó un nombre de lugar ni seleccionó un lugar en el árbol.

TIPOS DE CONSULTAS SOPORTADAS

La característica más importante de una estructura de indexación es el tipo de consultas que se pueden resolver con él. Los siguientes tipos de consultas son relevantes en un sistema de recuperación de información geográfica:

- *Consultas puramente textuales*. Estas son consultas del tipo *"recuperar todos los documentos donde aparezcan las palabras* hotel *y* mar*"*.
- *Consultas puramente espaciales*. Un ejemplo de este tipo de consultas es *"recuperar todos los documentos que se refieran a la siguiente área*

geográfica". El área geográfica en la consulta puede ser un punto, una ventana de consulta, o incluso un objeto complejo como un polígono.

- *Consultas textuales con nombres de lugar*. En este tipo de consultas, algunas palabras son nombres de lugar. Por ejemplo, *"recuperar todos los documentos con la palabra* hotel *referidos a España"*.
- *Consultas textuales sobre un área geográfica*. En este caso se proporciona un área geográfica de interés junto con el conjunto de palabras. Un ejemplo es *"recuperar todos los documentos con la palabra* hotel *que se refieren a la siguiente área geográfica"*. Al igual que en las *consultas puramente espaciales* el área geográfica de la consulta puede ser un punto, una ventana de consulta o un objeto complejo.

Las consultas puramente textuales se pueden resolver de forma trivial porque un índice textual forma parte de la estructura de indexación. De manera similar, las consultas puramente espaciales se pueden resolver porque la estructura de indexación se construye como un índice espacial y, por tanto, se puede emplear el mismo algoritmo empleado con índices espaciales. Se desciende en la estructura teniendo en cuenta sólo aquellos nodos cuyos *bounding box* intersecan con el área geográfica de la consulta. Esta operación devuelve un conjunto de documentos candidatos que tiene que ser refinado con la referencia geográfica actual para decidir si el documento es parte del resultado o no.

Sin embargo, la estructura de indexación que proponemos puede ser usada para resolver el tercer y el cuarto tipo de consultas que no pueden ser solucionados de manera sencilla empleando un índice textual o un índice espacial. Para el caso de la consulta con nombres de lugar, nuestro sistema puede descubrir que *España* es una referencia geográfica consultando al servicio de gazetteer y posteriormente emplear la tabla hash de nombres de lugar de la estructura para recuperar el nodo del índice que representa *España*. De este modo se puede ahorrar algún tiempo de acceso suprimiendo parte del recorrido en el árbol.

Con respecto al cuarto tipo de consultas, el índice textual se emplea para recuperar la lista de documentos que contienen las palabras y la estructura de indexación espacial se emplea para obtener la lista de documentos que hacen referencia al área geográfica. Por tanto, la intersección de ambas listas es el resultado de la consulta.

Otra mejora sobre los índices textuales y espaciales es que nuestra estructura de indexación puede realizar fácilmente expansión de los términos de consulta (*query expansion*) sobre referencias geográficas porque está construida sobre una ontología del espacio geográfico. Consideremos la siguiente consulta *"recuperar todos los documentos que se refieran a España"*. El servicio de evaluación de consultas descubrirá que España es una referencia geográfica. El índice de nombres de lugar se empleará para localizar rápidamente el nodo interno que representa el objeto geográfico *España*. Entonces, todos los documentos asociados con este nodo forman parte del resultado de la consulta. Sin embargo, todos los hijos de este nodo son objetos geográficos que están contenidos en España (por ejemplo, la ciudad de Madrid). De este modo, todos los documentos referenciados por el subárbol forman también parte del resultado de la consulta. La consecuencia es que la estructura de indexación ha sido empleada para expandir la consulta porque el resultado contiene no sólo aquellos documentos que incluyen el término *España*, sino también aquellos documentos que incluyen el nombre de un objeto geográfico contenido en España (por ejemplo, todas las ciudades y regiones de España).

CONCLUSIONES

En este artículo se ha presentado una arquitectura de sistema para recuperación de información que tiene en consideración no sólo el texto contenido en los

documentos sino también las referencias geográficas incluidas en los documentos y la ontología del espacio geográfico. Esto se logra mediante una nueva estructura de indexación que combina un índice invertido, un índice espacial y una ontología. También se ha presentado cómo las consultas tradicionales pueden ser resueltas usando esta estructura de indexación. Finalmente, se han descrito nuevos tipos de consultas que pueden ser resueltos con la estructura de indexación y se han esbozado los algoritmos que permiten resolver esas consultas.

Actualmente se está trabajando en la evaluación del rendimiento del índice y están planeadas posibles mejoras futuras de la estructura de indexación. En primer lugar, se debe definir un procedimiento para decidir si los hijos de un nodo se deben estructurar como una lista o como un R-Tree. Además, está planificado incluir otros tipos de relaciones espaciales en la estructura de indexación complementarias a la de inclusión (por ejemplo, adyacencia). Estas relaciones pueden ser fácilmente representadas en la ontología y la estructura de indexación puede ser extendida para soportarlas. Otra línea de trabajo futuro implica la introducción de técnicas de *Resolución de Topónimos* para resolver los problemas de ambigüedad de los topónimos a la hora de geo-referenciar los documentos. Finalmente, es necesario definir algoritmos para elaborar el ranking de los elementos recuperados por el sistema. Para esta tarea debemos definir una medida de relevancia espacial y combinarla con la relevancia obtenida empleando el índice textual.

AGRADECIMIENTOS

Este trabajo ha sido parcialmente financiado por el “Ministerio de Educación y Ciencia” (PGE y FEDER) ref. TIN2006-16071-C03-03, por la “Xunta de Galicia” ref. PGIDIT05SIN10502PR y ref. 2006/4, por el “Ministerio de Educación y Ciencia” ref. AP-2006-03214 (Programa FPU) para Oscar Pedreira, y por la “Dirección Xeral de Ordenación e Calidade do Sistema Universitario de Galicia, da Consellería de Educación e Ordenación Universitaria-Xunta de Galicia" para Diego Seco.

REFERENCIAS

1. Baeza-Yates R, Ribeiro-Neto B (1999) Modern Information Retrieval. Addison Wesley.
2. Worboys MF (1995) GIS: A Computing Perspective. Taylor & Francis. ISBN: 0-7484-0065-6.
3. ISO (2002) Geographic Information – Reference Model. International Standard 19101, ISO/IEC.
4. OpenGIS (2002) OpenGIS Reference Model. OpenGIS Project Document 03-040, Open GIS Consortium, Inc.
5. GSDI (2007) Global Spatial Data Infrastructure Association, http://www.gsdi.org.
6. Martins B, Silva MJ, Andrade L (2005) Indexing and ranking in Geo-IR systems. In GIR’05: Proceedings of the 2005 workshop on Geographic information Re-trieval, pp. 31-34, New York, NY, USA, 2005. ACM Press.
7. Chen Y-Y, Suel T, Markowetz A (2006) Efficient query processing in geographic web search engines. In SIGMOD Conference, pp. 277-288.
8. Gaede V, Günther O (1998) Multidimensional access methods. ACM Comput. Surv., 30(2), pp. 170-231.
9. Guttman A (1984) R-Trees: A dynamic index structure for spatial searching. In B. Yormark, editor, SIGMOD'84, Proceedings of Annual Meeting, Boston, Mas-sachusetts, June 18-21, 1984, pp. 47-57. ACM Press.
10. Jones CB, Purves R, Ruas A, Sanderson M, Sester M, van Kreveld M, Weibel R. (2002) Spatial information retrieval and geographical ontologies an overview of the

SPIRIT project. In Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 387-388.
11. Jones CB, Abdelmoty AI, Fu G (2003) Maintaining ontologies for geographical information retrieval on the web. In Proceedings of On The Move to Mean-ingful Internet Systems 2003: CoopIS, DOA, and ODBASE Ontologies, Da-tabases and Applications of Semantics, ODBASE'03, vol. 2888 of Lecture Notes in Computer Science.
12. Jones CB, Abdelmoty AI, Fu G, Vaid S (2004) The SPIRIT Spatial Search En-gine: Architecture, Ontologies and Spatial Indexing. In Proceedings of the 3rd Int. Conf. on Geogrphic Information Science, pp. 125-139, vol. 3234 of Lec-ture Notes in Computer Science.
13. Vaid S, Jones CB, Joho H, Sanderson M (2005) Spatio-textual indexing for geo-graphical search on the web. In Proceedings of the 9th Int. Symp. on Spatial and Temporal Databases (SSTD), pp. 218-235, vol. 3633 of Lecture Notes in Computer Science.
14. Fu G, Jones CB, Abdelmoty AI (2005) Ontology-based spatial query expansion in Information Retrieval. In Proceedings of In On the Move to Meaningful Inter-net Systems 2005: ODBASE 2005, pp. 1466-1482, vol. 3761 of Lecture Notes in Computer Science.
15. Gruber TR (1993) A translation approach to portable ontology specifications. Knowledge Acquisition, 5(2), pp. 199-220, June 1993.
16. Dellis E, Paliouras G (2006) Management of large spatial ontology bases. In Pro-ceedings of the Workshop on Ontologies-based techniques for DataBases and Information Systems (ODBIS) of the 32nd International Conference on Very Large Data Bases (VLDB 2006), September 2006.
17. WMS (2002) OpenGIS Web Map Service Implementation Specification. Open-GIS Project Document 01-068r3, Open GIS Consortium, Inc.
18. Lucene (2007) Apache Lucene, http://lucene.apache.org.
19. TREC (2007) NIST Special Database 22, NIST TREC Document Database: Disk 4, http://www.nist.gov/srd/nistsd22.htm.
20. LingPipe (2007) Natural Language Tool LingPipe, http://www.alias-i.com/lingpipe/.
21. Geonames (2007) Gazetteer Geonames, http://www.geonames.org.
22. VMAP (2007). Vector Map Level 0, http://www.mapability.com/index1.html?http&&&www.mapability.com/info/vmap0_intro.html.
23. Google Maps (2007) Google Maps API, http://www.google.es/apis/maps/.