UNIVERSIDAD CARLOS III DE MADRID.

PROYECTO FIN DE CARRERA.

T-SEARCH: BUSCADOR CON TESAURO PARA WIKIS

AUTOR: ISMAEL SAGREDO OLIVENZA.
TUTORES: SONIA SANCHEZ CUADRADO Y JORGE MORATO LARA.

Índice.

Índice de figuras.

Índice de Tablas

Resumen

El auge de los entornos **wiki**, cada vez más extendidos, hace necesarias herramientas que permitan recuperar de forma eficiente la información de estos entornos. Hasta ahora los buscadores genéricos como **Google** son las mejores herramientas para encontrar en wikis como **Wikipedia,** pero no para wikis de menor difusión, donde Google y otros buscadores sólo tienen indizada una pequeña parte del sitio. Por otro lado, los buscadores que incorporan los entornos wiki son muy limitados, y sólo realizan búsquedas a partir del titulo o de la URL. Los buscadores que asisten a los usuarios con búsquedas de términos relacionados, son muy útiles en ciertos entornos, como por ejemplo en los entornos académicos, ya que permiten descubrir y buscar conceptos relacionados con los que se buscaban inicialmente. Esta característica puede ser muy útil en las wikis, que suelen ser utilizadas como enciclopedias.

Este proyecto implementa un buscador para entornos wiki, de código abierto, multiplataforma, escrito en Java y *PHP*, distribuido y escalable, que se utilizará para buscar sobre el wiki del departamento de Ingeniería del Software de la Universidad Carlos III de Madrid[1].

[1] Wiki a indizar http://163.117.147.74/ie/doku.php [consultado 08/05/2009]

1. Introducción

1.1 Descripción del problema

Actualmente existen varios proyectos para dotar a Wikipedia de un mejor buscador debido a la importancia que esta enciclopedia colaborativa ha adquirido en los últimos años. El problema es que actualmente ningún buscador específico realiza búsquedas eficientes en estos entornos, utilizándose normalmente buscadores genéricos como Google, Yahoo, etc., para buscar los términos en la Wikipedia.

El problema aún es más grave si intentamos buscar en un wiki de menor entidad como es el caso que nos ocupa. El wiki del departamento de Ingeniería del Software de la Universidad Carlos III de Madrid está parcialmente indizada en estos buscadores genéricos pero no completamente. Así pues la búsqueda se debe realizar utilizando el buscador que aporta la herramienta DokuWiki, con la que se ha generado el wiki. Este buscador es bastante limitado y sólo busca los términos teniendo en cuenta el nombre de los ficheros almacenados en el wiki. No soporta búsqueda booleanas y precisa de varios click para resolver ambigüedades en la búsqueda. Esto dificulta su utilización ya que aunque es un wiki pequeña, entre revisiones, discusiones, y artículos, el sitio contiene más de 13.000 páginas.

Por lo tanto se precisa de un buscador más potente y que permita búsquedas en el contenido de las páginas, y no sólo en el título de las mismas, que permita búsqueda booleanas, búsqueda de términos literales (la búsqueda entre comillas de Google) y que sea suficientemente escalable como para mantener su capacidad de búsqueda aunque el sitio aumente de tamaño.

1.2 Objetivos del proyecto

El objetivo de este proyecto es construir un buscador para entornos wikis que cumpla con las necesidades y soluciones para buscar términos escritos dentro del contenido de las páginas y no limitarse solamente a los términos incluidos en la URL o en el título. Debe incorporar un sistema de búsqueda booleana para realizar combinaciones de búsqueda del tipo:

- Busca todas las páginas que contenga la palabra X o la palabra Y
- Busca todas las páginas que contengan la palabra X y no contengan la palabra Y
- Busca las páginas que contenga el texto 'X' literalmente.

Además, para aumentar las capacidades del buscador se incluye un tesauro que puede ser ampliado y/o sustituido, dependiendo del dominio donde se aplique el buscador. Este tesauro aporta la posibilidad de hacer búsquedas por

los sinónimos de las palabras que se consultan, obteniendo mejores resultados. El tesauro también se utiliza para proponer términos relacionados con la búsqueda realizada por el usuario. Estos términos están organizados de forma jerárquica mediante relaciones de generalización y especificación de términos. Por ejemplo, el término animal es el término genérico de mamífero, que a su vez es un término específico de animal.

El buscador utiliza el tesauro tanto para obtener los sinónimos de los términos de búsqueda, como para ayudar al usuario a buscar por otros términos relacionados con la búsqueda propuesta, lo que aumenta las posibilidades de uso del buscador.

Aunque el tamaño del wiki que se va a indizar no lo requiere, se ha diseñado el sistema para que sea fácilmente escalable. Si el índice crece demasiado, puede ser distribuido en varios equipos, lo que reduce tiempo tanto para la indización como para la consulta.

1.3 Definiciones, acrónimos y abreviaturas

- ADOdb es un conjunto de librerías de bases de datos para PHP y Python. ADOdb permite a los programadores desarrollar aplicaciones Web de una manera portable, rápida y fácil.
- Antonimia: se dice que dos palabras tienen una relación de antonimia cuando sus significados son antagónicos.
- Blog o bitácora: es un sitio Web periódicamente actualizado que recopila cronológicamente artículos de uno o de varios autores.
- CERN: es el acrónimo Conseil Européen pour la Recherche Nucléaire que se puede traducir por la Organización Europea para la Investigación Nuclear que es el mayor laboratorio de investigación en física de partículas del mundo situado cerca de Ginebra (Suiza).
- DTD: Document Type Definition, es una descripción de estructura y sintaxis de un documento XML.
- GCC: Compilador de código libre (GNU) para C/C++ que viene incorporado en las distribuciones de Linux.
- Hiperónimo: es aquel término general que puede ser utilizado para referirse a la realidad nombrada por un término más particular.
- Hipónimo: es aquel término que posee todos los rasgos semánticos, de otra más general, su hiperónimo, pero que añade en su definición otros rasgos semánticos que la diferencian de la segunda.
- Homonimia: Se dice que dos palabras tienen una relación de homonimia cuando su identidad fonética o gráfica es igual, pero su significado es distinto.
- IP-spoofing: técnicas de suplantación de identidad basadas en la modificación de la dirección IP de una máquina.

- J2SD (Java 2 Estándar Edition) es el paquete de desarrollo estándar para Java que se distribuye de forma gratuita en la página de Sun Microsystem.
- JDBC: Son las siglas de Java DataBase Connectivity2 que permite la conexión entre Java y cualquier base de datos simplemente disponiendo de un pequeño módulo denominado Driver.
- JSP: acrónimo de Java Server Pages es una tecnología creada por Sun Microsystem como complemento de su lenguaje de programación Java para crear páginas Web dinámicas.
- MD5: Message-Digest Algorithm 5, Algoritmo de Resumen del Mensaje 5, es un algoritmo de reducción criptográfico de 128 bits que sirve para resumir un conjunto de datos y garantizar su integridad. Si el resumen de los datos no concuerda con el resumen previamente calculado, esto indica que los datos han cambiado.
- Ontología: Es una forma de intercambio de información entre los sistemas. Se suele utilizar este término en Inteligencia Artificial para describir la representación del conocimiento en un dominio dado y para la Web como parte de la infraestructura de la Web semántica. Las ontologías, para ser definidas, de un lenguaje de descripción, en el caso de la Web se suele usar la combinación OWL y RDF para dar sentido a las etiquetas de una página Web y así permitir su tratamiento automático.
- PHP: es un acrónimo recursivo que significa PHP Hypertext Pre-procesor y que se utiliza para editar páginas Web de forma dinámica como extensión de servidores Web como apache.
- Redes sociales: es una estructura formada por individuos que es representable mediante un grafo en el que los nodos representan individuos y las aristas relaciones entre los individuos. Algunas relaciones posible son de amistad, de trabajo, de pareja, etc.
- Round robin: Es en proceso de asignación de tareas a procesos de forma racional e equitativa en la que cuando un proceso acaba con una tarea, se le proporciona la siguiente que esté en la lista de tareas pendientes sin ningún tipo de prioridad ni entre procesos ni entre tareas. A la larga, en media, se tenderá a que todos los procesos hayan procesado un número similar de tareas.
- RDF: es el acrónimo de Resource Description Framework que es un marco de trabajo para descripción de metadatos en la Web.
- Sinonimia: Se dice que dos palabras tiene una relación de sinonimia cuando sus significados son similares
- Smarty: es un motor de plantillas para PHP
- RSS: Really Simple Syndication. Sirve como forma de notificación de cambios en una página a programas que utilicen dicha sindicación.

[2] Tecnología para bases de datos en java http://java.sun.com/javase/technologies/database/ [consultado 08/05/2009]

- Taxonomías: Es un recurso que permite organizar el conocimiento de forma jerárquica. Se suele utilizar mucho en biología para la clasificación de animales en especies.
- TeX: sistema de tipografía Donald E. Knuth especialmente diseñado para editar fórmulas matemáticas rápidamente.
- URL: es el acrónimo de Uniform Resource Locator, es decir, localizador uniforme de recurso. Es una secuencia de caracteres, de acuerdo a un formato estándar, que se usa para nombrar recursos, como documentos e imágenes en Internet, por su localización.
- XML: eXtensible Markup Language. Es un metalenguaje que permite definir las etiquetas y la gramática de lenguajes específicos. Por lo tanto, XML no es realmente un lenguaje sino un conjunto de lenguajes que tienen ciertas características en común. XML es un subconjunto de SGML del que se creó HTML, pero sus reglas para definir la gramática son mucho más estrictas, lo que le hacen más fácil de validar. De hecho es posible construir analizadores sintácticos automáticos a partir de una definición en una DTD (Documento de definición de tipos).
- Diagrama de Gantt: popular herramienta gráfica cuyo objetivo es mostrar el tiempo de dedicación previsto para diferentes tareas o actividades a lo largo de un tiempo total determinado

2. Estado del Arte

En este capítulo se describirá el estado actual de los entornos de publicación Web colaborativa más famosos de la actualidad, denominados entornos wiki; sus ventajas y las carencias que actualmente estos sistemas poseen. Por otro lado, se describirá de forma breve los principales conceptos sobre buscadores: sus diferentes tipos, sus características, así como la ventaja que supone incorporar cierto tipo de buscadores en un wiki.

Por último se describirá de forma breve los conceptos de ontología, Web semántica y tesauro, así como las ventajas que estas tecnologías y herramientas pueden aportar tanto a los entornos wiki como a los buscadores.

2.1 Entornos wiki

2.2.1 ¿Qué es un wiki?

Los entornos wiki son marcos de edición colaborativos de documentos de hipertexto. Los usuarios de un wiki pueden editar (crear, modificar o borrar) el contenido de una página del sitio wiki de forma interactiva, lo que facilita enormemente la escritura en grupo de estos documentos. La escritura en grupo permite la edición de un documento por parte de uno o varios autores de forma simultánea. Esta forma de escribir documentos no es nueva ni surge necesariamente de los avances tecnológicos actuales, pero su difusión actual no puede ser entendida sin la progresión que ha experimentado la Web, sobre todo desde la aparición de la Web social (o también denominada Web 2.0).

El concepto de Web 2.0 no es análogo al de Web semántica, a pesar de que en muchas ocasiones se utilizan ambos conceptos para referirse al actual estado de la Web. El término Web 2.0 fue acuñado por O'Reilly Media en 2004 para referirse a una segunda generación de aplicaciones Web basada en comunidades de usuarios. Algunos ejemplos de dicha tecnología son las denominadas redes sociales, los blogs y también los wikis.

Con la Web 2.0 se pretende construir sitios Web que actúan como puntos de encuentro entre un grupo más o menos heterogéneo de usuarios, en contraste con la Web tradicional en la que la participación de los usuarios y la interacción entre ellos es más limitada.

Sin embargo, la Web semántica es una meta a alcanzar para incorporar a entidades software como usuarios del ciber espacio, a un nivel similar al que lo hacen los seres humanos. Esta idea futura se basa en la utilización de ontologías

y la estandarización de los lenguajes de definición de documentos (por ejemplo XML) y tiene como objetivo que los usuarios software puedan *comprender* los contenidos de la Web, de forma que la información sea procesada mucho más rápidamente y de forma más simple y completa por dichos agentes de forma que se automatice el procesamiento de la información almacenada en la Web. La Web 2.0 utiliza algunas ideas atribuidas a la Web semántica como la tecnología *XML* (por ejemplo *RSS* para avisar a los usuarios de sitios que se actualicen de forma continua), pero en muy pocas ocasiones se han utilizado ontologías.

Por tanto, un wiki es un tipo de aplicación de la denominada Web 2.0. Este término proviene del hawaiano *wiki wiki* que significa rápido y expresa la rapidez y facilidad con la que los usuarios de estos sistemas pueden modificar el contenido o el aspecto de una página. El término fue acuñado por primera vez por **Ward Cunningham** en 1995 [1] .

Las páginas de un wiki son editadas mediante texto plano utilizando etiquetas simples para aplicar los estilos. Normalmente no se suele utilizar XML para su edición para facilitar su utilización por personas no técnicas. Los wikis poseen una estructura de navegación no lineal ya que cada página contiene un gran número de vínculos a otras páginas. La primera convención de marcado para vincular las páginas de un wiki fue **CamelCase**. Esta convención consistía en describir frases o palabras compuestas todas juntas, poniendo la primera letra de cada palabra en mayúsculas. Este método se desechó con el tiempo porque se aleja de la escritura normal y se diseñaron otras formas de vincular páginas. Así surgió **Free Links** una forma de mostrar links mucho más sencilla utilizando la siguiente sintaxis: _(titulo_pagina). Este método es válido para un mismo entorno wiki, ya que no es necesario describir la localización de las páginas si se posee un índice de páginas.

Para describir vínculos entre diferentes sistemas wiki, se suelen utilizar **Interwiki**. Crear un vínculo con Interwiki genera una página Web sin edición y lo muestra como un vínculo roto (por ejemplo en Wikipedia el vínculo roto se muestra de color rojo). Los enlaces de Interwiki se describen así:

```
[[nombre_sitio:nombre_pagina]]
```

Y es necesario tener una base de datos con la URL que corresponda. Si alguien accede a un vínculo roto se mostrará una página de edición para escribir el contenido de dicha página, si por el contrario ya hay algo escrito en dicha página aparecerá con su color habitual.

Generalmente, la edición de las páginas de un wiki está abierta a cualquier usuario, aunque en algunos casos se precisa estar registrado en el sitio para modificar las páginas. Los artículos o páginas que se van editando normalmente no requieren de revisión, por lo que estos sistemas pueden tener problemas al publicar artículos erróneos o malintencionados. Por eso en

algunas ocasiones es preciso estar registrado para editar las páginas y así facilitar la detección de usuarios malintencionados e intentar atenuar este problema. Además, las herramientas wiki suelen mantener las páginas editadas en *cuarentena* de forma que si se detecta algún tipo de error, se pueda volver de nuevo al estado anterior de la Web o en algunas ocasiones, las herramientas wiki mantienen una lista de **revisiones** con los cambios que se han ido produciendo en un artículo. Los últimos cambios se suelen poder visualizar en una página específica denominada "cambios recientes" o se agrupan en una lista, así, junto con un historial de cambios, es muy sencillo revertir un cambio malintencionado o erróneo.

2.1.2 Herramientas de apoyo para creación de wikis

En la actualidad existen varios programas para construir de forma fácil un wiki. Normalmente este tipo de programas se basan en tecnología de script de servidor Web, como puede ser PHP o Perl. La forma de almacenar las páginas que se van realizando difiere según el sistema que se utilice. Algunas de estas herramientas utilizan una base de datos para almacenarlas, otras las guardan en forma de archivos del sistema de ficheros del servidor, pero toda está creación de páginas es siempre gestionada desde el motor del wiki, por lo que es trasparente para el usuario. Una lista de herramientas para crear wikis se puede ver en la Wikipedia[3], pero citaremos las más importantes a continuación:

- **UseModWiki**: Fue el primer software para wikis que se creó. Apareció por primera vez en el año 2000. Fue construido por Clifford Adams en Perl bajo l*icencia pública general GNU.* El motor de edición de wikis MediaWiki (uno de los motores actuales más conocidos) es derivado de UseModWiki. Una de sus características es que no almacena las páginas en una base de datos, sino en ficheros.

- **DokuWiki**: es un software de gestión de wikis distribuido bajo licencia GPL, de fácil uso y sencilla sintaxis. Toda la información se almacena en ficheros de texto plano, por lo que no requiere una base de datos. Este motor fue desarrollado por **Andreas Gohr** en PHP. Sus principales características son su soporte para ficheros multimedia, generación de índices automatizados de contenidos, gestión de usuarios, etc.

- **MediaWiki**: es un motor para wikis distribuido bajo licencia GLP escrito por **Magnus Manske** en PHP usando Apache y MySQL. Fue diseñado para desarrollar Wikipedia, un portal wiki diseñado para construir una enciclopedia colaborativa con acceso vía Web. Sus características más importantes son:

[3] Wikipedia, Software para Wikis http://es.wikipedia.org/wiki/Software_para_wikis [consultado 27/05/2009]

  - No necesita que los nombres de las páginas estén escritos en CamelCase lo que simplifica su edición.
  - Contiene páginas de discusión sobre los temas editados.
  - Incluye soporte para *TeX* con lo que se permite una buena visualización de fórmulas matemáticas.
  - Dispone de un interfaz personalizable.

- **TikiWiki**: es un sistema de gestión de wikis que dispone de un gran número de funcionalidades que mejoran la edición colaborativa como las siguientes: Chat, blogs, listas de correo, etc. Esta desarrollado por voluntarios en PHP y utiliza *ADOdb y Smarty* para su ejecución. Su uso es muy adecuado para entornos educativos.

- **JSPWiki**: es un wiki escrita en Java que está en proceso de desarrollo y que permite extender su funcionalidad incluyendo código Java de forma sencilla.

2.1.3 Wikipedia, el mejor ejemplo de un wiki

Wikipedia[4] es el wiki más conocido y utilizada del mundo y su gran expansión ha permitido que el concepto de wiki se haya extendido con gran facilidad. El proyecto original de Wikipedia era desarrollar una enciclopedia en línea editada mediante la colaboración de todos los internautas. El proyecto comenzó en 2001 y en el 2008 dispone de artículos en más de 260 idiomas. La Wikipedia, por tanto, es uno de los máximos exponentes de lo que la tecnología wiki puede realizar y de las posibilidades que puede ofrecernos. Actualmente la versión Inglesa de Wikipedia es reconocida tan fiable como la propia enciclopedia Británica en cuanto a artículos científicos. En el 2008, Wikipedia dispone de más de 11 millones de artículos almacenados en su base de datos de los que 2,6 millones son en ingles y 440.000 en español. Esta enorme enciclopedia online es gestionada por el software MediaWiki, especialmente diseñado para su construcción. Las discusiones sobre la redacción de los documentos corren en una página aparte asociada al artículo denominada *discusiones*, donde se muestra el contenido que ha sido añadido y por quien ha sido añadido. En el caso de que el artículo no haya sido firmado, se añade la dirección IP de la máquina donde se editó el artículo. Si un artículo tiene cierta controversia en su redacción, Wikipedia advierte de este hecho al inicio del artículo, esto suele suceder cuando los artículos tratan sobre temas políticos, religiosos, etc. Esta forma de redacción y el mantener la discusión sobre la misma hace a la Wikipedia una enciclopedia mucho más plural que las convencionales, dando un punto de vista consensuado que permite a los usuarios de Wikipedia tener una visión global de todas las posiciones sobre un tema.

[4] Wikipedia, enciclopedia libre www.wikipedia.org [consultado 8/06/2008]

Wikipedia permite la edición de artículos por parte de cualquier usuario, independientemente que éste esté o no registrado en el sistema aunque permite el registro y acceso de usuarios identificados para controlar mejor los cambios en los documentos.

La Wikipedia posee un buscador que muestra la relevancia de los resultados obtenidos, si no determina cual es el artículo que se estaba buscando, además de permitir buscar en otros buscadores, como Google o Yahoo, si los resultados obtenidos no han sido satisfactorios. A pesar de todo, la calidad de los resultados del buscador no es muy buena. Por ello se ha intentado crear un nuevo buscador denominado WikiSeek[5] que es externo al sitio y que utiliza al buscador de la propia Wikipedia y además permite refinar la búsqueda realizando una interacción con el usuario mediante categorías semánticas. WikiSeek se creó en 2007 y sólo funciona en la versión inglesa de la Wikipedia, aunque a fecha actual no se sabe de nuevas revisiones de este buscador y su enlace no está activo. Existe un proyecto de la propia fundación Wikia (Creadora de Wikipedia) para construir un buscador denominado *Wikiasari* que se basa en la relevancia que los usuarios de forma colaborativa den a las búsquedas.

2.1.4 DokuWiki

El software wiki sobre el que se ha desarrollado el wiki objeto de búsqueda es DokuWiki[6]. Este programa facilita la creación de sitios wiki y está desarrollado íntegramente en PHP. Dokuwiki está especialmente ideado para la generación de documentación en línea de forma colaborativa. Su sintaxis es simple, por lo que es muy sencillo leer las páginas fuentes sin utilizar el wiki y por lo tanto permite un fácil tratamiento del texto de los mismos de forma automática. Los ficheros de esta wiki se almacenan en el sistema de ficheros local y no precisa, por tanto, de una base de datos. En la figura 1 se muestra la apariencia de la página inicial de DokuWiki.

[5] Wikiseek, el buscador de la wikipedia http://www.wikiseek.com/ [consultado 30 enero 2008] enlace alterntativo: http://www.genbeta.com/web/wikiseek-el-buscador-de-la-wikipedia.
[6] Dokuwiki homepage http://wiki.splitbrain.org/wiki:dokuwiki [consultado 8/06/2008]

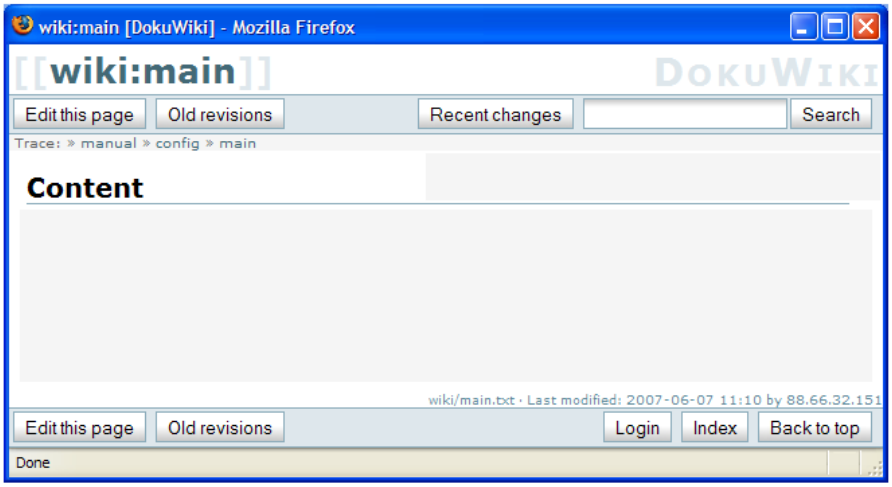

Figura: 1 Página de inicio de DokuWiki

En la Figura: 1 se puede apreciar una de las características de DokuWiki, que es la posibilidad de definir espacio de nombres o *namespaces* para organizar los documentos.

Como la mayoría de las wikis, cuando un usuario está editando una página, ninguno otro usuario puede editarla. Las últimas versiones poseen un instalador Web que facilita su configuración lo que la hace muy fácil de usar. Uno de los problemas de DokuWiki es que al no disponer de una base de datos, el sistema no puede hacer referencias automáticas entre páginas, como en el caso de MediaWiki. Otra característica de esta herramienta es que su configuración es muy flexible y existen multitud de extensiones y temas para personalizar su vista.

La sintaxis que se emplea para editar las páginas de esta wiki es la siguiente:

- El texto en negrita **texto**
- El texto en itálica //texto//
- El texto subrayado __texto__
- Texto monoespaciado ''texto''
- Subíndices _{texto}
- Superíndices ^{texto}
- Tachado ~~texto~~
- Cambio de línea \\
- Enlaces internos [[doku>wiki:nombre_pagina]] o bien [[doku>wiki:nombre_pagina|texto que se muestra]]. No se permiten caracteres especiales en el nombre y se almacena en minúscula. Los dos puntos se refieren al espacio de nombres. Si una página no está editada aparece el enlace en rojo.
- Se pueden enlazar a secciones de una página usando la almohadilla [[nombre_pagina#enlace|texto].
- Enlaces externos a el wiki, es decir a otras páginas de Internet [[http://url|texto]]

- Las direcciones de correo se editan así <usuario@servidor>
- Enlaces Interwiki a otras wiki de esta forma: [[descriptor_wiki>pagina]] por ejemplo el descriptor de Wikipedia es wp.
- Las notas al pie se escriben ((nota al pie)).
- Los encabezados se describen con = = titulo = =. Este sería el encabezado de nivel 5, par un encabezado de mayor tamaño poner más caracteres =. Si hay más de 3 encabezados se generará una tabla de contenidos. Para desactivar esta opción se debe poner al principio del documento ~~NOTOC~~.
- Las imágenes se añaden con {{namespace:nombre_imagen?altoxancho}} sin nada se obtiene el tamaño original. El alineamiento se realiza con espacios entre la descripción y el corchete.
- Para generar listas *nombre
- Para que sean ordenadas –nombre
- Soporta emoticonos y entidades tipográficas.
- Cita de respuesta con >texto >>texto.
- Gestión de tablas[7]:

```
^ Cabecera 1      ^ Cabecera 2       ^ Cabecera 3          ^
| Fila 1 Col 1    | Fila 1 Col 2     | Fila 1 Col 3        |
| Fila 2 Col 1    | unir filas (notar el doble pipe)      ||
| Fila 3 Col 1    | Fila 2 Col 2     | Fila 2 Col 3        |
```

- Con las etiquetas <code> </code> o <file> </file> se inserta código no interpretado.
- Resalto del código con <code lenguaje> Estos son los lenguajes permitidos *actionscript, ada, apache, asm, asp, bash, caddcl, cadlisp, c, c_mac, cpp, csharp, css, delphi, html4strict, java, javascript, lisp, lua, mpasm, nsis, objc, oobas, oracle8, pascal, perl, php-brief, php, python, qbasic, smarty, sql, vb, vbnet, visualfoxpro, xml*
- Código html y php embebido usando <html> </html> <php> </php>

Esta es la sintaxis básica, pero ésta puede ampliarse usando extensiones como por ejemplo extensiones para Latex que permiten editar ecuaciones y textos científicos fácilmente.

2.1.5 Ventajas de los entornos wiki

A modo de resumen, podemos enumerar las ventajas que aportan los entornos wiki en los siguientes puntos:

- La principal ventaja que tienen estos sistemas es que permite crear y modificar documentos en línea de forma rápida y flexible.
- Da el control de la creación de los contenidos a los usuarios, y por tanto los hace partícipes del proceso creativo, lo que permite que los usuarios se impliquen mucho más en el sistema.

[7] Sintaxis de DokuWiki http://www.planetacodigo.com/wiki/wiki:syntax#tablas [consultado 8/06/2008]

- Esta facilidad de edición permite que los usuarios del sitio contribuyan mucha más a su mejora y mantenimiento, haciendo que los sitios crezcan rápidamente porque aportan un beneficio común a todos sus usuarios.
- Todo lo anterior permite una forma muy natural de realizar co-autoría de documentos.
- Este sistema es muy útil en entornos empresariales dinámicos, como la I+D+I. La facilidad de la edición puede mantener a todo el entorno de investigación actualizado las novedades que cada grupo ha descubierto, por tanto también es muy utilizado en entornos académicos.
- Puede crear controversia entre los editores, lo que permite, siguiendo la página de discusiones que suelen tener todas las wikis, seguir el desarrollo argumental de cada una de las posturas y tener en una misma página las distintas opiniones que se tiene sobre un tema particular.

2.1.6 Principales limitaciones de los sistemas wiki

A pesar de que la tecnología wiki posee muchas ventajas, arrastra serios problemas. A continuación se detallan algunos de ellos:

- Dada su gran libertad, son sistemas con fallos de seguridad. Cualquiera puede editar una página de forma malintencionada y modificar su contenido. Se han intentado varias soluciones pero que no solventan por completo el problema. Una de las soluciones más extendida es añadir una identificación al creador de una modificación. Esta puede ser bien el nombre de usuario por estar registrado en dicha wiki o por la dirección IP de la máquina que lo creo. Tanto una como otra forma de identificación falla, ya que las direcciones IP actuales suelen ser asignadas a los clientes de forma dinámica a la vez que se pueden falsear realizando *IP-spoofing*, por lo que estos dos métodos no aportan una solución definitiva este problema. El registro de usuarios, siempre que sea libre tampoco es solución, los usuarios pueden registrarse con diferente nombre si son excluidos de la comunidad por realizar actos indebidos en el wiki.
- Otro problema es la posible inexactitud de los datos publicados por los usuarios, ya que en estos sistemas no suele ser habitual que los artículos se validen antes de ser publicados.
- La navegación hipertextual es adecuada para entender todos los conceptos que se muestran en la página pero desvía mucho la atención del artículo inicial. En ocasiones una navegación basada en enlaces dentro de la página resulta muy incómoda para el usuario y algunos de ellos suelen perderse. De ahí la necesidad de disponer de un buscador ágil y que permita obtener la información que se está buscando de forma fácil y cómoda.
- Los buscadores hasta ahora desarrollados para estas Wikis no ofrecen resultados muy prometedores, simplemente indexan el contenido y

muestran los resultados con un grado de relevancia, pero básicamente usan el titulo de la página wiki para recuperar las búsquedas.

Este último inconveniente es el que se pretende resolver con este proyecto. Actualmente, los sistemas de búsqueda en wikis no permiten búsquedas avanzadas. Por ejemplo si se busca zarzuela en la Wikipedia te lleva directamente a un artículo sobre el género lírico, pero no da la opción a lo que realmente estábamos buscando, el Palacio de la Zarzuela. Lógicamente si se busca como Palacio Zarzuela aparece con relevancia del 100% el artículo que habla sobre dicho palacio. Si se busca Madrid, en la Wikipedia, se muestra la ciudad y al principio un enlace advirtiendo que es un término ambiguo que debe ser desambiguado con una nueva página con enlaces a todas las posibles acepciones que la Wikipedia contiene de dicha palabra.

Este proyecto pretende construir un buscador para un wiki que solvente este problema, realizando una búsqueda que muestre un árbol de navegación de términos relacionados con los términos de la búsqueda de forma que el usuario pueda en unos pocos clicks de ratón, encontrar todo aquello que quiere buscar sin necesidad de replantear los términos de la búsqueda.

2.2 Buscadores

2.2.1 Introducción

Los buscadores son agentes software que se encargan de encontrar documentos de un repositorio que satisfagan los criterios de búsqueda establecidos por el usuario. Los buscadores pueden realizar búsquedas en un sistema de ficheros local como el buscador Beagle, puede buscar archivos sobre una red corporativa o sobre toda Internet como Google.

En todos ellos, el buscador suele crear un índice con los términos que aparecen en todos los documentos del repositorio donde se busca, de forma que cada palabra disponga de los enlaces a los documentos donde esa palabra aparece.

En general, un buscador pretende encontrar documentos a partir de un grupo de palabras que lo describan. Según el buscador, puede sacar el documento que considere más relevante, o una lista con los documentos más pertinentes, ordenados por un valor de **relevancia**. Por ejemplo Google utiliza el **page rank** como medida de relevancia de una Web, que es un algoritmo tiene en cuenta multitud de factores como: la repetición de las palabras en el documento, los enlaces que apuntan al documento, las etiquetas en las que están enmarcadas las palabras etc.

La generación del índice puede hacerse de forma manual o de forma automática. Si el índice se genera de forma manual al buscador se le denomina **directorio de búsqueda**. Estos directorios tienen menos páginas que los buscadores automáticos, pero al estar mantenidos por personas, las páginas que almacena son de calidad y los documentos relevantes suelen estar mejor ordenados.

Si el buscador genera el índice de forma automática, se le denomina **motor de búsqueda.** Estos buscadores deben resolver ciertos problemas para construir de forma eficiente sus índices:

- Se debe procesar los tokens o palabras útiles de los documentos. Existen problemas como las palabras compuestas, las siglas, los acrónimos etc. En idiomas como el castellano o el francés, existen palabras con un mismo lexema, pero con distintas desinencias como los verbos. Semánticamente los verbos independientemente de su tiempo verbal tienen el mismo significado, pero si comparamos dichas palabras por sus caracteres, el motor de búsqueda determinará que ambas son distintas. Para solucionar estos problemas se puede normalizar las palabras. Al proceso de normalización se le suele llamar **lematizar** y consiste en obtener la parte significativa de una palabra, es decir el lexema. Para esto se puede utilizar un diccionario, algunas reglas dependientes del idioma en cuestión (como las conjugaciones de los verbos) o algunos métodos más simples como eliminar el final de una palabra para eliminar plurales. El principal problema de estos métodos es que al realizar este proceso pueden confundirse palabras que tienen una grafía similar, pero que son distintas. Sin embargo si la proporción de errores no es muy grande, es compensada con la mejora en el proceso de búsqueda, siempre y cuando el número de documentos sea elevado, como en el caso e Internet.
- Se puede realizar un análisis morfológico para clasificar las palabras por categorías gramaticales (nombres, verbos, etc.). El análisis morfológico requiere un repositorio con el grupo morfológico de las palabras del idioma a tratar. Esta clasificación resulta compleja, ya que existen algunos problemas pendientes de resolver como, por ejemplo, la polisemia en algunas palabras en la que su significado depende del contexto donde se encuentren estas palabras o cómo clasificar los nombres propios. **Eagles**[8] es un sistema de etiquetado morfológico de palabras que se puede utilizar para este fin. Utilizando el etiquetado de Eagles, es posible construir un lematizador.
- Otro paso para construir el índice es determinar aquellos términos con mayor carga semántica. Esto hace que los resultados de la búsqueda sean normalmente mejores, aunque se pueden eliminar términos útiles pensando que no lo son. A este técnica se le conoce con el nombre de

[8] Eagles online http://www.ilc.cnr.it/EAGLES96/home.html[consultado 8/06/2008]

filtrado. El filtrado permite además reducir el tamaño de la base de datos con lo que se gana en eficiencia.

- Se denominan **palabras vacías** al conjunto de palabras de un lenguaje que no aportan un gran valor discriminatorio para la búsqueda. Por ejemplo palabras como: de, para, en, a, etc., pueden no aportar semántica a la búsqueda. Se pueden utilizar listas de palabras vacías para reducir el tamaño de la base de datos y no desvirtuar ciertas búsquedas con este tipo de palabras.
- El buscador debe asignar pesos a los documentos de forma que dada una palabra se asigne un peso a aquellos documentos donde estén estas palabras, de forma que tenga un peso proporcional a la relevancia del texto sobre ese documento. Por ejemplo, si se busca Madrid, el documento que tenga más peso será aquel que hable o bien de la ciudad de Madrid o bien de la comunidad de Madrid antes que un texto que nombre simplemente a Madrid, pero que no hable de ella tan exhaustivamente. Este es un proceso complicado, existen técnicas como TF (Term frecuency) - IDF (Inverse Document Frecuency) que pueden dar una relación entre la frecuencia de una palabra y su frecuencia total en todos los documentos de la colección, para indicar que palabras son más representativas de cada texto y determinar así la importancia que tiene ese documento con respecto al resto, para un conjunto de palabras concreto. Sin embargo, este sistema tiene ciertas limitaciones debido a la polisemia de ciertas palabras, a los problemas con las distintas terminaciones, etc. Por lo tanto se debe utilizar combinado con otros métodos para mejorar su efectividad.

Un buscador puede estar enfocado a la búsqueda por palabras clave, como puede ser Google o un **buscador de pregunta-respuesta** como puede ser Answer [9].

Los sistemas de pregunta-respuesta son sistemas mucho más complejos que los sistemas de búsqueda por palabras clave. Estos sistemas intentan contestar preguntas en lenguaje natural a partir del conocimiento extraído de un repositorio de documentos. La necesidad de recursos semánticos en estos buscadores es mucho mayor ya que tienen que intentar entender la pregunta que está haciendo el usuario, así como la información que está almacenada en los documentos a buscar. En estos sistemas hay que tener cuidado si se utilizan técnicas de filtrado, ya que palabras como Que, Cuando, Donde, etc., dejan de ser palabras vacías para indicar tipos de preguntas, lugares, organizaciones, etc. Si el número o volumen de documentos que compone el repositorio es muy grande, las técnicas a emplear en los sistemas de pregunta-respuesta suelen ser más similares a los motores de búsqueda, pero si hay pocos documentos, entonces debe haberse diseñado un buen motor de procesamiento de lenguaje natural para que el sistema muestre una tasa de aciertos más elevada.

[9] Buscador de pregunta-respuesta http://www.answers.com/[consultado 8/06/2008].

Para la construcción de buscadores, puede ser útil utilizar una **ontología** de términos asociado al mismo[24]. La ontología permite clasificar las palabras en categorías semánticas lo que facilita enormemente el procesamiento de las preguntas y de los documentos que vamos a ofrecer como respuesta. Una de las ontologías más conocidas que se utilizan en este tipo de sistemas es **WordNet**.

WordNet es una gran base de datos lexicográfica en inglés que agrupa las palabras en conjuntos de sinónimos. WordNet se puede utilizar para ayudar a desambiguar términos, a buscar por palabras relacionadas, a buscar temas relacionados, etc. Esta herramienta puede ser utilizada tanto para construir sistemas de pregunta-respuesta como buscadores clásicos, aunque es en los primeros donde se puede sacar más partido al conocimiento almacenado en sus bases de datos. El problema de Wordnet es que sólo está disponible en inglés y por tanto no puede ser aplicado a otros idiomas, al menos directamente.

2.2.1 Buscadores de Internet

Los buscadores de Internet además tienen ciertas características que los diferencian del resto. Internet no es un repositorio accesible como puede ser el disco duro de una máquina, donde un buscador puede tener acceso a todos los documentos del repositorio. En Internet sólo hay dos formas de encontrar los documentos, una de ellas es a través de los enlaces de unos documentos y la otra es dar de alta expresamente una página en el índice del buscador.

Los buscadores de Internet no pueden subsistir simplemente con las páginas que son registradas de forma manual por los usuarios en su base de datos. Por eso precisan de agentes software que exploren la Web a través de los enlaces que se añaden en las páginas para obtener nuevas páginas Web. Al software que se dedica a obtener nuevas páginas se le suele llamar **robots** de búsqueda o **araña**. Estas arañas se mueven por los enlaces de las páginas que están almacenadas en la base de datos para conseguir nuevas páginas o actualizar los cambios en las ya registradas.

Este planteamiento tiene ciertos problemas. Existen ciertas páginas Web que no son accesibles desde otras, ya que no disponen de enlaces externos a dicha página. Por lo tanto, estas páginas o son registradas por los usuarios en los buscadores de forma manual o no podrán ser indizadas por las arañas. A este tipo de Webs se le denominan **Webs ocultas**. La Web oculta o Web invisible tiene una dimensión aún mayor que las páginas indexadas por los buscadores. Estos motores no pueden indizar páginas que estén detrás de formularios Web o páginas generadas de forma dinámica, ya que, aunque existen métodos para realizar dichas consultas de forma automática, este proceso, aparte de suponer un modo ilícito de intromisión en los sistemas, tiene otros problemas:

- Para cada página Web y para cada petición, el paquete que se manda es diferente, por lo que es muy difícil obtener todas las formas posibles de generar una página web de estas características.
- Si dichas páginas están almacenadas detrás de una página con acceso restringido por contraseña, estás páginas nunca serán indizadas por un buscador, ya que debería conocer la contraseña de entrada.

Así pues, existen un gran número de páginas Web y recursos online a los que no se puede acceder de forma automática. Esto supone una limitación para un buscador que nunca encontrará un gran número de páginas que pueden tener información muy relevante. En el caso de los directorios, al ser procesados por personas, tienen menos restricciones para obtener páginas tras formularios, pero tampoco solucionan el problema completamente. Uno de los factores por los que se puede valorar un buscador es por la proporción de la Web que tiene en su índice. Sin embargo, el tamaño del índice no garantiza que un buscador sea mejor que otro, sino que debe ser combinado con la precisión del algoritmo de posicionamiento, ya que, como el número de páginas Web es tan elevado, es importante que las páginas con la información más relevante aparezcan en las primeras posiciones.

De la gran variedad de buscadores, sin duda, el más popular es Google. El algoritmo de posicionamiento de Google valora las apariciones de la palabra en el texto, la posición donde aparece el término de consulta y el orden de los términos encontrados de acuerdo a la posición en que aparecen en la consulta. También da mucha importancia al número de enlaces que apuntan a la página y el texto que acompaña al enlace. Este es el motivo por el que en algunas ocasiones se ha realizado actos de sabotaje a ciertas personas o entidades mediante la técnica del **bombing**. Se han producido históricamente muchos de estos sabotajes, por ejemplo en España durante mucho tiempo si se escribía "ladrones" aparecía en las primeras posiciones la Sociedad General de Autores Españoles (SGAE). Evidentemente en la página de la SGAE no aparece el término "ladrones" en su contenido con la suficiente importancia como para que aparezca entre las primeras por ese término. Esto se ha conseguido desde fuera del sitio, enlazando desde otras páginas a la página de la SGAE con el término "ladrones" en el texto del enlace.

A pesar de estos problemas, Google ocupa un puesto dominante en el sector de los buscadores debido a que se adelantó al resto de buscadores en obtener un algoritmo de búsqueda más eficiente, y un más difícil de manipular que el resto y en cierta forma, independiente de cómo esté escrita la Web, ya que da mucho peso a los enlaces. No usa diccionarios ni herramientas semánticas, al menos que se sepa, simplemente adquiere el conocimiento de la propia Web a través de sus enlaces. Otros de los motivos por los que se hizo muy popular fue debido a su simple interfaz y a su rapidez, sobre todo en los primeros años ya que la banda ancha cuando surgió Google, aún era un sueño para muchos internautas y los buscadores tardaban mucho tiempo en realizar sus consultas.

Otro de los buscadores de mayor éxito es Yahoo, que inicialmente no era un buscador si no un directorio de páginas clasificado por áreas temáticas y mantenido de forma manual por sus operarios. Al principio este sistema funcionó muy bien, pero cuando el número de páginas de Internet aumentó de forma exponencial, el directorio se volvió inmantenible. Así pues Yahoo optó por utilizar un motor de búsqueda. En algunos momentos llegó a utilizar el motor de Google y actualmente utiliza el motor de la empresa **Inktomi**. A pesar de todo, el directorio de Yahoo sigue existiendo y sigue actualizándose ya que las páginas que aparecen en este y otros directorios son páginas de calidad valoradas no por un sistema software sino por personas y esto ofrece ciertas ventajas que siguen siendo útiles hoy en día.

2.2.2 Buscadores de Escritorio

Los buscadores de escritorio examinan los documentos que hay en el sistema de ficheros de la máquina local. Al igual que los buscadores Web, estos buscadores deben generar un índice de los ficheros del disco. Este índice le sirve al programa para realizar búsquedas de forma rápida tanto en el título como en el contenido de los documentos del disco. La principal dificultad de estos buscadores es que deben ser capaces de leer muchos tipos de documentos con distintos formatos de representación y codificación. Últimamente también sucede esto con los buscadores Web, porque ya no sólo indizan documentos HTML sino también pdf, doc, etc., que están incluidos en las páginas Web a través de sus enlaces. Pero el número de formatos se dispara en la búsqueda dentro de los equipos o en una red local ya que cada vez que se instala un programa, potencialmente se puede incluir un nuevo tipo de codificación diferente para sus ficheros. Por el contrario, la forma de mantener actualizado el índice es mucho más sencilla en los buscadores de escritorio que en sus parientes de la Web, ya que se dispone de todos los ficheros, salvo problemas con los permisos de usuario, y puede conocerse aquellos que han sido modificados recientemente. Estos sistemas permiten mantener desorganizados los archivos en el disco duro y poder recuperarlos de forma fácil con sólo dar algunas indicaciones de que contiene o cual es su nombre. Dos de los buscadores para escritorio más utilizados son Google Desktop en Windows y Beagle en sistemas Linux.

Google Desktop busca texto en mensajes de correo electrónico, archivos, música, chats, páginas Web visitadas. Además permite buscar archivos borrados ya que hace copias de seguridad de los mismos durante cierto tiempo. Actualmente soporta los siguientes archivos: Excel, Word, PowerPoint, Iexplorer, Firefox, MSN Messenger, PDF, MP3, WMA, Varios formatos de vídeo, Imágenes, Ficheros comprimidos, y algunos otros más.

Beagle es una aplicación escrita en mono que es muy similar a Google Desktop. Es un sistema escrito para Linux y soporta un gran número de

archivos como mensajes de Gaim, historial de Firefox, noticias de Evolution, páginas sindicadas con RSS, archivos de openOffice, y Microsoft Office, páginas Web locales, imágenes, PDF, música, y otros. Su licencia es GNU y es completamente libre. Hay una versión para KDE que se denomina Kerry.

2.2.3 Buscadores para wikis

Los buscadores para los entornos wiki buscan sobre la base de datos o sobre los ficheros donde está almacenado el servidor del wiki. Si el wiki está distribuido, el buscador debe de tener acceso a todos los repositorios donde estén almacenadas las páginas. Por tanto, estos buscadores son, por así decirlos, más parecidos a los buscadores de escritorio que a los buscadores Web ya que su entorno es más limitado.

Según [10] las 15 wikis más visitadas en mayo del 2007 fueron por este orden:

- Wikipedia
- Adobe labs
- WikiAnswers
- TripAdvisor Wiki
- Apache Wiki
- WikiMapia
- Second Live Wiki
- Wikia
- AboutUS
- Debian
- WoWWiki
- Dreamhost
- Wiktionary
- WikiHow
- Ubuntu Wiki.

Se han realizado algunas búsquedas en algunas de estas wikis para comprobar el funcionamiento de sus motores de búsqueda.

- El buscador de Wikipedia ya fue valorado anteriormente. Por resumirlo aquí, muestra los resultados ordenados por relevancia, o si cree estar seguro de cuál es la página que se busca, se muestra directamente.
- Adobe labs: Se buscó por air install (air una aplicación de Adobe Labs), pero no supo obtener resultado y mostró una lista ordenada de posibles resultados en los que el primero fue Flash placer[11] una aplicación completamente distinta.

[10] Las diez wikis más visitadas 2007 http://sicrono.com/2007/05/08/10-wikis-mas-visitadas/ [consultado 8/06/2008]
[11] Adobe labs http://labs.adobe.com/wiki/index.php/Special:Search?search=air+install&go=Go [consultado 8/06/2008]

- WikiAnswers: Obtiene resultados a preguntes realizadas. Algunos resultados son muy buenos, a la pregunta “capital of Spain” contesta Madrid. Pero a la pregunta "where is Valencia" no encuentra respuesta.
- TripAdvisor Wiki: Es un wiki de opiniones sobre hoteles escritas por viajeros que han visitado estos hoteles. El buscador localiza los hoteles por la ciudad y muestra su ubicación en un mapa.
- Apache Wiki: Es el wiki del servidor apache y su forma de búsqueda es similar a la del buscador de Wikipedia. Genera una lista de artículos relacionados si no es capaz de resolver con claridad la búsqueda y si la resuelve, muestra la página que cree que se está buscando.
- WikiMapia: es un wiki con información a cerca de lugares en el mundo. Cuando se pincha sobre uno de estos lugares, muestra información a ceca del mismo como el nombre, fotografías, etc. Se apoya en Google Maps para visualizar las localizaciones y tiene un buscador que busca de forma similar a la Wikipedia, mostrando una lista ordenada según la relevancia del lugar y la distancia desde la que busques.
- WikiHow: es un wiki sobre cómo solucionar problemas cotidianos. El sistema de búsqueda sigue el mismo patrón que el resto de sistemas.

Existen además algunos buscadores para wikis externos a estos entornos. Suelen buscar sobre Wikipedia y utilizan tecnologías similares a los motores de búsqueda pero limitando su actuación a páginas de la Wikipedia. Uno de estos buscadores es SearchPedia[12] un buscador que da muy buenos resultados y los ordena por relevancia. Se buscó el término Madrid que el buscador Wikipedia no supo resolver correctamente y este buscador obtuvo mejores resultados, mostrando todas las acepciones de Madrid en la Wikipedia. A pesar de todo, sigue mostrando una lista de artículos relacionados con la búsqueda y ordenados según su relevancia, no permitiendo buscar por términos relacionados como si realiza este proyecto.

El único que parece utilizar búsquedas más avanzadas es WikiSeek que permite refinar la búsqueda pinchando sobre conceptos relacionados con los que se ha buscado. Este buscador actualmente sólo funciona para la Wikipedia de lengua Inglesa.

Ambos buscadores están en abril del 2009 en proceso de modificación y no funcionan de forma correcta.

2.2.4 Problemas encontrados y posibles soluciones

Todos los buscadores analizados tienen limitadas capacidades de búsqueda. A pesar de que algunos obtienen buenos resultados, no hay ninguno que aporte unos resultados del todo satisfactorios, en lenguaje castellano, que permita

[12]Buscador Wikipedia http://searchpedia.compuglobalhipermega.net/ [consultado 8/06/2008]

redefinir los términos de búsqueda, a la vez que corrija errores ortográficos, permita búsqueda booleana y palabras relacionadas estructuradas en forma de árbol.

En el marco de los buscadores wiki, salvo el buscador WikiSeek, parece que ningún otro buscador permite realizar búsquedas por términos relacionados. El resto se limitan a sacar una lista ordenada por relevancia de las páginas que contienen texto relacionado con la búsqueda. Sin embargo, WikiSeek parece estar más en la línea de este proyecto, es decir, construir un buscador para un wiki en el que no sólo aparezcan los resultados de las páginas ordenados por relevancia, sino que permita cierta interacción con el usuario para realizar búsquedas de temas relacionados, sinónimos, etc., para refinar dicha búsqueda.

El presente proyecto se construye sobre un wiki en lengua castellana y su funcionamiento es similar a WikiSeek aunque con algunas diferencias. El buscador desarrollado muestra un árbol y no sólo una lista de palabras relacionadas. Este árbol se apoyará en un tesauro externo para determinar la relación de los conceptos así como otras mejoras que se detallarán posteriormente.

2.3 Web Semántica

2.3.1 ¿Qué es la Web semántica?

Como se puede apreciar en el apartado anterior, los buscadores actuales tienen ciertos problemas para encontrar la información de forma eficiente debido a que los computadores no pueden comprende en toda su plenitud el lenguaje humano, que es el que se utiliza en la Web. La Web semántica es la alternativa propuesta para construir agentes software que sean capaz de leer las páginas Web y poder interpretarlas de forma automática. Para conseguir esto se precisa añadir metadatos semánticos a la información textual de las páginas.

Los metadatos semánticos podemos definirlos como una descripción de la semántica que contiene la página Web. Estos metadatos deben ser especificados de forma formal para que sean tratables computacionalmente. El precursor de esta idea es el mismo que ideo la Web, Tim Berners Lee, físico del CERN. Desde el principio, Tim Berners Lee quiso incluir esta información en su protocolo de hipertexto[13], pero lamentablemente tuvo problemas con las diferentes patentes y la falta de estandarización. Tim Berners Lee quiso incorporar a su protocolo un sistema para incluir información semántica en los enlaces, ya que actualmente los enlaces dicen muy poco de la relación que existe entre la página Web desde la que parte el enlace y la página Web a la que apunta. Una de las

[13] Tim Berners-Lee: Weaving a Semantic Web
http://www.digitaldivide.net/articles/view.php?ArticleID=20 [consultado 8/06/2008]

ideas que incorpora la Web semántica es precisamente añadir información semántica en dichos enlaces para que así los agentes software puedan interpretar la relación existente entre las páginas, convirtiendo la telaraña Web en una red semántica. La idea de Web Semántica de Berners Lee es que dentro del código se enlace a páginas de metadatos que expliquen semánticamente el contenido de la Web de forma que sea entendible por el computador. Esta es la idea seguida en RSS o RDF y otros mecanismos basados en XML en la que se especifica su estructura y contenido mediante un enlace externo a su esquema.

Inicialmente, Tim Berners Lee ideo la Web como un sistema más fácil para compartir información y la edición colaborativa. Con la llegad de la Web 2.0 y los blog o wikis, por fin la Web se ha convertido en un verdadero sistema de colaboración distribuida que era la idea inicial de su creación [17] .

Otra de las ideas que subyacen de la Web semántica es describir, utilizando algún lenguaje, el contenido de la página, es decir, si la página trata sobre Madrid ciudad o Madrid comunidad autónoma. O lo que es lo mismo, incluir información adicional par desambiguar términos y poder construir herramientas mucho más sencillas que las que están diseñadas ahora para que exploren la Web. Por ejemplo un agente software sobre una Web semántica podrá buscarnos el sitio donde se venda más barato un producto o nos recuperará todas las noticias que se han producido de nuestro equipo de fútbol en la última semana.

El esqueleto de la Web semántica se compondrá de documentos a los cuales asociaremos metadatos que los explique y le aporten semántica. Sin embargo, las verdaderas capacidades de la Web sólo se conseguirán aplicando lógica a los metadatos, ya que permitirá extraer automáticamente conclusiones de la información almacenada y hacer búsquedas de información muy especializadas. ¿Cómo se organizan estos metadatos? Las principales vías de investigación apuntan a la utilización de las **ontologías** como la mejor forma de representación de metadatos para la Web semántica, siendo los recursos de la red instancias de estas ontologías.

Actualmente se ha avanzado mucho en la Web semántica y se han definido muchos mecanismos y estándares para su formalización, pero aún hay ciertos problemas que resolver, sobre todo el de cómo adaptar las millones de páginas Web antiguas a los nuevos estándares y como garantizar que estos estándares serán cumplidos posteriormente.

2.3.2 ¿Qué es una ontología?

Una ontología formula un esquema conceptual dentro de un dominio dado, con la finalidad de facilitar la comunicación y la compartición de la

información entre diferentes sistemas, de forma que dichos sistemas puedan entender dicha información.

Por tanto una ontología no es más que una definición formal de conceptos, o clases de un dominio concreto, sus propiedades o atributos, las restricciones que estos poseen y las relaciones entre dichas clases [18].

Las Ontologías surgieron en el contexto de la inteligencia artificial para describir dominios concretos y permitir reutilizar dichos dominios en otros sistemas. Por lo tanto las ontologías deben ser independientes de la tecnología o de la solución que se plante al problema a resolver para poder, precisamente, reutilizarlas. Inicialmente se definieron las ontologías utilizando lenguajes de representación orientados a Marcos como puede ser **Clips**[14]. Los sistemas de marcos son una forma de representación muy potente de la que surgió posteriormente la programación orientada a objetos como un subconjunto de la misma. Su potencia para representar el conocimiento le ha permitido extender su utilización fuera de los dominios de la inteligencia artificial y se está comenzando a utilizar para modelar las relaciones de la Web semántica.

Las clases se pueden relacionar de forma muy variada, algunas de las relaciones más importantes son:

- Relación clase-subclase para formar jerarquías de clases. Las subclases especifican características más generales que describen su clase de nivel superior. Se suele denominar esta relación como "es un", por ejemplo un *gato "es un" mamífero*. Entre la subclase y la clase se produce un proceso de herencia, es decir, las características descritas en la superclase no son precisas de especificar en la subclase.
- Los atributos son las características que identifican a una clase.
- Una clase puede estar compuesta por una o varias clases. Esta composición hace que una clase disponga como atributo una o varias instancias de una o varias clases distintas que conjuntamente la componen.
- Una instancia es una concreción de una clase que contiene los valores concretos de sus atributos.
- El número de relaciones que componen una relación se denomina cardinalidad.
- Los axiomas describen restricciones que debe cumplir las relaciones entre las clases.

Actualmente se han definido varios lenguajes de descripción de ontologías para la Web. En la sección siguiente se resumirán los más utilizados.

[14] Herramienta de programación orientada a Marcos http://clipsrules.sourceforge.net/ [consultado 8/06/2008]

2.3.3 Estándares para la Web semánticas

La Web semántica se sustenta en una variedad de protocolos de descripción de metadatos. De algunos ya se ha hablado anteriormente pero en este apartado se describirá de forma resumida su utilidad dentro del paradigma de la Web Semántica.

2.3.3.1 XML

Es un estándar de definición de lenguajes de marcado estilo HTML. XML es un subconjunto de SGML, especificación de la que inicialmente se extrajo HTML, pero que debido a su complejidad, es muy costoso de tratar. Por ello XML simplifica SGML para permitir crear lenguajes nuevos de forma más sencilla pero sin perder la potencia de su predecesor. XHTML es un ejemplo de la utilización de XML en la Web. XHTML no es más que una nueva versión de HTML conforme a las especificaciones descritas por XML. Otros lenguajes que siguen las directrices de XML son por ejemplo: SMILE, SOAP, RDF, XAML, etc.

XML se ha utilizado para definir los lenguajes que dan soporte a la Web semántica debido a su popularidad, a la gran cantidad de interpretes y gestores XML ya existentes y a que un lenguaje definido utilizando XML está comprobada su formalidad. Sus principales ventajas son:

- Es extensible, es decir que si en la fase de definición del lenguaje se han cometido errores o las necesidades que pretendía cubrir han cambiado, con XML puede ser fácilmente redefinido añadiendo nuevas etiquetas, y lo que es mejor, manteniendo la operatividad de versiones anteriores.
- El analizador es un componente estándar, no es necesario crear un analizador específico para cada lenguaje, por lo tanto se reducen los errores y se acelera el desarrollo de la aplicación.
- La facilidad para entender el lenguaje mejora la compatibilidad entre aplicaciones.

XML usa **XML Schema** como lenguaje de definición de lenguajes. Usando este lenguaje se especifican de forma muy precisa las restricciones y la estructura de un documento XML. El propio lenguaje de definición XML Schema está definido en XML.

2.3.3.2 OWL

OWL[15] es el lenguaje para descripción de ontologías más potente. OWL es un lenguaje de marcado para la publicación de ontologías Web, construidas sobre RDF y codificado en XML que permite construir ontologías a partir de vocabulario más amplio que RDF. OWL dispone de tres lenguajes OWL-lite, OWL-DL y OWL-Full cada uno de los cuales proporciona un conjunto de posibilidades cada vez más amplias, pero eso si, cada vez más complejas de procesar.

OWL aporta mejoras sobre los otros lenguajes de definición de ontologías, algunas de ellas son las siguientes:

- Las ontologías definidas en OWL pueden ser distribuidas entre varios sistemas.
- Es completamente compatible con los estándares de accesibilidad.
- Es un lenguaje abierto y permite su extensibilidad.
- Existen un gran número de ontologías disponibles en OWL.
- Permite limitar las propiedades de una clase.
- Permite definir propiedades dentro de una clase que no sean comunes a todas las instancias de la clase.
- Permite distinguir relaciones uno-uno, uno-n o n-uno.
- Construcción de clases a partir de uniones, intersecciones y complementos de otras clases.

OWL-Lite permite establecer relaciones jerárquicas entre conceptos. Es el más sencillo y es el que más se suele utilizar. Este lenguaje permite sólo relaciones 0-1. Suele utilizarse para generar **taxonomías** y tesauros.

Algunos conceptos del lenguaje son:

- Class define un grupo de objetos que comparten algunas características.
- SUbClassOf permite organizar las clases jerárquicamente e inferir relaciones de herencia.
- Individuos: ejemplos de clases.
- Las características o propiedades expresan relaciones entre las clases y de las clases a sus valores. (propertys)
- Subcaracterísticas, las características también pueden estar jerárquicamente clasificadas. (subpropertys)
- Dominios (domine): limitaciones de las características a las clases que se puede aplicar. Un dominio reduce el número de individuos que pueden aplicar la propiedad.
- Rango: Limita a los individuos en el valor de sus características.

[15] Descripción de OWL http://www.w3.org/2007/09/OWL-Overview-es.html [consultado 8/06/2008]

Ejemplo de uso:

Manuel(individuo):clase persona.
UniversidadCarlosIII(individuo):Clase universidad
Estudia(caracteristica)
Manuel estudia UniversidadCarlosIII.

- EquivalentClass: Es posible definir dos clases como equivalentes.
- EquivalentProperty: Es posible definir dos propiedades como equivalentes.
- SomeAs: es posible definir dos individuos como similares.
- DifferentFrom: Es posible definir un individuo como diferente de otro.
- InverseOf: una característica contraria a otra característica.
- TransitiveProperty un grupo de características pueden poseer la propiedad transitiva. A Es B y B es C entonces A es B.
- SymmetricProperty: una característica es aplicable en las dos direcciones. A es_amigo B y B es_amigo de A.
- FunctionalProperty: si una caracteristica es declarada como funcional, quiere decir que sólo puede tomar cardinalidad 0 ó 1. Si definimos la característica morir, A sólo podrá morir una vez.
- InverseFunctionalProperty: El opuesto de la propiedad será funcional, por tanto tiene como máximo un valor de cada individuo.
- AllValuesFrom: Esta restricción de características hace que sólo pueda relacionar esta restricción con una instancia de una clase concreta. Por ejemplo si María tiene padre (relación):AllValuesFrom(Hombre) entonces la instancia con la que se relacione tiene_padre debe ser instancia de la clase Hombre.
- SomeValuesFrom: Es muy parecida a la anterior, pero simplemente cambia el hecho de que al menos uno de las instancias de la clase debe ser de la clase especificada, el resto pueden serlo o no.
- Restricciones de cardinalidad:
    - Cardinality: Mínima y máxima conjunta definido sobre las características.
    - MaxCardinality.
    - MinCardinality.

Algunos ejemplos[16]:

```
<owl:Ontology rdf:about="http://www.example.org/wine">
  <rdfs:comment>An example OWL ontology</rdfs:comment>
  <owl:priorVersion rdf:resource="http://www.example.org/wine-112102.owl"/>
  <owl:imports rdf:resource="http://www.example.org/food.owl"/>
  <rdfs:label>Wine Ontology</rdfs:label>
</owl:Ontology>
```

[16] Sintaxis de OWL http://www.w3.org/TR/owl-xmlsyntax/apd-example.html [consultado 8/06/2008]

2.4 Tesauros

El término tesauro proviene del griego y significa *colección* aunque en latín se adoptó con el término *tesoro*. La primera vez que se utilizó la palabra tesauro fue entre el 1262 y el 1268 por **Bruneto Latini** en la Enciclopedia *Livre dou Tresór*.

Un tesauro como lo conocemos actualmente es un lista estructurada de descriptores o términos de un ambiente científico o social entre los cuales se estable una serie de relaciones jerárquicas y asociativas [3]. Los tesauros tienen muchas utilidades pero en el campo de la recuperación de la información se suelen utilizar para aportar mayor semántica a los procesos de búsqueda ya que aporta un conjunto de palabras relacionadas con las palabras a buscar, por lo que ofrecen más posibilidades en la búsqueda. Los términos de los tesauros se relacionan entre ellos de dos formas distintas:

- **Relaciones jerárquicas**: Establecen relaciones del tipo jerárquico. Un término (hiperónimo) se refiere a un concepto más general que otro (hipónimo)
- **Relaciones de equivalencia**: Establece relaciones de **sinonimia**, **homonimia** y **antonimia** entre los términos.

Los tesauros no tienen por qué abarcar todo el conocimiento, sino que generalmente se limitan a un área temática, por lo que existen innumerables tesauros para deferentes dominios. Los tesauros no tienen por qué definir los términos que ellos contienen, aunque en algunas ocasiones posee definiciones cortas, siempre de cara a los usuarios humanos.

La finalidad última de un tesauro es facilitar al usuario, sea o no un software, disponer de todas las palabras que expresan un determinado concepto, así como palabras que estén relacionadas de forma jerárquica con dicho concepto. En la actualidad existen numerosos tesauros en español de dominios muy dispares como por ejemplo:

- Tesauro de carreteras realizado por Concepción Lallana del Valle en 1993
- Tesauro AGROVOC: dedicado a la agricultura.
- Tesauro del CIS (centro de investigaciones sociológicas).
- Etc.

La creación de un tesauro es una tarea muy laboriosa. Por lo tanto es muy recomendable utilizar herramientas para su construcción. Algunas de las herramientas que se pueden utilizar son las siguientes:

- Domain Reuser[17]: que es la herramienta que se utiliza en este proyecto para generar el tesauro.
- Amicus Thesaurus[18].
- Tesaurus Builder[19].

Estas herramientas facilitan la construcción de los tesauros ya que completan las relaciones entre las palabras de forma automática. Por ejemplo, si decimos que animal es hiperónimo de mamífero, automáticamente se crea la relación inversa de mamífero es hipónimo de animal. El buscador desarrollado incorpora un tesauro realizado por Sonia Sánchez Cuadrado como parte de su tesis doctoral [21].

[17] Página Web de Domine Reuse http://www.reusecompany.com/ [consultado 8/06/2008]
[18] Página Web de Amacus http://www.amicuscom.com/ [consultado 8/06/2008]
[19] Página Web de Tesaurus Builder http://www.thesaurusbuilder.com [consultado 8/06/2008]

3. Análisis del sistema

3.1 Motivación del sistema

Como se ha podido apreciar en **el apartado 2**, **Estado del arte**, no existe ningún buscador para el entorno wiki que permita realizar búsquedas avanzadas. El más similar podría ser WikiSeet, buscador para la Wikipedia, pero además de estar sólo disponible para artículos en inglés, tampoco genera resultados utilizando un tesauro de búsqueda organizado de forma jerárquica. Así pues, el objetivo de este proyecto es construir un buscador para un wiki que dispone el departamento de ingeniería del software y que puede ser visitada en la siguiente página web: http://163.117.147.74/ie/doku.php, aunando las técnicas tradicionales de búsqueda y apoyándonos en un tesauro para mejorar su funcionalidad y aportar mayor utilidad a los usuarios.

Esta wiki es un recopilatorio de información sobre temas relacionados con la Web, las ontologías, la Web semántica, los sistemas de recuperación de información y sistemas de reutilización. El sitio ha sido construido con DokuWiki, una herramienta Wiki de libre distribución que no se apoya en ninguna base de datos, es decir los documentos se almacenan en forma de ficheros. Además, incorpora un buscador similar al buscador que utiliza la Wikipedia, pero tiene muchas carencias. Por ejemplo, no admite palabras sin acentuar en la búsqueda, es decir si se busca “ingenieria informacion” sin acentos no recupera ninguna página y la misma búsqueda con acentos produce 30 resultados.

Ante la búsqueda del término XML los resultados que aparecen se pueden observar en la Figura: 2 como los resultados no son nada prometedores.

Figura: 2 Búsqueda de 'XML' en el buscador de DokuWiki

3.2 Características del sistema

El proyecto pretende dotar al buscador del wiki de mayor capacidad de búsqueda ya que el buscador desarrollado no solamente muestra una lista ordenada por los aciertos obtenidos como es el caso del buscador de esta wiki, sino que además permite consultar un tesauro que ayuda en la búsqueda de forma que el usuario pueda navegar por alguna de sus entradas y visitar las páginas relacionadas con los conceptos que en la búsqueda se plantea. El tesauro utilizado es específico sobre la materia de la que trata el wiki, es decir, sobre temas relacionados con la recuperación de información, la Web semántica y la Ingeniería del Software. Además, permite búsquedas booleanas con los operadores AND, OR, NOT para aumentar la potencia de búsqueda. La conjunción AND es implícita, de forma que al añadir palabras a los resultados obtenidos deben contener las dos palabras. Para búsquedas alternativas se utiliza el operador OR (|) y para indicar que cierta palabra no aparezca en la búsqueda se utilizará el operador NOT (-).

Asimismo, el buscador incorpora un sistema de corrección de errores ortográficos en el caso de que el usuario escriba una palabra que el buscador no tiene en el índice. Si esto sucede, el sistema sugiere una lista de palabras similares a la que el usuario ha introducido.

Como funcionalidad extra, el sistema permite realizar las búsquedas sobre el tesauro que incorpora el wiki así como términos en la Wikipedia. Para esta última funcionalidad puede utilizarse el propio indizador para construir el índice de la Wikipedia, (proceso que en las pruebas se ha visto factible) o bien utilizar la opción por defecto que es consultando a Google a través de nuestro buscador.

Por último, la búsqueda avanzada permite configurar el grado de exhaustividad del buscador así como la aparición o no de las páginas de revisión en la búsqueda. Para minimizar tiempos de proceso puede buscar con un grado de exhaustividad bajo, en la cual se tendrán en cuenta menos documentos por palabra. Si se quiere aplicar la capacidad del buscador, ampliando el tiempo de búsqueda, se puede incrementar el nivel de exhaustividad.

3.3 Tecnologías aplicables a su desarrollo

Para el desarrollo del proyecto se ha realizado un estudio sobre las diferentes tecnologías que se pueden aplicar al mismo. Por un lado, disponemos de algunos recursos que nos es útil reutilizar. La herramienta utilizada para la construcción del wiki es DokuWiki. Aunque se ha diseñado el programa de forma que sea fácilmente exportable a otros entornos, esta elección determina en gran medida las posibles elecciones posteriores. En primer lugar, no se

necesita, por tanto, disponer de una base de datos donde almacenar las páginas y en segundo lugar se precisa de un intérprete de la sintaxis de DokuWiki si se pretende indexar directamente los ficheros.

Lo mismo sucede con el tesauro que se utilizará para consultar los términos relacionados con la búsqueda, que se genera utilizando una base de datos Access o bien un fichero de descripción en texto plano. En la versión final se utiliza la versión en texto plano del tesauro para optimizar el tiempo de consulta.

Por lo tanto, las decisiones a la hora de construir el sistema se basan en la tecnología que se va a utilizar para generar la página Web del buscador y la tecnología con la que se va a utilizar para desarrollar el propio buscador. A continuación se mostrará algunas de las posibilidades tanto para el desarrollo de la interfaz como para el desarrollo del buscador seleccionando sus ventajas y sus inconvenientes.

3.3.1 Tecnologías para la creación de la interfaz Web

Como la página Web de consulta del buscador debe ser generada de forma dinámica, hay que utilizar alguna herramienta de generación de páginas dinámicas. Existen varias posibilidades que se pueden agrupar en tecnologías que se ejecutan en el servidor y tecnologías que se ejecutan en el navegador o cliente Web. En los siguientes apartados estudiaremos las ventajas y desventajas de cada una de las tecnologías y su adaptación al entorno al que vamos a aplicar.

3.3.1.1Tecnologias del lado del cliente

Esta forma de construir páginas Web se basan en el aprovechamiento de los recursos de las máquinas cliente que se conectan al servidor para liberar carga de trabajo a los servidores. Además la respuesta de la interfaz es mucho más rápida debido principalmente a que no se realizan conexiones con el servidor, lo que mejora enormemente la experiencia el usuario. Las principales tecnologías del lado del cliente son tres, JavaScript, Visual Basic Script y Applet de Java.

VISUAL BASIC SCRIPT

Visual Basic Script o VBScript es un lenguaje interpretado de sintaxis similar a Visual Basic. Actualmente este lenguaje no es muy utilizado en la práctica siendo mucho más utilizado JavaScript. El problema de este lenguaje es su mala imagen y sus fallos continuados de seguridad.

JAVASCRIPT

JavaScript es el lenguaje interpretado más usado para la programación del lado del cliente. Es completamente libre y fácil de editar. Su sintaxis es similar a C++ o Java aunque no es un lenguaje orientado a objetos ya que no dispone de herencia. El lenguaje lo creo Brendan Eich y se utilizó por primera ven con el navegador NetScape en su versión 2.0. Para incluir código JavaScript se debe añadir en la página Web la etiqueta

```
<script type="text/javascript">
<script>
```

Su principal ventaja es que es un estándar aceptado prácticamente por la totalidad de los navegadores actuales aunque dependiendo de la versión del navegador, ciertas instrucciones no se ejecutan de la misma manera a pesar de su estandarización. Otra de sus problemas están relacionados con accesibilidad aunque son solucionables en parte mediante la etiqueta noscript.

Actualmente se ha popularizado la utilización de Javascript ya que se utiliza como parte de AJAX (Asincronus JavaScript and XML), una nueva forma de programar interfaces Web con comunicación asíncrona con el servidor que permite una mejor interacción con los usuarios y permite convertir a una página Web en una aplicación Web muy similar a la que podemos encontrar en el escritorio de un PC.

APPLETS

Los Applets son clases de Java que se descargan junto con la página Web y que se ejecutan en el navegador, siempre que este disponga de un entorno de ejecución para Java incorporado en el navegador, que básicamente es una máquina virtual adaptada para él. Esto permite generar programas en código Java, un lenguaje de programación de carácter general, para mostrar en el navegador como si fuera un programa más. ¿Cuál es la ventaja de esta tecnología? Pues que Java es un lenguaje de programación muy potente y permite construir grandes aplicaciones que se ejecutan en la máquina del cliente. Otra ventaja sobre los lenguajes de guiones como JavaScript es que es un lenguaje compilado y por tanto, inicialmente no su código no es accesible a los clientes, salvo utilizando decompiladores, algo que lenguajes como JavaScript, al ser guiones interpretados, deben de ir en texto sin compilar. Además es muy sencillo utilizar un applet fuera de un navegador Web, por lo que el esfuerzo de construir un applet Web y a la vez un programa para ejecutar en local es práctica nulo. Como todo tiene sus inconvenientes, un applet consume muchos recursos en la máquina del cliente y aumenta notablemente el tiempo de carga de la página Web.

3.3.1.2Tecnologias del lado del servidor

Estas tecnologías se caracterizan por que su ejecución la realiza el propio servidor Web a partir de extensiones que se añaden al mismo. Existen varias tecnologías que permiten construir páginas dinámicas que se ejecutan en el servidor. La tecnología clásica es CGI (Common Gateway Interfaz) que es un protocolo que proporciona una interfaz entre el servidor y un proceso externo de forma que el servidor ejecuta un programa externo que genera una página o fragmento de página que es enviado de nuevo al servidor para que este lo muestre en pantalla. Actualmente existen otras tecnologías más potentes para la construcción de páginas Web dinámicas entre las que se incluyen PHP, ASP.NET o JSP.

PHP

PHP (PHP Hypertext Pre-processor) es un lenguaje interpretado que se utiliza en la creación de páginas Web. PHP es una extensión del servidor Web que se instala como un CGI que es capaz de pre-procesar los comandos PHP que se incluyen en las páginas para mostrar una página final en HTML. Esta solución tiene como ventaja su fácil adaptación al entorno de implantación ya que la herramienta de construcción del wiki (DokuWiki) está escrita en PHP por lo que sería fácilmente integrable en el entorno actual.

Además PHP se integra perfectamente en multitud de sistemas gestores de base de datos. Hay que tener en cuenta que la utilización de este tipo de herramientas puede ocasionar graves agujeros de seguridad si no se toman las medidas oportunas, como es la instalación de Plugins, o actualizaciones.

PHP es interpretado y ejecutado en un servidor Web compatible, (que son la mayoría) permite acceder a bases de datos y crear páginas de forma dinámica en el servidor. El cliente sólo recibe el resultado de la ejecución, por lo tanto oculta el código a los mismos. Cuando el cliente hace una petición al servidor para que le envíe una página Web, el servidor ejecuta el intérprete de PHP, el cual, procesa el script solicitado y genera el contenido de manera dinámica. Además, es posible utilizar PHP para generar archivos PDF, Flash, así como imágenes en diferentes formatos, entre otras cosas.

Los lenguajes de Scripting como PHP suelen ser más sencillos de utilizar y de aprender para programadores poco expertos que lenguajes de propósito general como Java o .NET. Por el contrario, la potencia de los mismos es menor y se vuelven difíciles de mantener cuando el sistema es muy complejo.

ASP.NET

Otra alternativa de desarrollo del interfaz está basada en la tecnología de Microsoft ASP.NET. Está tecnología está integrada en la plataforma .NET y se caracteriza por unificar el entorno de desarrollo con el compilador utilizando la

herramienta Visual Studio.Net. Esta herramienta permite la construcción de sistemas en distintos lenguajes de programación debido a que el código se precompila a CIL (Common Intermediate Language) y no directamente a código máquina. Este código se compila al vuelo al ser ejecutado en la máquina virtual la primera vez, de forma que se puede obtener sistemas multiplataforma tanto hardware como de sistema operativo y de código de implementación, sin consumir los recursos de una máquina virtual como la de Java. El problema de .NET es que es un sistema propietario y de momento sólo está disponible para Windows. Aunque existen implementaciones para Linux basadas en software libre (como Mono) estás no son tan completas como el entorno de Microsoft. ASP.NET es una parte de Visual Studio dedicada al desarrollo de sitios Web que está caracterizado por contar con una interfaz Web muy rica y útil, proporcionando generación de páginas HTML y XHTML dinámicas. ASP.NET es perfectamente compatible con la mayoría de gestores de bases de datos, no así con los servidores Web. Los únicos servidores Web que pueden ejecutar código ASP son los servidores de Microsoft. El más conocido es Microsoft Internet Information Services (IIS). A parte de este inconveniente, debemos denotar que ASP es un lenguaje de Scripting, no un lenguaje de programación de propósito general como pueda ser Java, lo que le hace fácil de utilizar pero limitado de potencia. Otra característica a reseñar es la enorme facilidad que ofrece la herramienta para crear interfaces gráficas muy vistosas y potentes. Como la plataforma a la que da soporte es muy rígida y el buscador se pretende que pueda ser integrable fácilmente dentro del wiki, este aspecto nos resulta incómodo, ya que el wiki está escrita en PHP sobre un servidor Web Apache y por tanto puede haber problemas para acoplar estas dos plataformas. Acometer esta costosa tarea puede no compensar las posibles ventajas de utilizar esta tecnología. Además, al ser una tecnología propietaria, el coste del proyecto se incrementaría sustancialmente.

JSP

JSP (Java Server Pages) es una tecnología que permite generar contenido dinámico para la Web. JSP permite utilizar código Java mediante scripts y dispone de varias etiquetas propias que pueden ser ampliadas utilizando librerías de etiquetas. JSP no es un script, es una tecnología compilada aunque transparente para el programador. La principal ventaja de **JSP** frente a otros lenguajes es que permite integrarse con clases Java (.class) lo que permite separar en niveles las aplicaciones Web, almacenando en clases Java las partes que consumen más recursos (así como las que requieren más seguridad) y dejando la parte encargada de formatear el documento HTML en el archivo JSP.

La idea fundamental detrás de esta forma de programar es el de separar la lógica del negocio de la lógica de presentación de la información. Al utilizar Java, puede ejecutarse en cualquier sistema operativo o arquitectura de máquina y además es gratuito, por lo que su incorporación no supone ningún coste adicional al proyecto. Esta tecnología tiene como principal desventaja que precisa de un servidor Web que soporte Java, como puede ser Apache Tomcat y

además que su rendimiento, al ser ejecutado dentro de la máquina virtual de Java, es menor que otras soluciones.

3.3.2 Tecnologías para la creación del buscador

Para la construcción del buscador podemos utilizar muchos lenguajes de programación. Se ha optado por el estudio entre C++, Java y la plataforma .NET, por ser los lenguajes sobre los que se tiene mayor experiencia. Los dos primeros son lenguajes orientados a objetos y de amplia difusión y se dispone de soluciones de software libre para su utilización como GCC y J2SD. El entorno .NET ofrece grandes ventajas de integración en sistemas operativos Windows. Las características de cada uno de ellos son las siguientes:

3.2.2.1 C++

C++ es la extensión orientada a objetos del lenguaje C. Fue diseñado a mediados de 80 por Bjarne Stroustrup. A pesar de ser orientado a objetos, soporta programación estructurada. Sus principales ventajas son su gran rendimiento en comparación con otras tecnologías como Java o .Net, su enorme versatilidad, su capacidad para producir software genérico y además, posee algunas características que no se suelen encontrar en otros lenguajes de programación, como:

- Permite redefinir los operadores.
- Permite identificar tipos en tiempo de ejecución.
- Casi todos los compiladores poseen potentes preprocesadores que permiten construir un código adaptado a las diferentes arquitecturas.
- Posee un potente manejo de bits y de conceptos de bajo nivel como los punteros.
- Permite incrustar código ensamblador.
- Permite definir punteros a funciones.

Sin embargo dispone de ciertas limitaciones que repercuten, sobre todo en el tiempo de desarrollo de los proyectos.

- Su enorme flexibilidad le hace propenso a errores.
- Si no se programa con cuidado, le código puede ser muy complejo de entender por otros programadores.
- Las herramientas de libre distribución no suelen disponer de muchas librerías de carácter general, aunque siempre estarán disponibles si se navega por Internet al ser un lenguaje de programación muy utilizado.
- La gestión de errores por parte del compilador es mucho más costosa que en entornos más modernos.

- La gestión de la memoria debe hacerse de forma manual asignando y liberando memoria de forma explícita.

3.3.2.2 Java

Java es un lenguaje orientado a objetos desarrollado por Sun Microsystems en la década de los 90. Su sintaxis es muy similar a C++ pero tiene un modelo de objetos mucho más simple y por lo tanto reduce mucho las posibilidades del programador con respecto a este último. Esto no significa que sea un lenguaje de programación peor que C++, simplemente su propósito es distinto. Java elimina gran parte de la flexibilidad de C++, para intentar reducir los errores de programación que se pueden cometer con el mismo. Elimina muchas herramientas de bajo nivel como los punteros y simplifica la gestión de memoria de forma que se libere de forma automática cuando no se necesite. Esto lo realiza a cambio de un mayor tiempo de ejecución de sus programas y de que estos ocupen mucha más memoria. Java es un lenguaje en parte compilado, en parte interpretado por una máquina virtual. El ejecutable de Java no se ejecuta directamente por el computador sino por un programa que lo interpreta. Esto también genera un incremento de recursos.

Java liberó su código bajo licencia GNU GLP en 2007. Esto le hace en parte un software ser software libre. Sus principales ventajas por tanto son las siguientes:

- Simplicidad a la hora de construir las aplicaciones, que permite una reducción en el tiempo de desarrollo de las mismas.
- Una enorme biblioteca de Objetos que se pueden utilizar para el desarrollo de programas.
- Independencia de la plataforma donde se vaya a ejecutar.
- Es posible disponer de herramientas de desarrollo de gran potencia y gratuitas como puede ser eclipse.

Sus principales inconvenientes son:

- A pesar de que gran parte del código de Java es de libre distribución, algunas partes de este pueden ser utilizadas de forma gratuita, pero no son software completamente libre.
- Aporta una importante sobrecarga en la ejecución de programas y un aumento considerable del consumo de recursos de la máquina.
- A pesar de su independencia de plataforma, tiene algunos problemas con el entorno gráfico en ciertos sistemas operativos y en aplicaciones sobre dispositivos de poca potencia de cálculo.

3.3.2.3: .NET

.NET es un proyecto de Microsoft para crear una plataforma de desarrollo de software transparente tanto para la comunicación, como en el lenguaje en el que esté desarrollado, con independencia de plataforma hardware y que permita un rápido desarrollo de las aplicaciones, aportando gran cantidad de librerías. Como se esbozó antes cuando hablamos de ASP.NET, la plataforma .NET tiene múltiples ventajas pero su principal inconveniente es que hay malas implementaciones en entornos que no sean de Windows, así como su coste al ser una herramienta propietaria.

3.4 Tecnología seleccionada.

En la Figura: 3 podemos apreciar el esquema arquitectónico general del sistema.

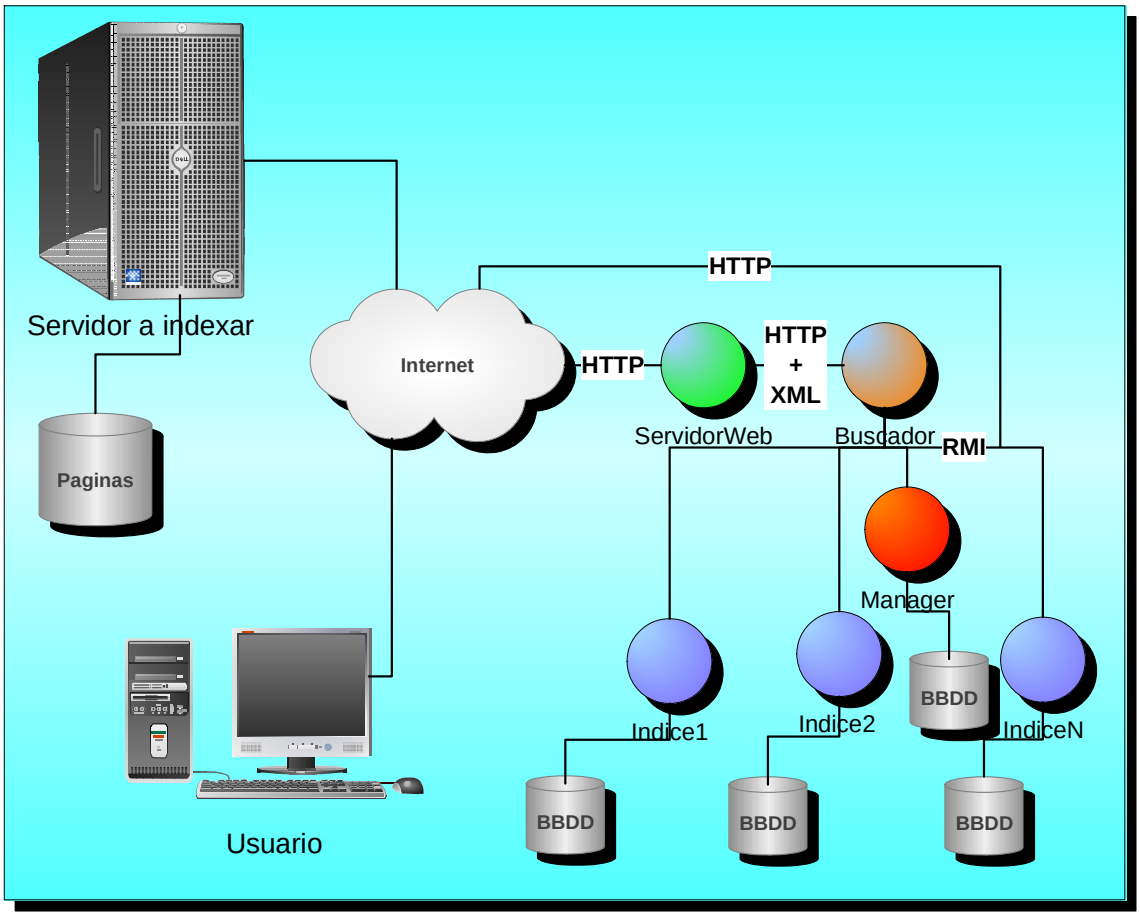

Figura: 3 Diseño arquitectónico del buscador

Para el desarrollo del buscador se ha seleccionado Java como la tecnología que se utilizará en el buscador. En cuanto a la construcción de la interfaz se ha seleccionado como tecnología principal PHP, aunque se ha utilizado JavaScript y Ajax para ciertos elementos más interactivos.

El servidor donde está instalado el wiki que se indiza puede estar en la misma máquina donde se instale el buscador o no, ya que éste, por defecto, adquirirá la página por HTTP.

La estructura del buscador está completamente distribuida. Existen tres tipos de procesos que se comunican entre sí a través de RMI. Estos procesos son:

- Índice: El índice es un proceso escrito en Java que se encarga de recuperar las páginas del sitio que se le indique en su configuración. El número de índices puede ser variable dependiendo de la dimensiones del sitio a procesar. Cada uno de estos índices puede ejecutarse en una máquina distinta o todos en la misma máquina si es una máquina multiprocesador. Cada uno de estos índices procesará una página del sitio y atenderá a un número de palabras. Estos procesos mantienen actualizado sólo una parte del índice en su base de datos. Más adelante se explicará el funcionamiento de este proceso con mayor detalle.
- Manager: Este proceso coordina la actuación de los índices, almacenando las documentos que los índices van encontrando y asignándoselos a los procesos en **round robin** para su procesamiento.
- Buscador: Este proceso se encarga de analizar las peticiones de búsqueda de los usuarios, así como de valorar los documentos obtenidos para darles una relevancia y procesar el mecanismo de comunicación con la interfaz. Este proceso está diseñado para que si el entorno de desarrollo al que se aplica este sistema cambia (es decir, deja de ser un buscador online para ser uno de escritorio, por ejemplo) o cambie el algoritmo de valoración de los documentos, los cambios sólo se propaguen a este proceso, aislando el resto del sistema de los mismos. Además el proceso se encarga de tratar con el tesauro de términos, consultándolo para obtener palabras relacionadas, sinónimos, etc.

La comunicación entre estas tres entidades se realiza por RMI, aunque, como veremos más adelante, el proceso buscador realmente no sabe cuál es la comunicación real que tiene con los índices. Luego podría cambiarse la implementación de la comunicación por otro paradigma.

Además de estos procesos, se ha dotado al sistema de una interfaz Web que se comunica únicamente con el buscador a través de un protocolo propio descrito en XML y trasmitido por HTTP. Esta interfaz puede ser sustituida por otra siempre y cuando se respete el sistema de comunicación XML establecido. La comunicación con el proceso buscador es por red, lo que permite que la aplicación Web y el buscador puedan estar en máquinas separadas. Esto facilita la escalabilidad global del sistema a la vez que reduce la integración de la interfaz con el resto de la aplicación.

La implementación de los distintos procesos se ha hecho en Java. Los motivos para seleccionar dicha tecnología han sido los siguientes:

- Java es una herramienta de desarrollo gratuita.
- La productividad de Java (en tiempo de desarrollo) es mayor que otros sistemas como C++. Se ha valorado más este hecho que el incremento en el tiempo de proceso que implica desarrollar en Java, debido principalmente a su máquina virtual.
- La potencia de este lenguaje y de sus librerías para la comunicación por red, así como la facilidad de programación que aporta la tecnología RMI.
- Las facilidades que aporta para construir un sistema de módulos basado en conectables o *plugins* que aísla y minimiza los cambios y que facilita la extensibilidad del sistema y lo protege ante cambios de implementación, lo que le convierte en un sistema fácilmente ampliable y modificable en el futuro.
- Su independencia de plataforma.

Para almacenar la base de datos se ha utilizado Mysql como gestor. Esta selección se ha hecho porque es un sistema muy utilizado y bastante robusto, aunque puede ser superado en rendimiento y productividad por otros sistemas, como Oracle, su principal ventaja es que es software libre.

En cuanto a la elección de PHP como tecnología para el desarrollo de la interfaz, se ha seleccionado ésta, porque a demás de ser gratuita, se adapta perfectamente tanto al entorno donde se va a instalar el sistema, (servidor Web Apache) como a las necesidades de comunicación e integración con el resto del sistema. Además, PHP es una tecnología muy utilizada y con muchas herramientas de desarrollo disponibles.

4. Análisis del proyecto

4.1 Introducción

En este apartado se describirá brevemente las distintas funcionalidades que soporta el buscador implementado. Estas funcionalidades se han resumido en un diagrama de casos de uso y una descripción textual. Además se muestran a continuación la lista de requisitos que se han seguido a la hora de diseñar e implementar el buscador. Los requisitos se van a agrupar en cuatro subcategorías:

- Requisitos Funcionales: especifican "qué" tiene que hacer el software. Definen el propósito del software y se derivan de los casos de uso.
- Requisitos de Rendimiento: especifican valores de rendimiento y escalabilidad para la aplicación.
- Requisitos de Interfaz: especifican el hardware y/o software (como por ejemplo bases de datos) con el que el sistema o componentes del sistema deben interactuar o comunicarse.
- Requisitos de Operación: Los requisitos de operación son aquellos que van a indicar cómo va a realizar el sistema las tareas para las que ha sido construido.

La fuente de los requisitos procede de las reuniones con los directores de proyecto, ideas aportadas por el autor, así como necesidades extraídas del dominio del problema. Todas ellas se describirán en forma de tabla explicando brevemente su significado.

4.2 Identificación de los usuarios finales

Es importante para el análisis del proyecto identificar los distintos roles de usuario y su perfil, tanto técnico como social, ya que esto ayuda a crear sistemas más usables ya que se tiene en cuentas las características de cada tipo de usuario. Por lo tanto, se han detectado un conjunto de posibles roles de usuario y sus perfiles que nos permitirán adaptar tanto de la interfaz de uso Web como la parte de administración a sus características.

Inicialmente el proyecto está planteado para que la aplicación resultante sea usada por alumnos de la Ingeniería Informática de la universidad Carlos III de Madrid, por lo que los usuarios finales dispondrán de cierto conocimiento técnico. A pesar de esto, se ha dotado a la aplicación de un pequeño manual de ayuda online que puede ser consultado si se tiene alguna duda en su utilización.

Existe otro tipo de usuario, que sería el usuario administrador, el cual, tendría un perfil también técnico, ya que en principio la administración correría

a cargo de profesores o alumnos becarios del departamento de ingeniería del software de la universidad Carlos III de Madrid. Se han generado archivos por lotes que simplifican la creación y mantenimiento de la base de datos, así como la ejecución de los índices, pero no se ha definido una interfaz gráfica avanzada, para simplificar las tareas administrativas remotas. Las tareas administrativas son muy simples, tan solo comprobar si los índices siguen activos, ya que toda la gestión se realiza de forma automática.

En cuanto a la monitorización, se ha implementado una simple interfaz gráfica que puede ser o no iniciada y que facilita la monitorización, pero puede hacerse a través de la consola por si se quiere acceder de forma remota al servidor utilizando una conexión por terminal tipo SSH (Security Shell).

4.3 Casos de uso

Los casos de uso fueron introducidos por Ivar Jacobson en 1986 [22] con el objetivo de facilitar la detección de los requisitos funcionales de un sistema, estructurados en torno a diversas categorías de usuarios. Según Jacobson, un caso de uso es una *forma de usar* el sistema por los usuarios, o lo que es lo mismo, una serie de usos típicos del sistema. El modelo de casos de uso se suele expresar gráficamente, pero también puede explicarse de forma textual.

4.3.1 Diagrama de casos de uso

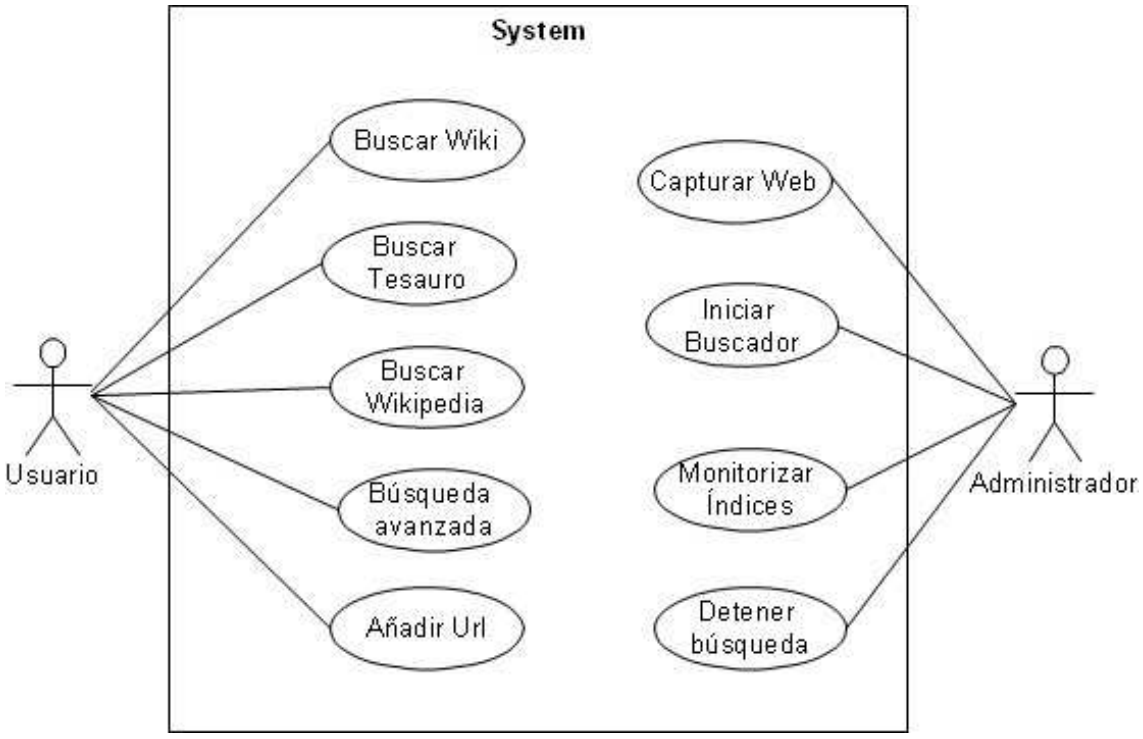

Figura: 4 Diagrama de casos de uso en UML

La Figura: 4 describe en UML el modelo de casos de uso. Como se ha explicado en el apartado 4.2, al sistema pueden acceder dos tipos de usuarios con dos roles claramente diferenciados: los usuarios **administradores** y los **usuarios finales** de la aplicación. Los usuarios finales normalmente accederán a la Web para realizar consultas sobre el wiki indexado. Como se puede apreciar hay 3 tipos de búsquedas:

- Sobre el wiki indexado.
- Sobre términos relacionados en el tesauro
- Sobre términos en la Wikipedia.

Además de estas búsquedas, se dispone de una búsqueda avanzada que permite refinar la búsqueda del usuario añadiendo búsquedas booleanas o determinando la exhaustividad de la búsqueda a realizar.

En cuanto al administrador, su labor es poner en funcionamiento el buscador y supervisar que no se producen erres que impidan el funcionamiento del mismo.

4.3.2 Descripción textual

Para explicar más detenidamente el modelo de casos de uso, a continuación se detallan los casos de uso con una breve descripción textual de los mismos indicando el escenario básico para su utilización.

CU-001	
Nombre	***Buscar Wiki***
Actores	***Usuario***
Objetivo	***Buscar un término en el wiki***
Precondiciones	***Buscador iniciado***
Poscondiciones	***Ninguna***
Escenario básico	***-Usuario selecciona la opción buscar en el wiki.*** ***- Introduce los términos de búsqueda.*** ***- Enviar búsqueda.***

CU-002	
Nombre	***Buscar tesauro***
Actores	***Usuario***
Objetivo	***Buscar términos relacionados en el tesauro que usa el buscador***
Precondiciones	***Buscador iniciado***
Poscondiciones	***Ninguna***
Escenario básico	***-Usuario selecciona la opción buscar en tesauro.*** ***- Introduce los términos de búsqueda.*** ***- Enviar búsqueda.***

CU-003	
Nombre	***Buscar Wikipedia***
Actores	***Usuario***
Objetivo	***Buscar términos en la Wikipedia***
Precondiciones	***Buscador iniciado***
Poscondiciones	***Ninguna***
Escenario básico	***- Usuario selecciona la opción buscar en Wikipedia.*** ***- Introduce los términos de búsqueda.*** ***- Enviar búsqueda.***

CU-004	
Nombre	***Búsqueda avanzada***
Actores	***Usuario***
Objetivo	***Realizar una búsqueda más específica sobre cualquiera de los tres tipos de búsquedas posibles.***
Precondiciones	***Buscador iniciado***
Poscondiciones	***Ninguna***
Escenario básico	***- Usuario selecciona la opción búsqueda avanzada.*** ***- Introduce los términos de búsqueda con todas las palabras*** ***- Introduce los términos de búsqueda con algunas de las palabras*** ***- Introduce los términos de búsqueda sin las palabras*** ***- Introduce los términos de búsqueda exacta*** ***- Determina la precisión de la búsqueda*** ***- Elige si desea ver revisiones o no*** ***- Selecciona el lugar a buscar: Wiki, Tesauro, Wikipedia*** ***- Enviar búsqueda.***

CU-005	
Nombre	***Añadir URL***
Actores	***Usuario***
Objetivo	***Introducir una URL nueva en el buscador.***
Precondiciones	***Buscador iniciado***
Poscondiciones	***Ninguna***
Escenario básico	***- Seleccionar añadir URL*** ***- Introducir la URL propuesta*** ***- Enviar la petición.***

CU-006	
Nombre	*Iniciar Buscador*
Actores	*Administrador*
Objetivo	*Lanzar las aplicaciones que conforman el buscador*
Precondiciones	*Captura realizada*
Poscondiciones	*Buscador iniciado*
Escenario básico	*- Iniciar el Manager* *- Iniciar cada uno de los índices en modo índice o híbrido.* *- Iniciar el buscador.*

CU-007	
Nombre	*Capturar Web.*
Actores	*Administrador.*
Objetivo	*Capturar las páginas del sitio.*
Precondiciones	*Ninguna.*
Poscondiciones	*Captura realizada.*
Escenario básico	*Lanzar el manager.* *Lanzar los índices.*

CU-008	
Nombre	*Monitorizar índices.*
Actores	*Administrador.*
Objetivo	*Controlar el estado de los índices de búsqueda / indexación.*
Precondiciones	*Buscador iniciado.*
Poscondiciones	*Ninguna.*
Escenario básico	*Visualiza si los índices están activos.*

CU-009	
Nombre	*Detener búsqueda.*
Actores	*Administrador.*
Objetivo	*Finalizar la ejecución de los buscadores / indexadores.*
Precondiciones	*Buscador iniciado.*
Poscondiciones	*Buscador finalizado.*
Escenario básico	*Parar el buscador desde el Manager.*

4.4 Requisitos de software

Los requisitos son una descripción completa del comportamiento del sistema que se va a desarrollar. En este apartado se describirán los requisitos software del proyecto, agrupados en las cuatro categorías descritas en el apartado 4.1.

4.4.1 Requisitos funcionales

En este apartado se describirán en forma de tabla los principales requisitos de software que son aquellos que describen que acciones debe llevar a cabo el software que se ha de realizar

Identificador: RF-001	
Nombre	***Búsqueda en el wiki.***
Fuente	***Cliente***
Descripción	***El sistema debe realizar búsquedas en las páginas del sitio http://163.117.147.74/ie/***

Identificador: RF-002	
Nombre	***Búsqueda de términos relacionados***
Fuente	***Cliente***
Descripción	***El sistema debe interactuar con un tesauro que proporcione búsquedas sobre términos relacionados a los inicialmente planteados en la búsqueda, teniendo en cuenta, de forma automática, los sinónimos de los términos a buscar. Los términos propuestos deben estar organizados en forma de árbol, en la que los términos más generales estén en un nivel jerárquico superior a los términos más específicos.***

Identificador: RF-003	
Nombre	***Cambiar la búsqueda por los términos relacionados***
Fuente	***Cliente***
Descripción	***El sistema debe permitir realizar búsquedas por términos relacionados a los incluidos en la búsqueda.***

Identificador: RF-004	
Nombre	***Buscar en Wikipedia.***
Fuente	***Cliente***
Descripción	***El sistema debe permitir realizar búsquedas en la Wikipedia. Para ello se utilizará Google y simplemente se obtendrá el primer valor de la consulta hecha con Google.***

Identificador: RF-005	
Nombre	*Excluir revisiones, discusiones y otras páginas no informativas.*
Fuente	*Cliente*
Descripción	*El sistema debe indizar todo el sitio Web, pero debe dar la opción de excluir las páginas de los resultados que, aunque por el algoritmo de búsqueda resulten relevantes, en la práctica se sepa que no lo son, como pueden ser páginas de revisiones o de discusión, típicas de los entornos Wiki.*

Identificador: RF-006	
Nombre	*Añadir URLs no indexadas*
Fuente	*Autor.*
Descripción	*Si un usuario detecta que una URL del sitio no ha sido indexada, debe poder sugerirla al buscador para que la incorpore a su índice.*

Identificador: RF-007	
Nombre	*Búsqueda booleana*
Fuente	*Cliente*
Descripción	*Se debe permitir búsquedas con conectores OR, AND, NOT*

Identificador: RF-008	
Nombre	*Búsqueda exacta*
Fuente	*Cliente*
Descripción	*Se debe permitir búsquedas con palabras exactas marcas entre comillas.*

Identificador: RF-009	
Nombre	*Ayuda de usuario*
Fuente	*Cliente*
Descripción	*Debe estar disponible en línea un manual de ayuda para el usuario.*

Identificador: RF-010	
Nombre	*Asistente de escritura incorrecta*
Fuente	*Autor*
Descripción	*Si el sistema no encuentra una de las palabras a buscar, debe proponer palabras similares a la buscada para que el usuario pueda corregir su búsqueda en caso de haber cometido algún error de entrada de los datos de búsqueda.*

Identificador: RF-011	
Nombre	*Ordenar por relevancia las búsquedas.*
Fuente	*Cliente*
Descripción	*El buscador debe dar como resultado una lista con las urls de los documentos encontrados, ordenados por su relevancia, de forma que los más relevantes, (es decir los que se consideren más próximos a lo que el usuario pretende buscar) se ordenarán al principio de la lista.*

4.4.2 Requisitos de rendimiento

Identificador: RR-001	
Nombre	*El sistema debe ser escalable.*
Fuente	*Cliente.*
Descripción	*El sistema debe permitir realizar, de forma eficiente, búsquedas en dominios más amplios al propuesto inicialmente, como por ejemplo otros sistemas Wiki más grandes como puede ser la Wikipedia.*

4.4.3 Requisitos de interfaz

Identificador: RI-001	
Nombre	*Compatibilidad con principales navegadores*
Fuente	*Cliente*
Descripción	*La interfaz Web debe ser compatible con Internet Explorer, Firefox y Safari*

Identificador: RI-002	
Nombre	*Validación XHTML 1.0 Transactional*
Fuente	*Autor*
Descripción	*La página Web debe estar validad para XHTML 1.0 Transactional por la W3C.*

Identificador: RI-003	
Nombre	*Validación de accesibilidad AA*
Fuente	*Autor*
Descripción	*La página Web debe estar validad con accesibilidad AA en TAW.*

Identificador: RI-005	
Nombre	*Usabilidad.*
Fuente	*Cliente*
Descripción	*La página Web debe ser usable*

4.4.4 Requisitos de operación

Identificador: RO-001	
Nombre	*Interfaz Web para el usuario final*
Fuente	*Autor*
Descripción	*La interfaz del usuario final debe ser accesible por Web.*

Identificador: RO-002	
Nombre	*Supervisión del buscador*
Fuente	*Autor*
Descripción	*El usuario administrador deberá poder supervisar el funcionamiento del buscador tanto de forma local como de forma remota a través de un terminal.*

4.5 Modelo E/R de la base de datos

El índice generado por el buscador se almacena en una base de datos. Como hay varios índices que pueden estar en varias máquinas, cada índice crea una base de datos propia en su máquina local. El modelo Entidad–Relación (E/R) de la base de datos de los índices se puede ver en la Figura: 5. El manager lleva la cuenta de que páginas están pendientes de procesar y quien (que índice) las ha procesado, así pues, sólo almacena una tabla en la base de datos como se muestra en la Figura: 6.

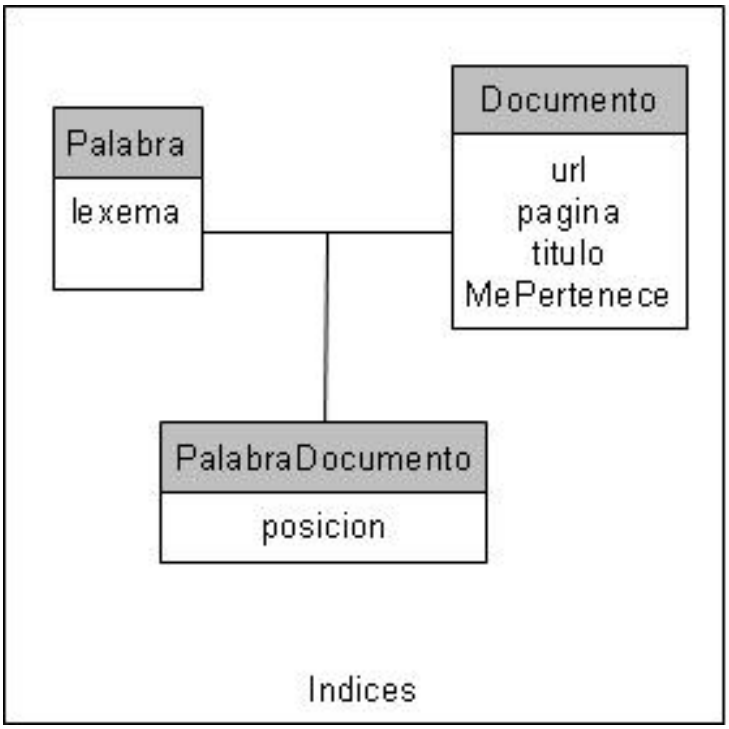

Figura: 5 Modelo E/R del indice

Figura: 6 Modelo E/R del Manager

Como se puede observar en la Figura: 5, el índice construido es un índice invertido, en la que se almacena las palabras, indicando a que documento pertenecen y en que posiciones dentro del documente se encuentra. En la base de datos además se guarda el texto íntegro de la página, quitando espacios para facilitar la búsqueda exacta y para determinar si la página ha cambiado o no. El atributo 'Me pertenece' sirve para saber si la página a sido procesada por el índice o la a obtenido a través de otro índice.

4.6 Planificación del proyecto

Para el apartado de planificación del proyecto, hemos tenido que calcular el mismo, extrapolando a una jornada laboral de 8 horas al día. Como no se ha tenido dedicación exclusiva al desarrollo del proyecto, el tiempo de desarrollo ha sido muy superior a los incluidos aquí en esta planificación. A si pues es una planificación teórica de lo que se tardaría en desarrollar cada una de las actividades que lo componen, si se dispusiera de tiempo suficiente para trabajar en el 8 horas diarias, todos los días de la semana.

A continuación se muestra el diagrama de Gantt realizado con Microsoft Proyect 2000 a fecha de inicio del 1-10-2007, fecha en el que se inició el proyecto y que finaliza según la planificación el 15-4-2008, empleando un único trabajador a jornada completa. La duración por tanto sería de 7 meses y medio.

Id	Nombre de tarea
1	**Análisis**
2	Obtención de requisitos
3	Análisis del modelo de Datos
4	casos de uso
5	planificación proyecto
6	
7	**Diseño**
8	Elección de la tecnologia
9	Diseño Arquitectonico
10	Diseño de la base de datos
11	Diseño arquitectura comunicaciones
12	Diseño de protocolos
13	Diseño de la estructura de clases
14	
15	**Implementación**
16	Implementacion de la Base Datos
17	implementación del indice
18	Implementacion manager
19	Implementación buscador
20	Implementación Interfaz
21	Implementación ayuda linea
22	
23	**Validación y Pruebas**
24	Validación indizacion
25	Validación algoritmo búsqueda
26	Validación interfaz
27	Validación usabilidad
28	
29	Implantación
30	Memoria

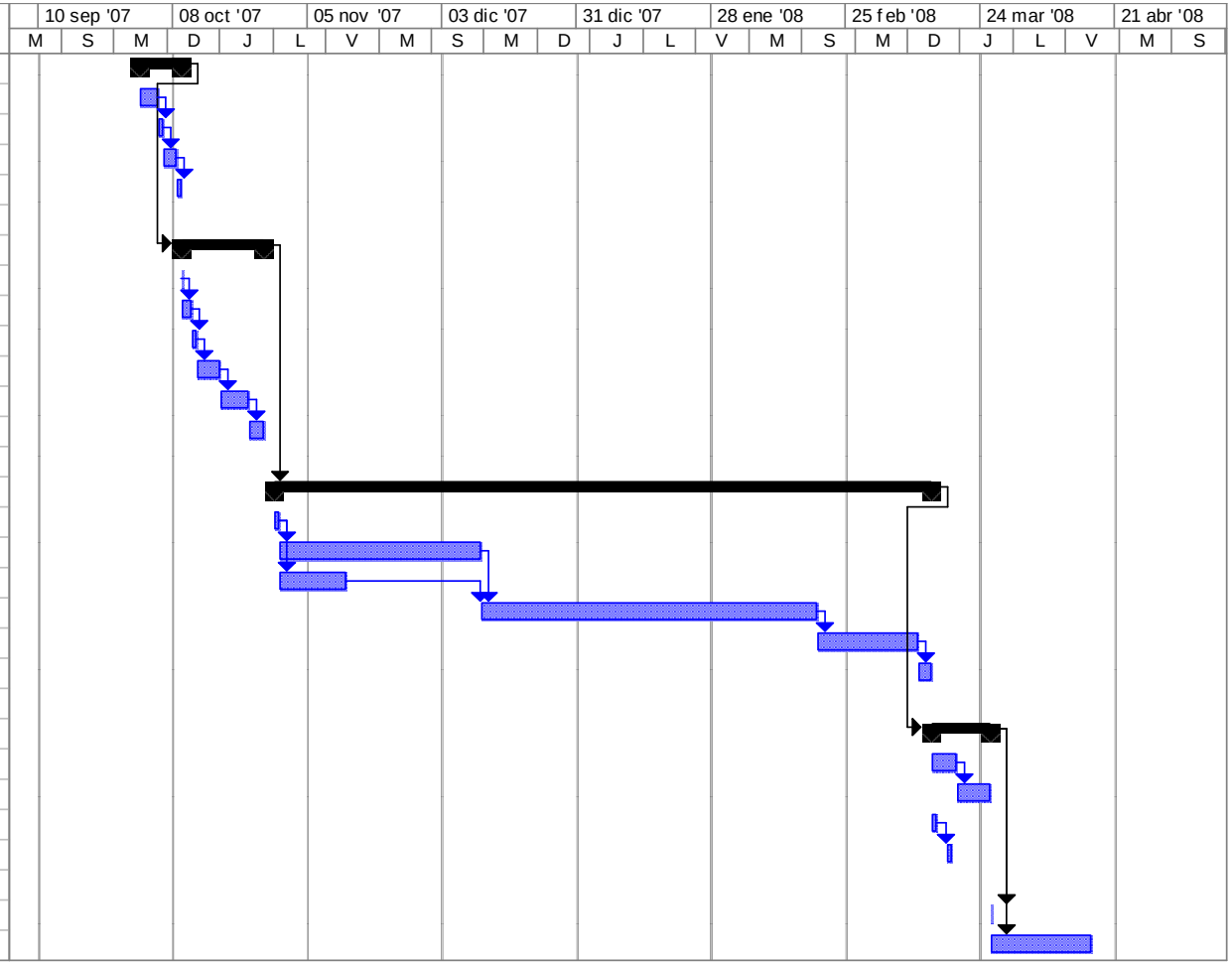

Figura: 7 Diagrama Gantt de la planificación del proyecto

FASE DE ANÁLISIS

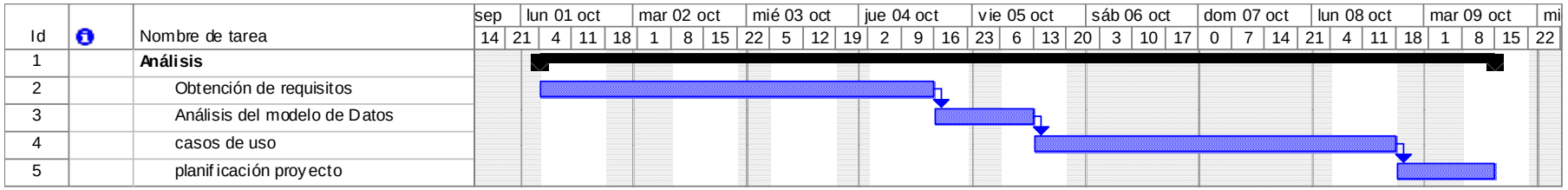

Figura: 8 Diagrama Gantt de la fase de análisis.

FASE DE DISEÑO

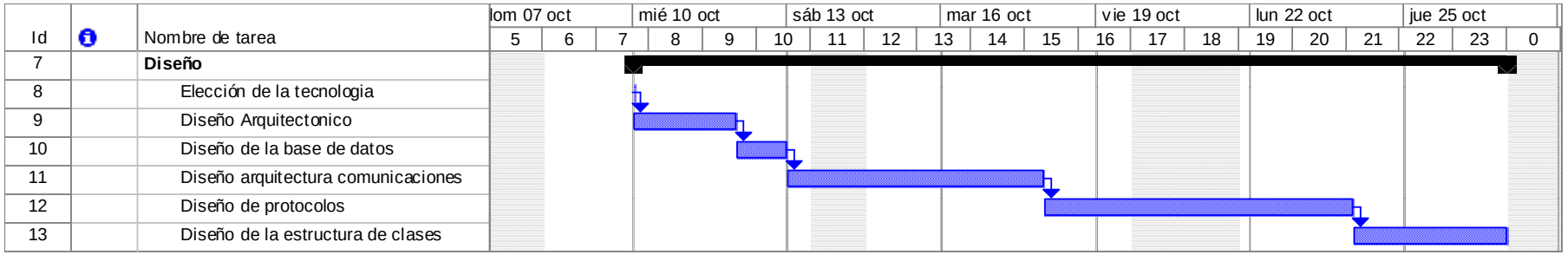

Figura: 9 Diagrama Gantt de la fase de diseño.

FASE DE IMPLEMENTACIÓN

Id	Nombre de tarea
15	**Implementación**
16	Implementacion de la Base Datos
17	implementación del indice
18	Implementacion manager
19	Implementación buscador
20	Implementación Interfaz
21	Implementación ayuda linea

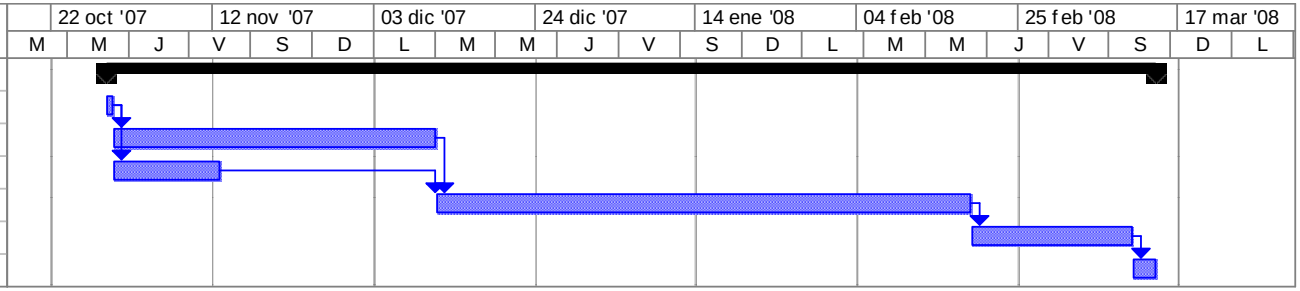

Figura: 10 Diagrama Gantt de la fase de implementación.

VALIDACIÓN, PRUEBAS, AYUDA E IMPLANTACIÓN.

Id	Nombre de tarea
23	**Validación y Pruebas**
24	Validación indizacion
25	Validación algoritmo búsqueda
26	Validación interfaz
27	Validación usabilidad
28	
29	Implantación
30	Memoria

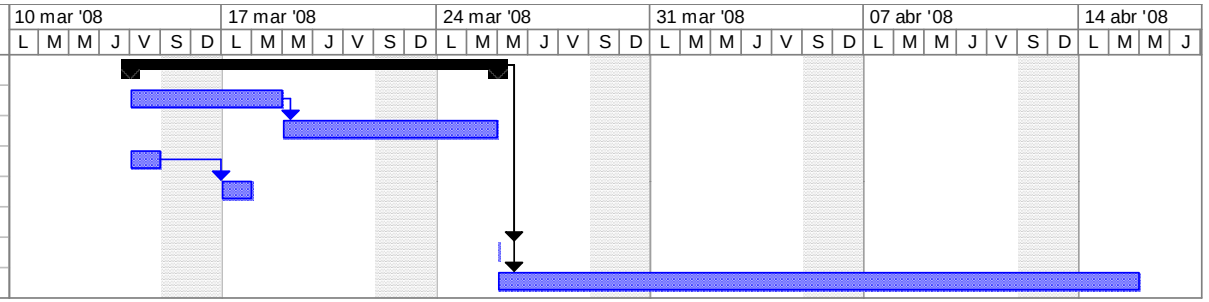

Figura: 11 Diagrama Gantt de las fases de validación - pruebas, implantación y memoria.

4.7 Presupuesto estimado

Para la creación del presupuesto, se ha tenido en cuenta la planificación realizada en el apartado anterior al calcular el coste en recursos humanos del proyecto. Para realizar el cálculo se han definido varios roles de personal. Por un lado se necesitaría un analista para las fases de Análisis. Para la fase de diseño se necesitaría un diseñador de software. Para la implementación del proyecto se necesitaría un programador y para la interfaz Web sería preciso un diseñador gráfico. Además sería necesario un evaluador independiente para realizar el proceso de validación. Los días de trabajo se calculan en base a la planificación, incluyendo fines de semana y festivos a razón de 8 horas al día.

El salario bruto anual para cada trabajador en 2008, los días trabajados y su coste total se muestran en la Tabla 1:

PUESTO	*Bruto anual*	*Diario*	*Días trabajo*	*Total*
Analista	*34.000 €*	*94 €*	*43 días*	*4.042 €*
Diseñador software	*25.000 €*	*68 €*	*27 días*	*1.836 €*
Programador	*22.000 €*	*60 €*	*190 días*	*11.400 €*
Evaluador	*21.000 €*	*57 €*	*16 días*	*912 €*
Diseñador gráfico	*20.000 €*	*54 €*	*7 días*	*378 €*
TOTAL				*18.569 €*

Tabla 1 Presupuesto en RRHH

El software necesario para realizar el proyecto sería el siguiente:

Software	*Descripción*	*Coste por licencia*	*Número de licencias*	*Total*
Windows XP profesional	*Sistema operativo*	*135 €*	*2*	*270 €*
Notepad++	*Editor texto*	*0 €*	*1*	*0 €*
Gimp	*Herramienta gráfica y retoque fotográfico*	*0 €*	*1*	*0 €*
XAMPP	*Servidor Web y Base datos integrada*	*0 €*	*2*	*0 €*
Mozilla Firefox	*Navegador Web*	*0 €*	*1*	*0 €*
Safari	*Navegador Web*	*0 €*	*1*	*0 €*
JDK 1.6	*Entorno de programación Java de Sun Microsystem*	*0 €*	*1*	*0 €*
JRE 1.6	*Maquina virtual de Java*	*0 €*	*2*	*0 €*
Microsoft Office 2003	*Suite de oficina de Microsoft*	*150 €*	*1*	*150 €*
Eclipse	*Herramienta de ayuda a la programación.*	*0 €*	*1*	*0 €*
TOTAL				*320 €*

Tabla 2 Presupuesto del software

El hardware necesario para realizar el proyecto sería el siguiente:

Hardware	*Descripción*	*Coste por unidad*	*Número de unidades*	*Total*
Equipo servidor	*Quad-core 2 GB RAM*	*700 €*	*1*	*700 €*
Equipo desarrollo	*Core 2 Duo 2 GB RAM con monitor*	*500 €*	*1*	*500 €*
TOTAL				*1.200 €*

Tabla 3 Presupuesto Hardware

Sumando todos los costes, el coste total se muestra en la Tabla 4:

Concepto	*Descripción*	*Coste*
Coste RRHH	*Coste del personal necesario para desarrollar el proyecto*	*18.579 €*
Coste Software	*Coste del software necesario para desarrollar el proyecto*	*320 €*
Coste Hardware	*Coste del hardware necesario para desarrollar el proyecto*	*1.200 €*
Costes totales		*20.099 €*
Beneficios	*Beneficio obtenido por el equipo de desarrollo del 20 % sobre los costes totales.*	*4.019 €*
TOTAL		*24.099 €*

Tabla 4 Presupuesto Final

5. Diseño arquitectónico

5.1 Patrón de diseño arquitectónico. MVC

El sistema ha sido diseñado pensando siempre en las siguientes ideas claves:

- El sistema debe ser escalable ya que debe soportar que el sitio a indizar evolucione y crezca.
- El sistema debe ser versátil y adaptable a cualquier sitio Web con el menor número de cambios posibles en el código.
- EL sistema debe estar estructurado de forma que pueda ser fácilmente modificable y ampliable en el futuro (por tanto, mantenible y escalable).

Para conseguir estos objetivos de diseño, el sistema está planteado siguiendo el patrón de diseño: Modelo – Vista – Controlador (MVC).

Este modelo es útil debido a que separa perfectamente los datos, de la lógica de negocio y de la presentación de todo esto al usuario, lo que permite una mayor abstracción y facilita el mantenimiento del sistema. Además este patrón se utiliza frecuentemente en aplicaciones Web [20].

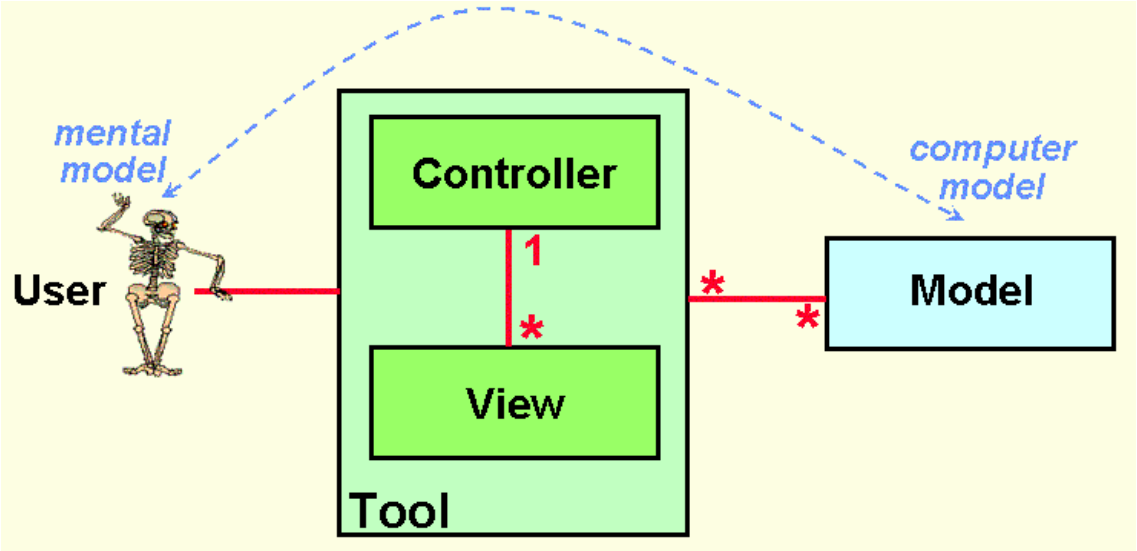

Figura: 12 Patrón clásico Modelo – Vista – Controlador (MVC) [20]

Este patrón de diseño se compone de 3 entidades que son:

[20] Imagen obtenida de: http://heim.ifi.uio.no/~trygver/themes/mvc/mvc-index.html

- **Modelo**: Esta entidad es la representación específica de la información con la cual el sistema opera. La lógica de datos asegura la integridad de estos. Todos los accesos a los datos se hacen con procedimientos almacenados de MySQL para optimizar los tiempos de acceso.
- **Vista:** Esta entidad presenta el modelo en un formato adecuado para interactuar con el usuario. La vista ha sido desarrollada con tecnología Web mezclando HTML, CSS, PHP y Javascript y las imágenes con el programa de dibujo GIMP[21].
- **Controlador**: Esta entidad responde a eventos, usualmente acciones del usuario e invoca cambios en el modelo y en la vista. El controlador ha sido desarrollado íntegramente en Java.

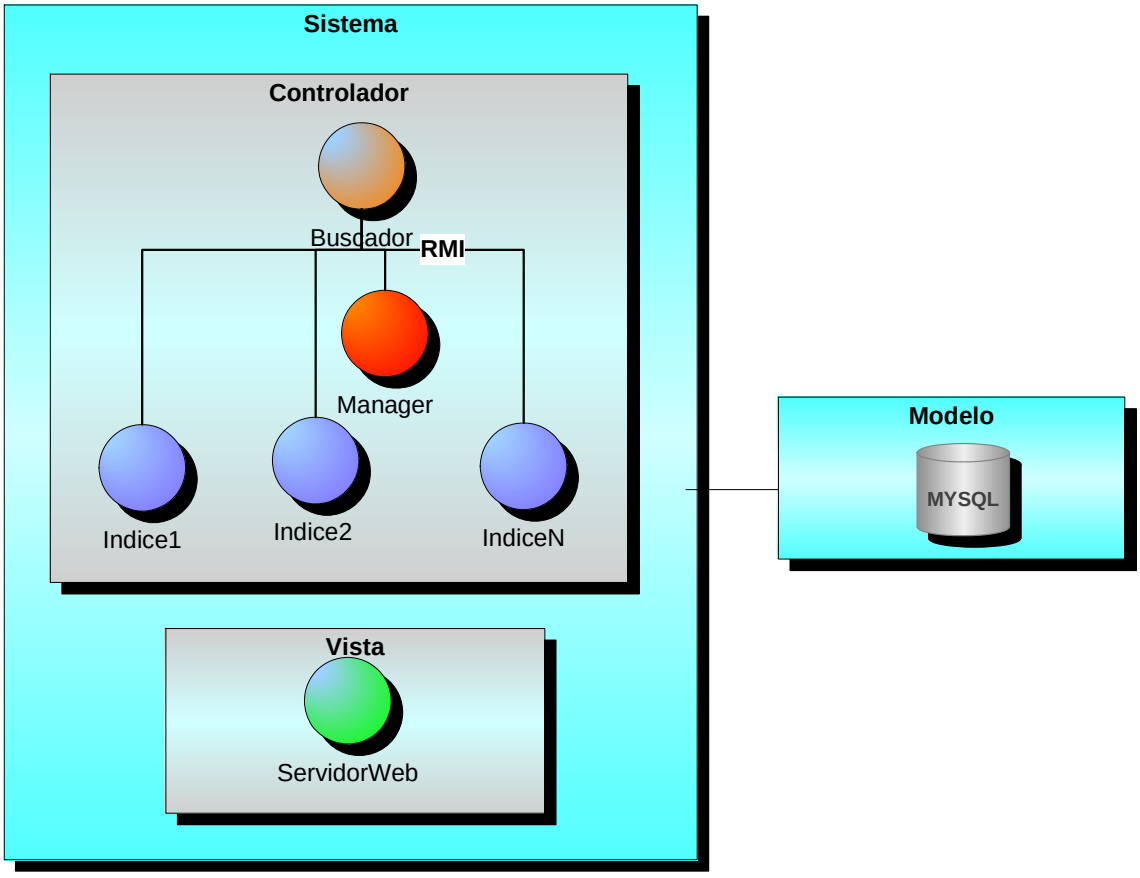

Figura: 13 Adaptación del patrón MVC al buscador.

Al ser una aplicación distribuida, cada nodo posee una parte del modelo, la cual gestiona. Esto hace que existan datos duplicados en los nodos, un problema que se intenta minimizar con la existencia del Manager, que coordina a todos los nodos de búsqueda. De todas formas, para minimizar las comunicaciones por red, los índices tienen copias de los documentos que han

[21] Editor de imágenes libre http://www.gimp.org.es/

sido procesados por otros índices, pero no de las palabras que no les sean asignadas. Así pues, los documentos están almacenados en todas las máquinas, pero las palabras están distribuidas entre todos los índices. Como hay muchas más palabras que documentos, el tamaño de la base de datos es menor que si se duplicasen las palabras. De esta forma conseguimos minimizar las comunicaciones entre los agentes a la vez que mantenemos no demasiado elevado en disco, lo que permite una mayor escalabilidad. Para la página indizada, la base de datos ha ocupado 220 MB en total usando dos índices, unos 100 MB por índice, almacenando aproximadamente 13.000 urls diferentes.

5.2 El modelo

El modelo está gestionado por la Base de datos Mysql versión 5.1. Esta base de datos soporta transacciones y procedimientos almacenados que son utilizados para garantizar la integridad de los datos del sistema. Además existe una clase Java dentro del Controlador que se encarga de operar con la misma a través de **JDBC**.

Cada uno de los nodos del sistema tiene su propia base de datos, excepto el Buscador que no dispone de ninguna excepto el tesauro que puede estar en una base de datos o en un fichero. En esta implementación del sistema, el tesauro está almacenado en un fichero.

- Los índices guardan en la base de datos una lista invertida en la cual se asocia a cada palabra encontrada en el corpus documental, en qué documentos ha sido encontrada, así como la posición en dicho documento. Se almacena la posición para tener en cuenta la proximidad de las palabras en el algoritmo de búsqueda.
- El manager también dispone de una base de datos donde almacena las páginas que los índices encuentran, un resumen de las mismas calculado con MD5 y el nombre del agente *indexador* que lo ha procesado.

El modelo relacional de la base de datos se observa en la Figura: 14.

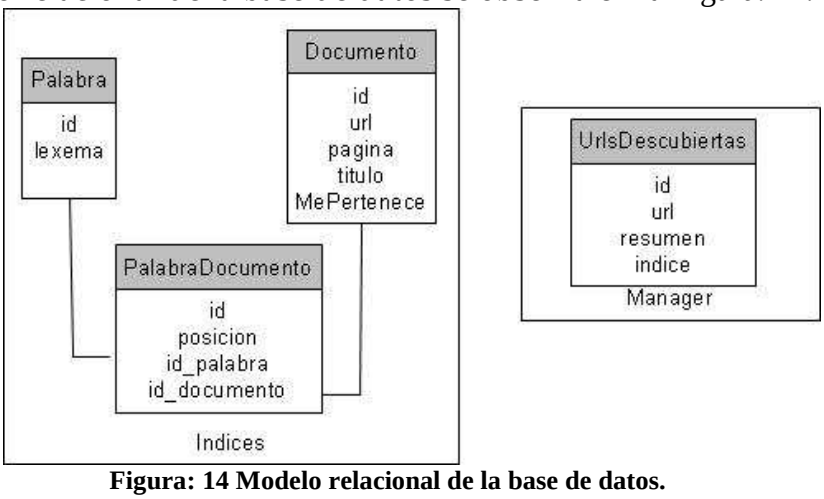

Figura: 14 Modelo relacional de la base de datos.

5.3 La vista

La vista está desarrollada como interfaz Web y se comunica con el controlador a través de un protocolo definido en XML. El protocolo es muy simple y es procesado por un modulo PHP independiente, lo que garantiza su abstracción si dicho protocolo cambiase en el futuro.

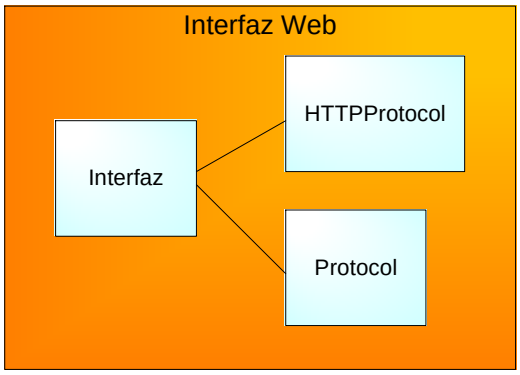

Figura: 15 Esquema del componente Vista

La comunicación con el buscador es a través del protocolo HTTP. El XML se envía dentro de un comando POST adoptando, en este caso la interfaz Web el rol de Cliente, y el buscador el rol de Servidor Web. Al estar completamente separada la vista del controlador, se puede sustituir esta interfaz de forma fácil, ya que la interfaz de comunicación entre la *Vista* y el *Controlador* es un protocolo descrito en una DTD. Una vez recogida la pregunta del usuario, el componente *Vista* espera la respuesta del componente *Controlador* con las páginas Web más relevantes, así como las sugerencias provenientes del tesauro, formateándolos para su adecuada presentación por pantalla.

5.4 El Controlador

El Controlador es la pieza fundamental del sistema y donde reside toda la lógica del buscador. Como mencionamos anteriormente, está compuesto de 3 tipos de nodos cada uno con su funcionalidad: el Manager, el Buscador y los Índices. El esquema de clases y comunicaciones del módulo Controlador se muestra en la Figura: 16.

El manager coordina al resto de elementos, luego debe ser el primero en iniciarse. El manager se encarga de crear el servidor de nombres de RMI (Registro) y se comunica con el resto de elementos mediante retrollamadas o Callbacks. De forma que los índices deben registrarse en el Manager para que este pueda invocar sus métodos. Una vez iniciado, el manager mira su configuración para determinar cuántos índices se deben conectar y cuál es la página y dominio objetivo de la búsqueda y espera a que todos los agentes estén conectados. Los índices se pueden encontrar en tres modos:

- Modo boot: El índice se comporta como una araña e intenta descubrir y procesar las páginas que le va proporcionando en round-robin el Manager. Las URLs nuevas se las pasan al Manager para que las almacene y vaya construyendo el índice de sus documentos. A cada índice le corresponden un grupo de palabras a procesar. Este grupo de palabras es determinado por un valor hash MD5 de la palabra módulo el número de índices que haya, de forma que va balanceando la carga de palabras y de búsqueda entre los diferentes índices. Si una palabra no pertenece al índice que la adquiere, éste se la envía al índice que le pertenezca, junto con los datos del documento que la contiene, de forma que los documentos, normalmente estarán almacenados en todos los índices, y serán sólo las palabras las que estarán balanceadas. Los índices se comportan de esta manera para minimizar en la búsqueda las comunicaciones entre los índices y con el manager, sacrificando espacio de almacenamiento. Es de suponer que el espacio de almacenamiento en un sistema distribuido sea un problema menor tratándose de ficheros de texto, que el tiempo de las comunicaciones.
- El modo índice: En este modo, el índice espera a que el buscador le realice las peticiones. La petición básica que realiza el buscador al índice es ¿Qué documentos contienen la palabra X? A esto el índice le responde con una lista documentos y las posiciones de dicha palabra en cada documento, así como un resumen del contenido de la página entorno a la primera aparición del término en el documento, así como el título y la URL de dicho documento. Con esta información, el buscador puede calcular la importancia de ese documento para la consulta realizada, calcular las intersecciones y uniones de conjuntos que necesite para las búsquedas booleanas o a partir de los ficheros almacenados en disco, obtener las búsquedas exactas.
- En modo update: El índice a demás de resolver las preguntas del buscador, actualiza cada cierto tiempo alguna página del índice si detecta que se ha modificado, alertando al resto de los cambios producidos si fuera necesario. En este modo se detecta un cambio cuando la página ha modificado su contenido, dejando cambios como espacios en blanco, cabeceras, código JavaScript, etc., fuera de dicha actualización. El sistema cuando encuentra un enlace nuevo lo incorpora en la URL y se comporta como un índice para adquirir nuevas páginas que vayan apareciendo.

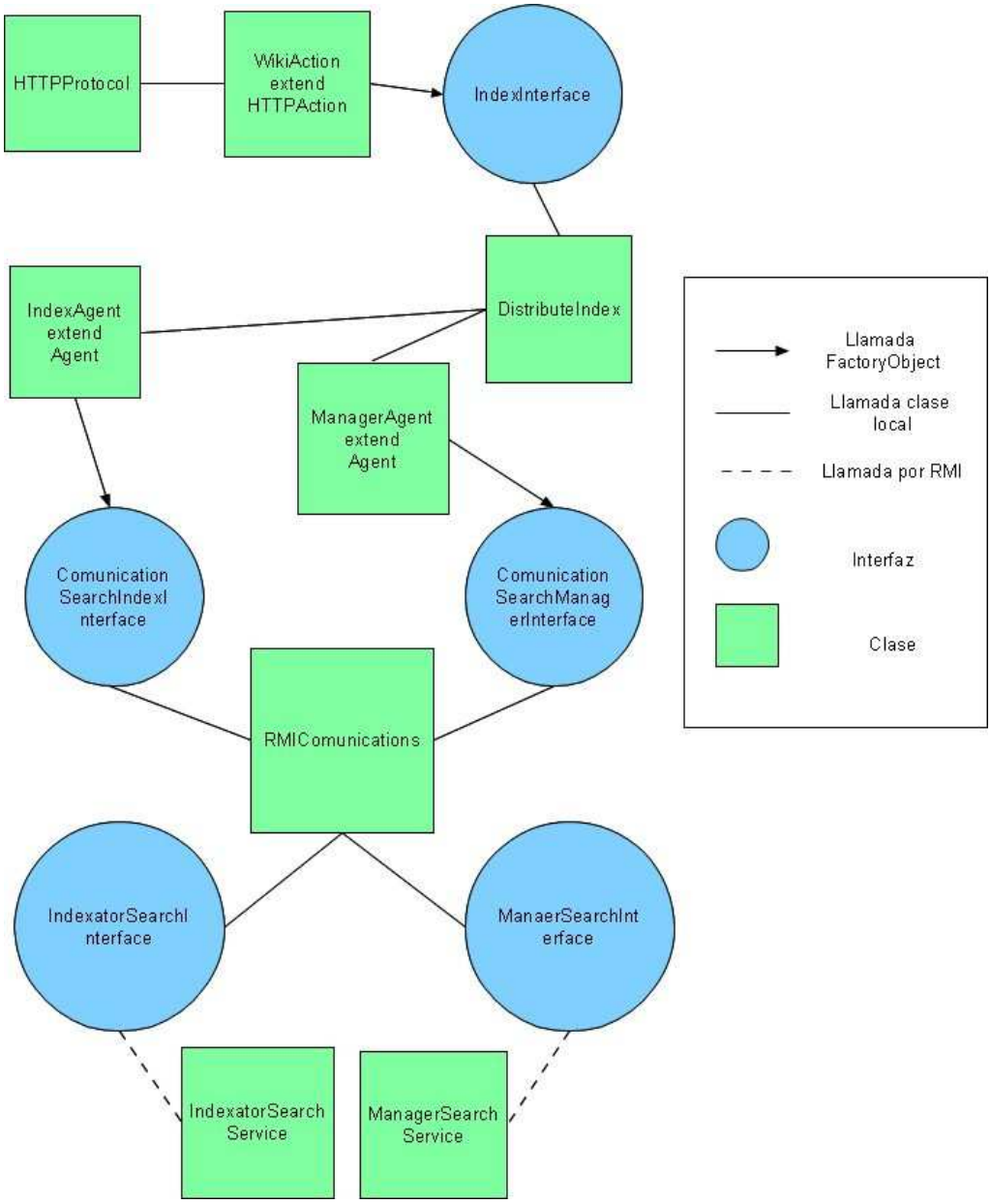

Figura: 16 Esquema de comunicaciones entre clases del controlador

Finalmente es el buscador el que se comunica con la interfaz, comportándose como un servidor Web. El buscador espera peticiones HTTP POST con las opciones elegidas por el usuario. El buscador permite búsqueda booleana con los operadores AND, NOT y OR. Por defecto las palabras incluidas son tratadas como AND.

Una vez que el buscador interpreta el XML con las órdenes del usuario, este se conecta con los índices que gestionan las palabras de la búsqueda para recuperar sus documentos. Además de las palabras incluidas en la búsqueda por el usuario, se añaden sus sinónimos, extraídos del tesauro. Estos sinónimos tendrán menos peso en la búsqueda, pero también serán tenidos en cuenta.

Cuando se tienen todos los documentos, el buscador los valora siguiendo un algoritmo basado en TF-IDF, pero modificado a las características del sistema. El algoritmo se detallará en el apartado 6.2. Como principal característica se destaca que tiene en cuenta: la proximidad de las palabras, si las palabras están en el título del documento y en la URL, la importancia de la palabra (si es muy descriptiva dentro del conjunto de palabras almacenadas) y las veces que se repite en el documento.

A parte de la lista de documentos encontrados, el buscador envía a la interfaz el árbol del tesauro con los términos relacionados con los que ha buscado. De esta forma, el usuario puede hacer consultas de términos similares o los que indicó previamente para obtener más información o simplemente para refinar los resultados obtenidos.

Todas las comunicaciones como ya se ha mencionado, se realizan con RMI por su potencia y simplicidad a la hora de desarrollar el sistema, excepto la conexión entre la interfaz y el buscador que se realiza por sockets.

5.5 Diseño basado en plugin o conectables

Todo el diseño se ha pensado para que el sistema sea lo más fácilmente ampliable y mantenible posible. El diseño del controlador se ha basado en la utilización de módulos que se conectan unos con otros a través de unas interfaces constantes predefinidas. Estas interfaces ocultan los detalles de implementación de dichos módulos, lo que facilita la modificación del software en el futuro. Java dispone de un sistema de carga de clases que permite implementar un sistema de módulos basado en **conectables** o **plugins**. La clase de java es **URLClassLoader** que permite cargar una clase sabiendo su nombre.

Los plugins son módulos que incrementan las funcionalidades de un software. Se puede crear un módulo cargador de plugins genéricos utilizando el patrón de diseño *Command*, pero en nuestro caso lo que nos interesa no es ampliar la funcionalidad de forma dinámica, sino poder modificar la forma en la que el sistema realiza ciertas cosas.

La clase de carga de módulos es FactoryObject que sigue el patrón AbstractFactory[22] y a la vez el patrón de diseño Singleton[23] e incorpora la clase URLClassLoader. Esta clase, al ser cargada, lee dos ficheros, el modules.cfg y el mapingServices.cfg. El fichero modules determina todos los módulos que son

[22] Diagrama del patrón Abstract factory http://es.wikipedia.org/wiki/Abstract_Factory_(patr%C3%B3n_de_dise%C3%B1o) [consultado 8/06/2008]
[23] Diagrama patrón Singleton http://es.wikipedia.org/wiki/Patr%C3%B3n_de_dise%C3%B1o_Singleton [consultado 8/06/2008]

tratados como conectables. Un ejemplo de este fichero de configuración es el siguiente:

```
thesaurus   thesaurusAccess   Thesaurus.ThesaurusAccess
index       distributeIndex         Index.DistributeIndex
index       localIndex              Index.LocalIndex
comunications rmiComunications
      Index.Comunications.RMIComunications
thesaurus   thesaurusFile           Thesaurus.ThesaurusFile
```

La primera columna indica el nombre del servicio que aporta, la segunda columna indica el nombre de la clase que implementa dicho servicio, y la tercera, la clase en sí que lo implementa.

El fichero mapingServices simplemente determina de todos las implementaciones de un servicio, cuál es el que se desea utilizar. De forma que modificar o ampliar la capacidad de cualquiera de estos servicios es tan sencillo como, respetando su interfaz, implementar una clase y añadir su nombre a estos ficheros de configuración, para que el sistema se comporte de acuerdo a la nueva implementación.

Se han definido 3 funcionalidades modificables. Pero se podrían añadir más si fuera necesario.

- Index: Es el módulo que implementa el índice. En la Figura: 16 se puede apreciar como el buscador (WikiAction) crea un índice que realmente no sabe cómo está implementado y del que sólo conoce su interfaz. Lo utiliza como si fuera una clase local, pero realmente es un índice distribuido (DistributeIndex). Este índice podría estar en una sola máquina o simplemente que lea del sistema de ficheros local todos los documentos. Incluso se podría tener implementados los dos módulos para adaptar la herramienta a varias necesidades. Todo esto no modificaría en nada el resto del programa y sólo habría que añadir la clase que gestionara todo esto y añadirla en los ficheros de configuración mencionados anteriormente.
- Comunications: es la comunicación entre los índices y el manager. El sistema está diseñado para que dicha comunicación sea transparente, es decir que sea independiente del protocolo de comunicación. Puede hacerse a través de Sockets o a través de RMI o cualquier otro método. Como se puede apreciar en la figura, tanto ManagerAgent como IndexAgent crean su comunicación a través de la factoría, que realmente implementa ambas interfaces. La clase que mapea esta comunicación podría ser definida más adelante mediante sockets si fuese necesario.
- Thesaurus: está funcionalidad maneja el tesauro que incorpora el buscador. La Figura: 16 no lo muestra, porque lo carga el buscador

y no hay comunicación remota. El tesauro está definido en un fichero de texto, pero podría estar en una base de datos Mysql, Oracle, o incluso distribuido en red. De la forma que dependiendo de dónde esté almacenado y de su estructura, simplemente hay que modificar la clase que lo maneja y el resto del sistema funcionará de la misma forma que antes.

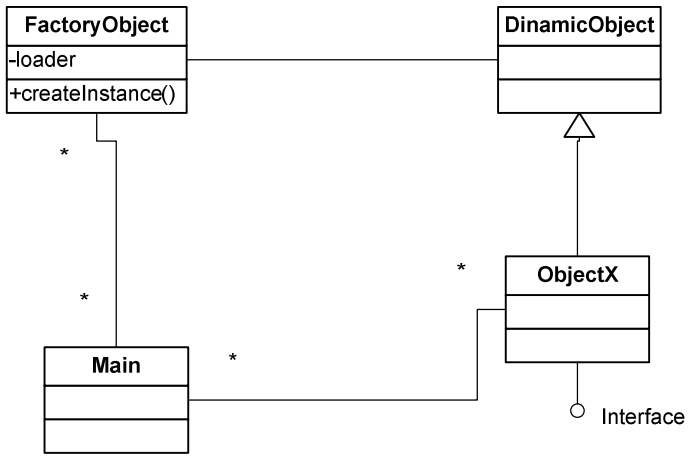

Figura: 17 Estructura del FactoryObject

En la Figura: 17 vemos el esquema de carga de la case ObjectX por parte de Main utilizando para ello la clase FactoryObject.

La clase Main accede a la factoría, ejecutando al método createInstance() al que le indica el servicio que quiere ejecutar. Este, atendiendo a su fichero de configuración (mapingService), determina que el objeto es ObjectX que debe heredar de DinamicObject e implementar la interfaz que determina como usarlo.

A pesar de la aparente complejidad del diseño ideado, pensamos que ese diseño facilita enormemente el proceso de mantenimiento y modificación posterior del sistema, lo que permite que el buscador pueda fácilmente adaptarse a múltiples entornos y escenarios, haciéndola una herramienta muy versátil y extensible para ser perfeccionada en el futuro, o aplicada a otros entornos diferentes al de este proyecto.

6. Diseño detallado

6.1 Funcionalidades del buscador

En este apartado se describe todas las funcionalidades que dispone el usuario a la hora de utilizar el buscador y la interfaz gráfica de la herramienta.

La interfaz básica del buscador se muestra en la Figura: 18.

Figura: 18 Interfaz del buscador

La barra central sirve para escribir las palabras claves de la búsqueda. Si no se especifica nada, el buscador examinará documentos donde estén todas las palabras que se indiquen en la barra. Si se quiere expresar que dichas palabras puedan estar o no, se utilizará el operador OR que se escribe poniendo delante de las palabras concatenadas con OR el carácter "|". Si se quiere decir que no aparezcan ciertas palabras en los documentos, se indicarán en la lista de NOT colocando el carácter "-"delante de la palabra. Un ejemplo de una búsqueda sería:

Recuperación información –tesauro –software |ontologías

Que significaría: *dime los documentos donde aparezca la palabra Recuperación de información y pueda aparecer la palabra ontologías, pero no aparezca ni tesauro ni software.*

El buscador permite escribir palabras con y sin acentos, ya que los elimina. Esto puede provocar errores ya que ciertas palabras acentuadas no significan lo mismo que las mismas palabras sin acentuar, pero las ventajas que aporta al usuario son muy grandes ya que mejora los resultados de la búsqueda al eliminar errores ortográficos en los documentos y permite la búsqueda aunque el usuario escriba la palabra sin acentos.

Además, el buscador omite los plurales acabados en es y s. También puede cometer errores por esta simplificación, pero generalmente los resultado aplicando esta lematización son mejores que si no se aplican ya que se pierden muchas palabras que significan lo mismo.

El usuario no necesita recordar los caracteres especiales para búsqueda booleana, ya que la herramienta dispone de una búsqueda avanzada donde indica el lugar donde debe colocar el usuario la lista de palabras del AND, OR y NOT. Se puede apreciar estas opciones la figura Figura: 19. Además de esta simplificación para el usuario, en la búsqueda avanzada, se puede determinar la precisión de la búsqueda. Hay cinco posibles posiciones, búsqueda muy rápida, búsqueda rápida, búsqueda normal, búsqueda lenta y exhaustiva. La única diferencia entre una y otra búsqueda es el número de documentos por palabra que procesa el buscador, normalmente con rápida y media se obtienen los mismos resultados que con el resto pero en ciertas ocasiones, puede mejorar algo la búsqueda aunque, como es lógico, incrementa el tiempo de procesado.

En la búsqueda avanzada también se puede definir si se quiere mostrar los resultados de los documentos antiguos o en revisión o no. Por defecto no sacará las revisiones ya que no se consideran relevantes. La exclusión de estas urls se define en una pequeña clase aislada del resto del código. Si se aplicase el buscador a otro entorno diferente que no tuviera estas características, habría que modificar sólo un método de una clase.

La búsqueda, como se puede apreciar tanto en la Figura: 18 como en la Figura: 19, se puede realizar sobre 3 dominios: uno es el wiki del departamento de ingeniería del software de la Universidad Carlos III de Madrid, objeto del trabajo de este proyecto, pero también se puede hacer búsqueda sobre el propio tesauro que utiliza el sistema para buscar términos relacionados. Además, existe una tercera opción que consiste en hacer búsquedas sobre la Wikipedia. A pesar que el buscador puede perfectamente indizar la Wikipedia no se ha construido el índice para la misma, y estos resultados se obtienen utilizando el buscador de Google. Como se puede observar en la Figura: 20, sólo se recupera el primer resultado obtenido por Google.

Figura: 19 Interfaz de búsqueda avanzada.

Cuando se busca en el tesauro, los términos obtenidos pueden ser buscados nuevamente en el tesauro o en el wiki pulsando sobre el enlace correspondiente.

Figura: 20 Resultados de búsqueda en la Wikipedia

Por último, el buscador aporta una ayuda al usuario si este se equivoca al escribir una palabra. Si el usuario escribe una palabra que no encuentra en la Web indizada, advierte de este hecho y muestra una lista de 5 palabras que el buscador considera más parecidas a la introducida incorrectamente. Pulsando sobre la palabra a corregir, automáticamente esta se corrige de la barra de búsqueda, permitiendo nuevamente preguntar al buscador.

6.2 Algoritmo de búsqueda

El algoritmo de búsqueda que utiliza el sistema, ordena los documentos según su relevancia, teniendo en cuenta los términos introducidos por el usuario en su consulta. Este algoritmo dispone de unos pesos que pueden ser configurables de forma externa a la aplicación a través de un fichero de configuración, lo que permite modificar los valores de los pesos sin compilar el programa.

El peso de un documento i $w(D_i)$ es la suma del peso de todas las palabras de la búsqueda que estén dentro del documento i multiplicado por una serie de factores:

- K: es el factor que determina el peso de la palabra. Si es una palabra original de la búsqueda tendrá un peso, si es una palabra obtenida como sinónimo del tesauro, tendrá otro peso distinto.
- U: es un factor que depende de si dicha palabra aparece o no en parte de la URL. Si aparece en la URL, tendrá más valor (normalmente) si no multiplicará por 1.
- T: es un factor que depende de si dicha palabra aparece o no en el título del documento. Si no aparece se multiplicará por 1.
- Prox: indica el número de palabras que aparecen en un entorno dado en configuración. Si varias palabras de la búsqueda están en ese entorno (por ejemplo a una distancia de cómo mucho 3 palabras de diferencia). A toda la suma se le sumara el número de palabras que están en el entorno, multiplicado por el peso de proximidad definido. Definiendo un entorno de 1, la proximidad valida será que todas las palabras se encuentren unas seguidas de las otras.

$$w(D_i)=\sum_{j=0}^{|P|}(P_j\in D_i * K * U * T)+Prox * PProx$$

Para calcular el peso de una palabra en un documento se utiliza la fórmula TF-IDF ligeramente modificada.

$$w(P_j\in D_i)=\frac{|P_j\in D_i| * FP}{IMP(P_j)}$$

$$IMP(P_j)=\frac{|P_j\in D|}{|D|}$$

El peso de la palabra j en el documento i es igual al número de veces que aparece la palabra j en el documento i por un factor de aparición que

normalmente es 1, dividido entre la importancia de la palabra j, que se calcula como el número de veces que aparece la palabra j, dividido entre el número de documentos que hay en el corpus de búsqueda.

A todo este peso hay que aplicarle un factor de corrección propio del dominio. En nuestro caso, las páginas de revisión o las versiones antiguas, aunque contengan la misma información que las páginas normales, su importancia es menor, así pues las penalizamos aportándoles la mitad de su puntuación. De todas formas, estos resultados se pueden filtrar si se realiza una búsqueda avanzada. Esta corrección podría variar dependiendo del dominio donde se utilizara el buscador, de forma que sería fácilmente adaptable, cambiando el valor de los pesos y modificando simplemente una función, nuestro algoritmo de búsqueda se adaptaría a nuevos entornos.

6.3 Algoritmo de corrección de términos mal escritos

Una de las funcionalidades del sistema es ayudar al usuario si este ha cometido un error al escribir una palabra. Para conseguir esto, se utiliza como palabras válidas aquellas que están en los documentos indizados. Esta aproximación tiene el problema de que algunas palabras pueden estar mal escritas en el wiki, y por tanto parecer correctas para el buscador sin serlo realmente. Una alternativa podría utilizar una lista de palabras válidas de un diccionario de la lengua castellana, pero entonces no se tendría en cuenta tipos de palabras como nombres propios o siglas, que son consideradas como palabras incorrectas.

Independientemente de donde se obtenga la lista de palabras válidas, el buscador cuando no encuentra páginas con las palabras de búsqueda introducidas, sugiere palabras que conoce y que son ortográficamente similares a las introducidas. Para ello utiliza el algoritmo de K vecinos más cercanos (K-NN) [9] para escoger aquellos que más se parecen al introducido por el usuario de los ejemplos supuestamente válidos.

Esta práctica se engloba dentro de las técnicas de aprendizaje automático denominadas técnicas vagas. Se denominan así, porque no generan ningún modelo de generalización abstracta de la solución y se basan sólo en casos previos resueltos satisfactoriamente. En nuestro estudio, disponemos de una colección de palabras correctas. Si el usuario introduce una palabra no correcta, entonces el algoritmo K-NN determinará según un criterio de distancia o similitud entre instancias, cuales son las K instancias más similares. Una vez obtenidas las instancias más similares, se pueden seguir varias estrategias para decidir cual seleccionar. En nuestro caso se ha resuelto mostrar las cinco de mayor similitud ordenadas de forma ascendente.

¿Cuál es la medida de distancia correcta a utilizar?

K-NN suele utilizar la distancia euclídea. Cuando los atributos de esa instancia son valores numéricos, la distancia euclídea suele funcionar de forma correcta. Sin embargo, al tratarse de atributos simbólicos (los caracteres), la distancia euclídea no aporta suficiente información, y se requiere otra medida de distancia más específica para este dominio.

Se ha definido una medida de distancia basada en la distancia de Edición [15]. La distancia de edición se define como el *mínimo número de inserciones y borrados que transforman una cadena en la otra*.

A esta medida se le puede añadir el número de secuencias de n caracteres que aparecen en ambas palabras. La distancia implementada se basa en estos principios, pero incorpora otras medidas. Las características que se tienen en cuenta en la medida de distancia son las siguientes:

- Se comprueba si hay rotación de letras, si una letra está rotada, cuenta la distancia de rotación, es decir el número de posiciones que hay desde la letra correcta a la rotada.
- Comprueba si falta una letra. Es decir si se ha olvidado escribir una letra. Por cada letra que falte se produce una inserción.
- Comprueba si se ha añadido letras de más, es decir que es necesario borrados para dejar las palabras idénticas.
- Comprueba si una letra se ha escrito mal (se ha cambiado por otra).

Sumando estos fallos nos da la distancia entre dos palabras, la supuestamente correcta y la incorrecta.

El algoritmo funciona bastante bien y soporta casi todos los errores típicos al escribir. Si además no se tienen en cuenta ni las mayúsculas ni los acentos, el sistema facilita mucho la búsqueda al usuario, incluso de palabras que no sabe muy bien como se escriben.

6.4 Búsqueda en el Tesauro

El tesauro de términos se utiliza en varias fases del programa:

- Para obtener sinónimos de las palabras buscadas.
- Para obtener los términos relacionados con los de la búsqueda realizada.
- Para buscar palabras en el propio tesauro y obtener sus palabras relacionadas.

El tesauro está encapsulado para que sea fácilmente sustituible por otro, mediante el desarrollo de un plugin para el nuevo formato.

El tesauro que utiliza el sistema está basado en un fichero de texto. Este fichero es procesado para cargar el tesauro en la memoria del buscador, que realiza las consultas de los términos de forma que, si coincide parte de las palabras a buscar con la palabra del tesauro, se considera que se ha producido una equiparación y se muestra dicha entrada del tesauro.

Los resultados obtenidos se envían a la interfaz que construye una lista con las palabras relacionadas, permitiendo buscar más sobre ellas en el tesauro, o bien consultarlas en el wiki. Este proceso se puede ver en la Figura: 21.

Figura: 21 Búsqueda en el tesauro

6.5 Búsqueda en la Wikipedia

La búsqueda en la Wikipedia la realiza el buscador Google. El sistema simplemente pregunta por las palabras de búsqueda y le añade el término Wikipedia, recuperando siempre el primer resultado obtenido por Google. Esta funcionalidad la realiza, por simplicidad, el proceso buscador ya que este módulo tiene implementado el protocolo HTTP. El resultado obtenido por Google es procesado para extraer la primera entrada, así como el resumen, la URL y el título. El proceso de comunicaciones e interacción se muestra en la Figura: 22

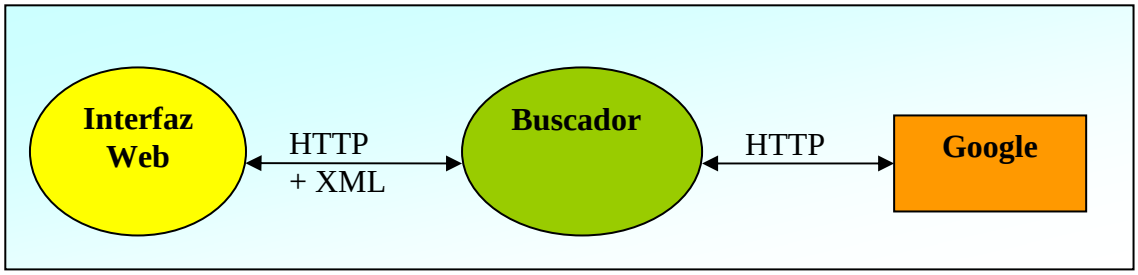

Figura: 22 Esquema de protocolos y comunicaciones para buscar en Wikipedia

6.6 La interfaz Web y sus características.

La interfaz Web está desarrollada en HTML + CSS y Javascript. Esta implementada para soportar diferentes resoluciones de pantalla sin que se deforme su contenido y está validad para su correcta visualización en los navegadores IExplorer, Firefox y Safari.

La página ha sido validada en XHTML 1.0 Transactional, así como su accesibilidad Web. Ha sido evaluada mediante TAW 3 y ha obtenido la calificación de AA en accesibilidad.

La interfaz contiene una ayuda online que permite ser consultada sin perder el estado en el que se encuentra la aplicación. Esta ayuda muestra, de forma resumida, las principales funcionalidades del sistema y como utilizarlas.

6.7 Optimizaciones en el proceso de indización

El proceso de indización está optimizado para sacar mayor rendimiento si utilizamos varias máquinas para albergar los índices, así como aprovechar las cada vez más crecientes capacidades de los procesadores multinúcleo. Las comunicaciones RMI, que no precisan respuesta, se han implementado como llamadas asíncronas para no interrumpir al proceso principal de forma innecesaria. Asimismo, cada petición realizada al buscador se ejecuta en un hilo distinto para permitir la concurrencia de varias peticiones.

En cuanto a las comunicaciones entre los índices, se han minimizado lo más posible, construyendo caches de datos temporales para no pedir información que ya se ha obtenido, así como se han agrupado las peticiones para reducir el número de conexiones que se establecen, de forma que las conexiones trasporten la mayor cantidad de datos posible. Por ejemplo, se almacenan todas las palabras que pertenecen a otros índices y se le envían juntas en vez de enviarlas a la vez que se procesan.

Estas optimizaciones permiten aprovechar las capacidades de procesado multinúcleo y permite que el sistema sea más escalable. Si se quiere indizar un sitio con un elevado número de páginas, se pueden ampliar los índices en otras máquinas y así reducir el tiempo de indización. Por lo general, para sitios Web reducidos será suficiente con instalar un índice. Al utilizar RMI es necesario que todos los índices estén ejecutándose en la misma red, para que puedan tener acceso al Manager y al registro de RMI.

6.8 Optimizaciones en el proceso de recuperación

El proceso de recuperación de los documentos se ha optimizado para permitir adaptar la herramienta a distintos entornos.

Las llamadas RMI no se han creado asíncronas, pero se han intentado minimizar a través del envío de mayor volumen de datos por conexión.

Se eliminan ciertas palabras que se consideran palabras con poca carga semántica. Estas palabras son denominadas palabras vacías. La lista de palabras vacías se ha obtenido de http://snowball.tartarus.org/ aunque el recurso ha sido adaptado, eliminado varias de ellas por considerarlas útiles. Con esta lista, se reduce el tiempo de cálculo y el tamaño de la base de datos así como se mejora el algoritmo de posicionamiento, ya que palabras como el, a, le, que no tienen carga semántica, no interfieren en la búsqueda.

Existe una opción en *búsqueda avanzada* con la que se puede definir la precisión de la búsqueda para adaptarla a las necesidades de los usuarios. Si se precisan búsquedas exhaustivas, el buscador tardará más tiempo, pero tendrá en cuenta más documentos, mientras que si se quieren resultados rápidamente sin importar tanto la calidad, se puede reducir la precisión de la búsqueda, utilizando menos documentos.

Como última optimización, se pueden eliminar las páginas de revisión y las páginas antiguas para reducir el número de elementos. Las páginas a eliminar son muy dependientes del dominio y por tanto si se adapta el buscador a otros dominios habría que modificar los criterios para determinar si una página hay que procesarla o no. Afortunadamente, el código que realiza esta tarea está encapsulado para evitar cambios innecesarios en el resto del buscador.

7 Validación y evaluación del sistema

7.1 Proceso general de validación

Se han definido y ejecutado un conjunto de pruebas para validar el funcionamiento del buscador. Los aspectos que se han validado y evaluado han sido los siguientes:

- Evaluación del algoritmo de búsqueda: estas pruebas pretenden determinar si el algoritmo de búsqueda es adecuado para el entorno en el que se está utilizando.
- Tiempos del proceso de indización: estas pruebas miden el tiempo que necesita el buscador para indizar un sitio Wiki y su capacidad de escalabilidad a Wikis de mayor tamaño.
- Tiempos del proceso de recuperación: mide el tiempo en que el buscador recupera las páginas solicitadas.
- Evaluación de la usabilidad: Mide el comportamiento del buscador ante varias peticiones simultáneas.
- Evaluación de la actualización: Mide el comportamiento del buscador cuando el número de peticiones aumenta.

El equipo en el que se han realizado los experimentos es un Turion a 1,6 GB con 1 GB de memoria RAM con una conexión a Internet de 6 Mbps. Las pruebas se han realizado con dos índices ejecutándose en la misma máquina para comprobar si la comunicación entre los índices es correcta.

Para ciertas pruebas se ha definido un sitio más pequeño y controlado, de forma que sea más sencillo comprobar que los resultados obtenidos son los correctos. El sitio está compuesto por varias páginas Web que tratan sobre temas relacionados con el contenido del wiki que estamos indizando y con las palabras del tesauro para que este tenga utilidad en la búsqueda. Para hacer ciertas pruebas se han añadido nuevas relaciones al tesauro para comprobar el funcionamiento del buscador. En la construcción de los documentos se ha seguido las mismas normas con las que están redactadas las páginas del wiki, es decir, con el título de la Web idéntico al nombre de la página, ya que los pesos están configurados para que la búsqueda se optimice para este tipo de documentos.

7.2 Evaluación del algoritmo de búsqueda

Para comprobar la idoneidad del algoritmo de búsqueda se ha utilizado el dominio de pruebas para tener mayor control sobre el corpus de la búsqueda. Se han buscado por los siguientes términos: Windows, XML, Tesauro, razonadores, sistemas operativos, XHTML. Estos términos se han escogido

debido a que existen páginas Web en el dominio de prueba que hacen referencia a los mismos.

Como se ha definido el corpus del documento de forma manual, se conoce aproximadamente la relevancia de los documentos para cada una de las búsquedas. A continuación para cada búsqueda se muestra la relevancia que se ha otorgado a los documentos del dominio. Cada documento tiene una puntuación de forma que un documento aporte tantos puntos como relevante sea. Si dos documentos son más o menos igual de relevantes se les ha otorgado la misma puntuación:

Windows:

1. Windows.html: 3p
2. Windows _95.html: 2p
3. Windows_98.html: 2p
4. Windows_2000.html: 2p
5. Windows_xp.html: 2p
6. Windows_vista.html: 2p
7. Windows_7.html: 2p
8. Sistemas_operativos.html: 1p

XML:

1. xml.html: 6p
2. editores_xml.html: 5p
3. tecnología_xml.html: 4p
4. xhtml.html: 3p
5. dom.html: 2p
6. indice.html: 1p

Tesauro:

1. Tesauro.html: 4p
2. creación_tesauros.html: 3p
3. Tipos_diccionarios.html: 2p
4. index.html: 1p

Razonadores:

1. Razonadores.html: 2p
2. indice.html: 1p

Sistemas operativos:

1. Sistemas_operativos.html: 4p:
2. Linux.html: 3p
3. Windows.html: 3p
4. Windows _95.html: 2p
5. Windows_98.html: 2p

6. Windows_2000.html: 2p
7. Windows_xp.html: 2p
8. Windows_vista.html: 2p
9. Windows_7.html: 2p
10. indice.html: 1p

XHTML:

1. XHML.html: 4p
2. Extensible_hypertext_markup_language.html: 4p
3. html.html: 3p
4. xml.html: 2p
5. indice.html: 1p

Buscadores:

1. Buscador.html: 4p
2. tipos_buscador.html: 3p
3. google.html: 2p
4. metadatos.html: 1p
5. indice.html: 1p

Metadatos:

1. metadatos.html: 2p
2. indice.html: 1p

Lenguajes de recuperación:

1. lenguajes_recuperacion.html: 2p
2. indice.html: 1p

Encabezamiento de materias:

1. encabezamiento_materias:html: 2p
2. indice.html: 1p

Para medir el resultado de las búsquedas, se va a utilizar una matriz de confusión como la siguiente:

	Recuperados	***No-recuperados***
Relevantes	***A***	***B***
No relevantes	***C***	***D***

De esta forma, podemos tener una idea de su capacidad recuperación. De forma muy compacta y resumida Llamaremos 'A' a los documentos relevantes recuperados, 'B' a los documentos relevantes no recuperados y así

sucesivamente como aparecen en el recuadro. Con estos datos, se calcula la media armónica del Precission – Recall.

Recall es la proporción de material relevante recuperado.

$$Recall = \frac{A}{A+B}$$

Precission es una medida del material recuperado realmente relevante.

$$Precission = \frac{A}{A+C}$$

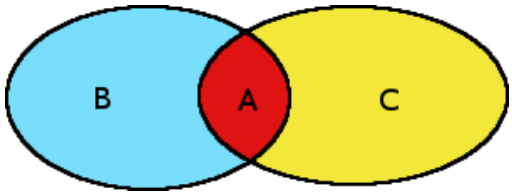

Figura: 23 Ilustración de Precisión-recall

La media armónica de ambos se calcula como:

$$media\ armonica = \frac{2}{Precission + Recall}$$

Según algunos autores como Chignell proponen modificar la medida de precisión aplicando el grado de relevancia ponderado.

A demás de estas medidas se calcula la distancia entre la ordenación realizada de forma manual y la ordenación de los documentos propuesta por el buscador. Esta distancia se calcula como el número de puntos de diferencia entre el resultado obtenido y la puntuación inicialmente definida por ese orden dado, dividido entre el sumatorio de los resultados manuales. Hay que tener en cuenta que la ordenación puede ser algo subjetiva sobre todo en los documentos menos relevantes, por lo que a demás de la ordenación, tendremos en cuenta si el documento más relevante ha sido posicionado en los primeros puestos.

La ordenación es un porcentaje que indica el grado de desordenación que se produce en el sistema. El grado máximo sería 1 que resultaría una consulta con ningún resultado relevante.

$$ordenacion = \frac{\sum_{i=0}^{N} |PuntuacionDocumentoObtenido_i - PuntuacionManual_i|}{\sum_{i=0}^{n} PuntuacionManual_i}$$

Los resultados de las pruebas con las consultas realizadas se muestran en los siguientes apartados.

7.2.1 Búsqueda 1: Windows:

Para la búsqueda "Windows" la ordenación obtenida es:

1. windows.html: 3p
2. windows _98.html: 2p
3. windows _95.html: 2p
4. windows _7.html: 2p
5. windows _xp.html: 2p
6. windows _vista.html: 2p
7. windows _2000.html: 2p
8. sistemas_operativos.html: 1p

Matriz de confusión:

	Recuperados	***No-recuperados***
Relevantes	***8***	***0***
No relevantes	***0***	***20***

La primera vez describiremos paso a paso los cálculos de la ordenación:

Windows 3p – Windows 3p = 0.
Windows_p5 2p – Windows 95 2p = 0… y así sucesivamente

Medidas	
Precission	***1***
Recall	***1***
Media	***1***
Precision-chignell	***0,66***
Ordenación	***0***
Puesto 1º	***1***
Puesto 2º	***3***

7.2.2 Búsqueda 2: XML:

Ordenación obtenida:

1. editores_xml.html: 5p
2. tecnologia_xml.html: 4p
3. xml.html: 6p
4. dom.html: 2p
5. xhtml.html: 3p
6. indice.html: 1p

Matriz de confusión:

	Recuperados	*No-recuperados*
Relevantes	*6*	*0*
No relevantes	*0*	*22*

Medidas	
Precission	*1*
Recall	*1*
Media	*1*
Precision-chignell	*0,58*
Ordenación	*6/21 = 0,28*
Puesto 1º	*3*
Puesto 2 º	*1*

7.2.3 Búsqueda 3: Tesauro:

Ordenación obtenida:

1. tesauro.html: 4p
2. creación_tesauro.html: 3p
3. encabezamiento_materias.html : 0p
4. tipos_diccionarios.html: 2p
5. indice.html: 1p
6. razonadores.html: razonadores 0p

Matriz de confusión:

	Recuperados	*No-recuperados*
Relevantes	*4*	*0*
No relevantes	*2*	*22*

Medidas	
Precission	***0,83***
Recall	***1***
Media	***1,09***
Precision-chignell	***0,41***
Ordenación	***4/10 = 0,4***
Puesto 1º	***1***
Puesto 2º	***2***

7.2.4 Búsqueda 4: Razonadores:

Ordenación obtenida:

1. razonadores.html: 2p
2. indice.html: 1p

Matriz de confusión:

	Recuperados	*No-recuperados*
Relevantes	***2***	***0***
No relevantes	***0***	***26***

Medidas	
Precission	***1***
Recall	***1***
Media	***1***
Precision-chignell	***0,5***
Ordenación	***0***
Puesto 1º	***1***
Puesto 2º	***2***

7.2.5 Búsqueda 5: Sistemas operativos:

Ordenación obtenida:

1. sistemas_operativos.html: 4p
2. linux.html: 3p
3. Windows.html: 3p
4. Windows_98.html: 2p
5. Windows_vista.html: 2p

6. Windows_97.html: 2p
7. Windows_7.html: 2p
8. indice.html: 1p

Matriz de confusión:

	Recuperados	*No-recuperados*
Relevantes	*8*	*2*
No relevantes	*0*	*18*

Medidas	
Precission	*1*
Recall	*0,8*
Media	*1,11*
Precision-chignell	*0,37*
Ordenación	*0*
Puesto 1º	*1*
Puesto 2º	*2*

7.2.6 Búsqueda 6: XHTML:

Ordenación obtenida:

1. xhtml.html: 4p
2. html.html: 3p
3. extensible_hypertext_markup_language.html: 4p
4. dom.html: 0p
5. editors_xml.html: 0p
6. tecnologia_xml.html: 0p
7. indice.html: 1p

Matriz de confusión:

	Recuperados	*No-recuperados*
Relevantes	*4*	*1*
No relevantes	*3*	*20*

Medidas	
Precission	***0,57***
Recall	***0,8***
Media	***1,46***
Precision-chignell	
Ordenación	***6/14 = 0,42***
Puesto 1º	***1***
Puesto 2º	***3***

7.2.7 Búsqueda 7: Buscadores:

Ordenación obtenida:

1. tipos_buscador.html: 3p
2. Buscador.html: 4p
3. google.html: 2p
4. metadatos.html: 1p
5. indice.html: 1p

Matriz de confusión:

	Recuperados	*No-recuperados*
Relevantes	***5***	***0***
No relevantes	***0***	***0***

Medidas	
Precission	***1***
Recall	***1***
Media	***1***
Precision-chignell	***0,55***
Ordenación	***0,18***
Puesto 1º	***2***
Puesto 2º	***1***

7.2.8 Búsqueda 8: Metadatos:

Ordenación obtenida:

1. metadatos.html: 2p
2. indice.html: 1p

Matriz de confusión:

	Recuperados	*No-recuperados*
Relevantes	***2***	***0***
No relevantes	***0***	***0***

Medidas	
Precission	***1***
Recall	***1***
Media	***1***
Precision-chignell	***0,5***
Ordenación	***0***
Puesto 1º	***1***
Puesto 2º	***2***

7.2.8 Búsqueda 9: Lenguajes de recuperación:

Ordenación obtenida:

1. lenguajes_recuperacion.html: 2p
2. encabezamiento_materiales.html: 0p
3. indice.html: 1p

Matriz de confusión:

	Recuperados	*No-recuperados*
Relevantes	***2***	***0***
No relevantes	***1***	***0***

Medidas	
Precission	***0,66***
Recall	***1***
Media	***1,20***
Precision-chignell	***0,33***
Ordenación	***0,61***
Puesto 1º	***1***
Puesto 2º	***3***

7.2.8 Búsqueda 10: Encabezamiento de materiales

Ordenación obtenida:

1. encabezamiento_materias.html: 2p
2. indice.html: 2p

Matriz de confusión:

	Recuperados	*No-recuperados*
Relevantes	***2***	***0***
No relevantes	***0***	***0***

Medidas	
Precission	***1***
Recall	***1***
Media	***1***
Precision-chignell	***0,5***
Ordenación	***0***
Puesto 1º	***1***
Puesto 2º	***2***

7.3 Comparativa con el buscador del wiki.

Para realizar la comparativa con el buscador del wiki, al no poder conocer cuál es la precisión y el recall, haremos las siguientes suposiciones:

Por búsqueda tendremos en cuenta los 10 primeros resultados. Puntuaremos con 4 puntos si ha obtenido un documento muy relevante, 3 si no ha sido tan relevantes y así sucesivamente. Se sumarán los puntos y se dividirán entre 10 * 4, de forma que tendremos la medida de precisión.

El buscador del wiki no soporta palabras sin acentos. Es decir, si una palabra tiene acento y en la búsqueda no se pone el acento, no encuentra los resultados, así pues no tendremos en cuenta esta limitación.

Además calcularemos la medida de ordenación definida en el apartado anterior pero como no disponemos de la ordenación correcta, suponemos que la ordenación correcta de los documentos que se recuperan entre las dos búsquedas, la del buscador del proyecto y el buscador del wiki.

7.3.1 Búsqueda 1: Wiki.

Resultado del buscador del wiki:

1. doku.php?id=que_es_xhtml: 4p
2. doku.php?id=xhtml: 3p
3. doku.php?id=como_funciona&s=xhtml: 3p
4. doku.php?id=xforms&s=xhtml: 1p
5. doku.php?id=diferencias_con_html&s=xhtml: 3p
6. doku.php?id=que_es_xhtml&s=xhtml: 0p
7. doku.php?id=que_es_xhtml&s=xhtml: 0p
8. doku.php?id=mas_informacion&s=xhtml: 2p
9. doku.php?id=bbcode_y_slip&s=xhtml: 1p
10. doku.php?id=css&s=xhtml: 1p

Total: 17 / 40 = 0,42
Ordenación: 6 / 20 = 0,3

Buscador del proyecto:

1. doku.php?id=que_es_xml: 4p
2. doku.php?id=*XHTML: 3p*
3. doku.php?id=como_funciona: 3p
4. doku.php?id=diferencias_con_html: 3p
5. doku.php?id=mas_informacion: 2p
6. doku.php?id=xlink: 1p
7. oku.php?id=wiki:syntax: 1p
8. doku.php?id=wml: 1p
9. doku.php?id=xsl: 1p
10. doku.php?id=xforms: 1p

Total: 20 / 40 = 0,5
Ordenación: 0 / 20 = 0

7.3.2 Búsqueda 2: Tesauro.

Resultado del buscador del wiki:

1. doku.php?id=creacion_de_tesauros: 3p
2. doku.php?id=diferencias_entre_tesauros_ontologias_y_topicmaps: 3p
3. doku.php?id=edicion_y_mantenimiento_de_tesauros: 4p
4. doku.php?id=ejemplo_practico_de_creacion_de_un_tesauro: 4p
5. doku.php?id=herramientas_para_crear_tesauros: 3p
6. doku.php?id=que_es_un_tesauro: 4p
7. doku.php?id=tesauros: 4p
8. doku.php?id=visualizacion_de_tesauros: 2p
9. doku.php?id=xml:dtds_y_esquemas_para_tesauros: 2p

Total: 29 / 40 = 0,725
Ordenación: 7 / 30 = 0,23

Buscador del proyecto:

1. doku.php?id=edicion_y_mantenimiento_de_*tesauros: 4p*
2. doku.php?id=creacion_de_*tesauros: 3p*
3. doku.php?id=xml:dtds_y_esquemas_para_*tesauros: 2p*
4. doku.php?id=*tesauros: 4p*
5. doku.php?id=ejemplo_practico_de_creacion_de_un_*tesauro 4p*
6. doku.php?id=diferencias_entre_*tesauros*_ontologias_y_topicmaps 3p
7. doku.php?id=herramientas_para_crear_*tesauros: 3p*
8. doku.php?id=visualizacion_de_*tesauros: 2p*
9. doku.php?id=que_es_un_*tesauro: 4p*
10. doku.php?id=skos: 1p

Total: 30/ 40 = 0,75
Ordenación: 6 / 30 = 0,2

7.3.3 Búsqueda 3: XML.

Resultado del buscador del wiki:

1. *doku.php?id=3.5._rdf_xml: 0p*
2. *doku.php?id=3.5_rdf_xml: 1p*
3. *doku.php?id=alternativas_a_xml: 2p*
4. *doku.php?id=xml-fo: 2p*
5. *doku.php?id=xml-xquery: 2p*
6. *doku.php?id=xml:buscadores_de_documentos_xml: 2p*
7. *doku.php?id=xml:editores_xml_dtd_y_xml_schemas: 3p*
8. *doku.php?id=xml:introduccion:estructura_de_un_fichero_xml: 3p*
9. *doku.php?id=xml:introduccion:mas_sobre_xml: 3p*
10. *doku.php?id=xml:introduccion:ventajas_e_inconvenientes_de_xml: 3p*

Total: 21/40 = 0,525
Ordenación: 14 / 30 = 0,46

Buscador del proyecto:

1. doku.php?id=xml:tecnologias_*XML: 4p*
2. doku.php?id=xml:introduccion:ventajas_e_inconvenientes_de_*XML: 3p*
3. doku.php?id=xml:validadores_de_dtds_esquemas_y_documentos_*XML: 3p*
4. doku.php?id=xml:editores_de_*XML*_dtd_y_*XML*_schema:aproximacion_a_los_tres_lenguajes: 4p
5. doku.php?id=xml:dtds_y_esquemas_para_imagenes: 2p
6. doku.php?id=xml:dtds_y_esquemas_para_topic_maps: 2p
7. doku.php?id=xml:editores_de_*XML*_dtd_y_*XML*_schema:caracteristicas_deseables_de_un_editor: 2p
8. doku.php?id=xml:editores_*XML*_dtd_y_*XML*_schemas: 3p
9. doku.php?id=xml:introduccion:mas_sobre_*XML: 3p*
10. doku.php?id=xml:introducción: 4p

Total: 30/ 40 = 0,75
Ordenación: 9 / 30 = 0,3

7.3.4 Búsqueda 4: Lenguajes de recuperación.

Resultado del buscador del wiki:

1. *doku.php?id=lenguajes_de_recuperacion.: 4p*
2. *doku.php?id=lenguajes_de_recuperacion_con_topic_maps: 4p*

Total 8/ 40 = 0,2
Ordenación: 7 / 15 = 0,46

Buscador del proyecto:

1. doku.php?id=*lenguajes_de_recuperacion: 4p*
2. doku.php?id=*lenguajes_de_recuperacion*_con_topic_maps : 4p
3. doku.php?id=metadatos_para_imagenes: 2p
4. *doku.php?id=dublin_core: 1p*
5. *doku.php?idx=wiki: 1p*
6. *doku.php?id=xml-xquery: 1p*
7. *doku.php?id=topic_maps: 1p*
8. *doku.php?id=rdf_y_owl: 1p*
9. *doku.php?id=inicio&idx=talk: 0p*

Total 15 / 40 = 0,375
Ordenación: 0 / 15 = 0

7.3.5 Búsqueda 5: Ontología.

Resultado del buscador del wiki:

1. *doku.php?id=diferencias_entre_tesauros_ontologias_y_topicmaps: 3p*

2. *doku.php?id=edicion_y_mantenimiento_de_ontologias: 4p*
3. *doku.php?id=herramientas_para_crear_ontologias: 4p*
4. *doku.php?id=lenguajes_para_expresar_ontologias: 0p (pagina vacía)*
5. *doku.php?id=ontologias: 4p*
6. *doku.php?id=fusion_y_mapeados&s=ontologías: 3p*
7. *doku.php?id=herramientas_para_crear_ontologias&s=ontologías: 3p*
8. *doku.php?id=edicion_y_mantenimiento_de_ontologias&s=ontologías: 3p*
9. *doku.php?id=visualizacion&s=ontologías: 3p*
10. *doku.php?id=jena&s=ontologías: 2p*

Total: 29/ 40 = 0,725
Ordenación: 5 / 29 = 0,17

Buscador del proyecto:

1. doku.php?id=herramientas_para_crear_*ontologias: 4p*
2. doku.php?id=edicion_y_mantenimiento_de_*ontologias: 4p*
3. doku.php?id=*ontologias: 4p*
4. doku.php?id=*ontologia: 3p*
5. doku.php?id=diferencias_entre_tesauros_*ontologias*_y_topicmaps: 3p
6. doku.php?id=fusion_y_mapeados: 3p
7. doku.php?id=lenguajes_para_expresar_*ontologias: 0p (vacia)*
8. doku.php?id=visualizacion: 3p
9. doku.php?id=jena: 2p
10. doku.php?id=herramientas_relacionadas_con_*ontologias: 3p*

Total: 29/ 40 = 0,725
Ordenación: 5 / 29= 0,17

7.3.6 Búsqueda 6: Hojas de estilo

Resultado del buscador del wiki:

1. *doku.php?id=css&s=hojas estilo: 4p*
2. *doku.php?id=dom&s=hojas estilo: 2p*
3. *doku.php?id=xsl&s=hojas%20estilo: 3p*
4. *doku.php?id=xml:editores_de_xml_dtd_y_xml_schema:aproximacion_a_los_tres_lenguajes&s=hojas estilo: 1p*
5. *doku.php?id=xml:tecnologias_xml&s=hojas estilo: 1p*
6. *doku.php?id=que_es_xhtml&s=hojas estilo 1p*

Total: 12/40 = 0,3
Ordenación: 4 / 14 = 0,28

Buscador del proyecto:

1. doku.php?id=css: 4p
2. doku.php?id=xsl: 3p
3. doku.php?id=dom: 2p

4. feed.php: 0p
5. doku.php?id=xml:editores_*de*_xml_dtd_y_xml_schema:aproximacion_a_los_tres_lenguajes: 1p
6. doku.php?id=xml:tecnologias_xml: 1p
7. doku.php?id=xml:introduccion:estructura_*de*_un_fichero_xml: 1p
8. doku.php?id=que_es_xhtml: 1p
9. xml:editores_*de*_xml_dtd_y_xml_schema:caracteristicas_deseables_*de*_un_editor: 1p

Total: 14/40 = 0,35
Ordenación: 1 / 14 = 0,07

7.3.7 Búsqueda 7: Sindicación.

Resultado del buscador del wiki:

1. *doku.php?id=sindicacion&s=sindicación: 4p*
2. *doku.php?id=rss&s=sindicación: 3p*
3. *doku.php?id=foaf&s=sindicación: 3p*
4. *doku.php?id=atom&s=sindicación: 3p*
5. *doku.php?id=metadatos&s=sindicación: 2p*
6. *doku.php?id=metadatos_para_imagenes&s=sindicación : 2p*

Total: 17/40 = 0,425
Ordenación: 2 / 19 = 0,11

Buscador del proyecto:

1. doku.php?id=*sindicacion: 4p*
2. doku.php?id=rss: 3p
3. doku.php?id=atom: 3p
4. doku.php?id=fofa: 3p
5. doku.php?idx=wiki: 1p
6. doku.php?id=metadatos_para_imagenes: 2p
7. doku.php?id=metadatos: 2p
8. doku.php?id=inicio&idx=xml:introducción: 1p

Total: 19/ 40 = 0,475
Ordenación: 2 / 19 = 0,11

7.3.8 Búsqueda 8: Recuperación de la información.

Resultado del buscador del wiki:

1. *doku.php?id=metadatos_para_imagenes&s=recuperación información: 3p*
2. *doku.php?id=metadatos&s=recuperación información: 3p*
3. *doku.php?id=dublin_core&s=recuperación información: 3p*

4. *doku.php?id=lenguajes_de_recuperacion_con_topic_maps&s=recuperación información: 4p*
5. *doku.php?id=metadatos_para_audio&s=recuperación información: 3p*
6. *doku.php?id=topic_maps&s=recuperación información: 2p*
7. *doku.php?id=ventajas_del_trabajo_con_razonadores._alternativas_a_su_empleo&s=recuperación información: 2p*
8. *doku.php?id=xml-xquery&s=recuperación información: 1p*
9. *doku.php?id=xml-xquery&s=recuperación información: 1p*
10. *doku.php?id=aspectos_positivos_y_carencias_de_los_topic_maps&s=recuperación información: 1p*

Total: 23/ 40 = 0,525
Ordenación: 5 / 24 = 0,28

Buscador del proyecto:

1. doku.php?id=lenguajes_*de_recuperacion: 4p*
2. doku.php?id=lenguajes_*de_recuperacion*_con_topic_maps: 4p
3. doku.php?id=metadatos_para_imagenes: 3p
4. doku.php?id=dublin_core: 3p
5. doku.php?id=metadatos: 3p
6. doku.php?id=topic_maps: 2p
7. doku.php?id=xml-xquery: 1p
8. doku.php?id=metadatos_para_audio: 3p
9. doku.php?id=rdf_y_owl: 1p
10. feed.php: 0p

Total: 24/40 = 0,6
Ordenación: 4 / 24 = 0,16

7.3.9 Búsqueda 9: Scalable Vector Graphics.

Resultado del buscador del wiki:

1. *doku.php?id=svg&s=scalable vector Graphics: 4p*
2. *doku.php?id=xml:svg&s=scalable vector graphics: 4p*
3. *doku.php?id=xforms&s=scalable vector graphics: 1p*

Total: 9/ 40 = 0,225
Ordenación: 3 / 12 = 0,25

Buscador del proyecto:

1. doku.php?id=svg: 4p
2. doku.php?id=xml:svg: 4p
3. doku.php?id=xforms: 1p
4. doku.php?id=inicio&idx=xml:introducción: 0p
5. doku.php?id=metadatos_para_imagenes: 1p
6. doku.php?id=xml:index: 1p
7. doku.php?id=fofa: 1p

Total: 12/40 = 0,3
Ordenación: 2 / 12 = 0,16

7.3.10 Búsqueda 10: Extensible Hipertext Markup Language.

Resultado del buscador del wiki:

1. *doku.php?id=metadatos_para_audio&s=extensible hipertext markup language: 1p*
2. *doku.php?id=wml&s=extensible hipertext markup language: 1p*
3. *doku.php?id=xslt&s=extensible hipertext markup language: 1p*
4. *doku.php?id=yaml&s=extensible hipertext markup language: 1p*
5. *doku.php?id=dublin_core&s=extensible hipertext markup language: 1p*
6. *doku.php?id=xml:tecnologias_xml&s=extensible hipertext markup language: 1p*
7. *doku.php?id=mas_informacion&s=extensible hipertext markup language 1p*
8. *doku.php?id=xsl&s=extensible hipertext markup language: 1p*
9. */doku.php?id=xml:buscadores_de_documentos_xml&s=extensible hipertext markup language: 1p*
10. *doku.php?id=xml:mathml&s=extensible hipertext markup language: 1p*

Total: 10/ 40=0 ,35
Ordenación: 16 / 26 = 0,61

Buscador del proyecto:

1. doku.php?id=HTML: 3p
2. doku.php?id=diferencias_con_html: 4p
3. doku.php?id=xhtml: 4p
4. doku.php?id=que_es_xhtml: 4p
5. doku.php?id=dom: 2p
6. doku.php?id=como_funciona: 3p
7. doku.php?id=css: 2p
8. doku.php?id=mas_informacion: 2p
9. doku.php?id=bbcode_y_slip: 1p
10. doku.php?id=xslt: 1p

Total: 26/ 40= 0,65
Ordenación: 4 / 26 = 0,15

En la Figura: 24 podemos comparar gráficamente los resultados de la ordenación y en la Figura: 25 se puede apreciar los resultados de la precisión.

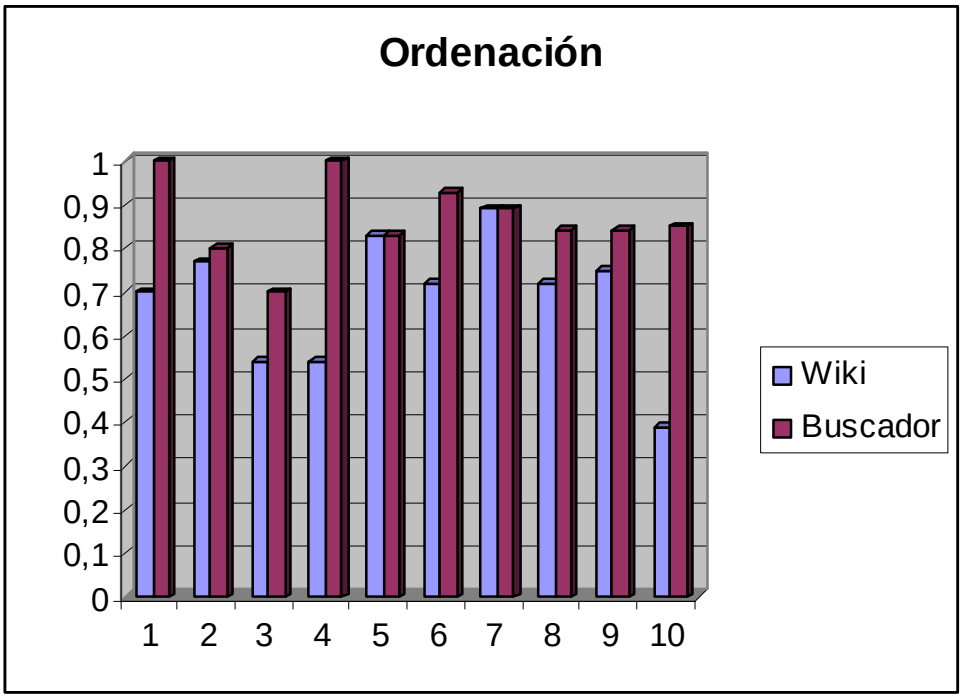

Figura: 24 Comparativa de la métrica de ordenación entre el buscador del wiki y el proyecto.

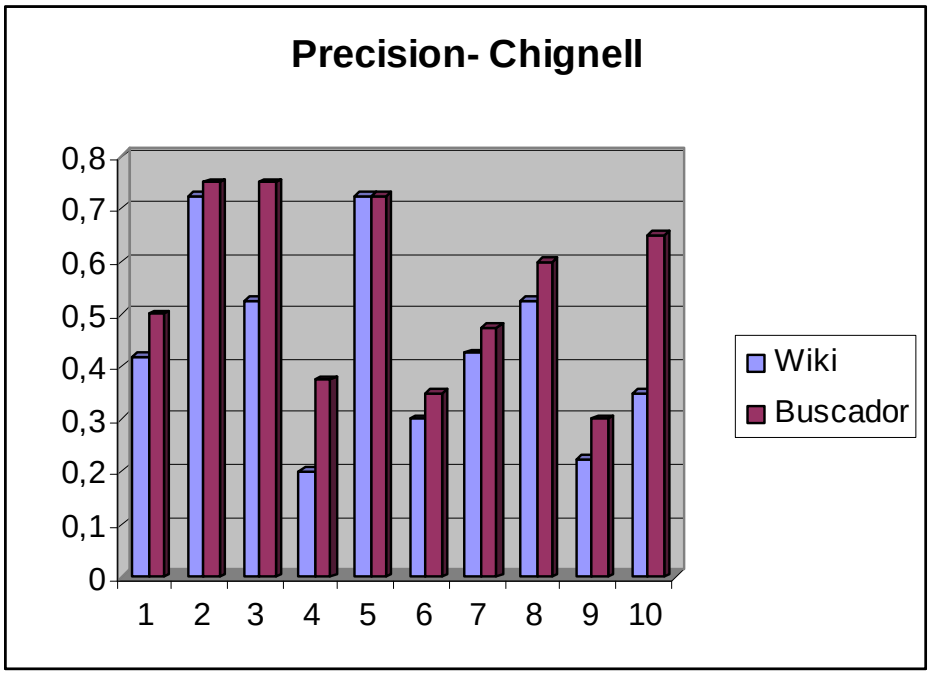

Figura: 25 Comparativa de la precisión entre el buscador del wiki y el del proyecto.

Los resultados obtenidos son iguales o ligeramente mejores al del buscador del wiki, en los casos en los que se utilizan los sinónimos del tesauro (Ejemplos 1, 9 y 10) los resultados obtenidos por el buscador son claramente mejores a los obtenidos por el buscador del wiki, ya que este buscador no tiene en cuenta los sinónimos, lo que permite a nuestro sistema una mayor capacidad semántica. Se ha podido apreciar que la página *feed.php* no contiene información relevante,

por lo que se puede incluir en el filtro de revisiones para mejorar aún más las búsquedas.

7.4 Evaluación de la utilización del Tesauro.

En las búsquedas 1 y 6 del apartado 7.2 así como la búsqueda 10 del apartado 7.3, podemos apreciar la utilidad del tesauro. Windows tiene como sinónimos Windows Vista y XP, lo que ha favorecido su recuperación. En este caso al contener la palabra Windows no se aprecia la mejora, pero en el ejemplo 6 el resultado del tesauro se ve potenciado. En el archivo extensible_hypertext_markup_language.html no aparece ni una sola vez la palabra XHTML al igual que en HTML.html. Sin embargo ambos archivos han sido recuperados por sus sinónimos. Al igual que editores_xml.html y tecnología_xml.html que contenían la palabra HTML y que inicialmente no han sido clasificados como relevantes.

En la comparativa con el buscador que incorpora el wiki, se ve claramente en la búsqueda 10 como los sinónimos ayudan de forma significativa a la búsqueda y amplía la capacidad del buscador, permitiendo mostrar resultados con los que a priori no contábamos si usamos una búsqueda estándar.

7.5 Tiempos del proceso de indización

Se han realizado varias pruebas del proceso de indización para medir el tiempo que tarda en ser indizada.

En concreto, se han realizado 5 experimentos. El tiempo medio de cada experimento ha sido de 3 horas y 16 minutos. El número de páginas procesadas han sido 13289, por lo que el promedio de páginas por minuto han sido de 70 páginas por minuto.

El tiempo mínimo ha sido de 2 horas y 50 minutos y el máximo de 3 horas y 20 minutos. La tabla con los tiempos obtenidos se muestra a continuación.

Tiempos de indización en minutos	
Experimento 1	***190***
Experimento 2	***185***
Experimento 3	***201***
Experimento 4	***194***
Experimento 5	***170***
Experimento 6	***179***

Tabla 5 Tiempos de indización en minutos.

7.6 Tiempos del proceso de recuperación

Se han medido varios tiempos de recuperación de la información en las 5 diferentes medidas de precisión de que dispone el buscador: muy rápida, rápida, normal, lenta y exhaustiva. En cada medida de precisión se han realizado 5 consultas de términos frecuentes en el vocabulario del wiki objeto de la búsqueda. De cada una se han realizado 5 experimentos y se ha calculado la media. Los términos a buscar han sido: Tesauro, XHTML, sistema de organización del conocimiento, metadatos y Web semántica

En las tablas siguientes observamos los tiempos de recuperación obtenidos.

Precisión: muy rápida		
Tesauro	*2,28 – 1,07 – 2,51 – 0,88 – 1,12*	*1,57*
XHTML	*4,86 – 1,7 – 2,04 – 3,4 – 3,16*	*3,02*
sistema de organización del conocimiento	*1,04 – 1,15 – 1,13 – 1,16 – 1,2*	*1,14*
metadatos	*1,75 – 3,26 – 0,85 – 3,03 – 0,62*	*1,92*
Web semántica	*7,16 – 1,18 – 1,02 – 2,19 – 1,13*	*2,56*

Precisión: rápida		
Tesauro	*0,88 – 1,33 – 1 – 1,2 – 1,23*	*1,13*
XHTML	*3,3 – 2,43 – 2,76 – 4,85 – 4,5*	*3,56*
sistema de organización del conocimiento	*1,6 – 1,62 – 2.05 – 1,78 – 1,78*	*1,76*
metadatos	*2,29 – 1,88 – 1,13 – 1,92 – 1,07*	*1,66*
Web semántica	*4,06 – 1,35 – 1,37 – 1,16 – 1,59*	*1,9*

Precisión: normal		
Tesauro	*1,4 – 1,25 – 1,04 – 1,11 – 1,28*	*1,27*
XHTML	*3,36 – 3,39 – 3,28 – 3,51 – 3,84*	*3,48*
sistema de organización del conocimiento	*2,8 – 1,97 – 1,61 – 1,95 – 3,18*	*2,31*
metadatos	*0,8 – 1,34 – 1,06 – 2,32 – 0,82*	*1,34*
Web semántica	*1,92 – 1,96 – 1,78 – 1,62 – 1,16*	*1,68*

Precisión: lenta		
Tesauro	*1,4 – 2,47 – 2,76 – 1,56 – 1,46*	*1,93*
XHTML	*8,85 – 4,8 – 5,43 – 5,01 – 7,11*	*6,24*
sistema de organización del conocimiento	*3,97 – 2,74 – 3,96 – 2,88 – 4,54*	*3,62*
metadatos	*3,18 – 1,43 – 1,15 – 1,18 – 1,57*	*1,7*
Web semántica	*3,49 – 2,56 – 3,59 – 3,07 – 2,48*	*3,03*

Precisión: exhaustiva		
Tesauro	***3,37 – 2,14 – 2,12 – 2,14, 3,34***	***2,62***
XHTML	***6,74 – 7,21 – 6,61 – 7,91 – 7,7***	***7,23***
sistema de organización del conocimiento	***8,42 – 3,82 – 5,3 – 4,17 – 5,27***	***5,39***
metadatos	***1,66 – 1,37 – 2,95 – 2,65 – 1,85***	***2,09***
Web semántica	***11,28 – 10,8 – 6,01 – 5,94 – 5,87***	***7,98***

7.7 Evaluación de usabilidad.

Para realizar la evaluación de usabilidad, se ha utilizado a un grupo de 20 personas al que se le han pedido realizar una serie de acciones en la interfaz del buscador, midiendo tanto los errores que han cometido, como si han utilizado o no la ayuda, y el tiempo en el que lo han realizado.

Antes de utilizar el buscador se explicó brevemente sus características. Se informó a los usuarios que el sistema realizaba búsquedas con sinónimos extraídos de un Tesauro, que podía buscar en la Web, en el tesauro y en la Wikipedia; que además, el buscador, ofrecía sugerencias de búsqueda por palabras relacionadas y que se disponía de ayuda en línea para poder utilizar si fuera necesario.

Las preguntas que se les han formulado han sido las siguientes:

- Buscar en el wiki: XHTML
- Buscar en el tesauro: XHTML
- Buscar en la Wikipedia: XHTML
- Buscar de forma exhaustiva en el wiki : Ontología
- Buscar en el wiki: *Ontología* sin la palabra *herramientas*
- Buscar en el wiki: 'lenguajes para expresar ontologías' literalmente.
- Buscar en el wiki: XHTML o HTML.

Los factores que se midieron han sido:

- El **tiempo máximo** medido en segundos, que es el número de segundos que tardó en encontrar el último usuario la búsqueda planteada.
- **Número fallos**: número de personas que no consiguió realizar la búsqueda.
- **Número aciertos sin ayuda**: es el número de usuarios que realizaron correctamente la búsqueda sin utilizar la ayuda.

- **Número aciertos con ayuda**: es el número de usuarios que realizaron correctamente la búsqueda utilizando la ayuda para realizarla.

Los resultados obtenidos han sido agrupados, de forma resumida, en las siguientes tablas:

Buscar en el wiki: XHTML	
Tiempo máximo	***80 s***
Fallos	***0***
Aciertos sin ayuda	***20***
Aciertos con ayuda	***0***

Buscar en el tesauro: XHTML	
Tiempo máximo	***95 s***
Fallos	***2***
Aciertos sin ayuda	***18***
Aciertos con ayuda	***0***

Buscar en la Wikipedia: XHTML	
Tiempo máximo	***74 s***
Fallos	***1***
Aciertos sin ayuda	***19***
Aciertos con ayuda	***0***

Buscar de forma exhaustiva en el wiki : Ontología	
Tiempo máximo	***255 s***
Fallos	***3***
Aciertos sin ayuda	***13***
Aciertos con ayuda	***4***

Buscar en el wiki: Ontología sin la palabra herramientas	
Tiempo máximo	***305 s***
Fallos	***4***
Aciertos sin ayuda	***14***
Aciertos con ayuda	***2***

Buscar en el wiki: lenguajes para expresar ontologías literalmente	
Tiempo máximo	*190 s*
Fallos	*2*
Aciertos sin ayuda	*17*
Aciertos con ayuda	*1*

Buscar en el wiki: XHTML o HTML	
Tiempo máximo	*102 s*
Fallos	*1*
Aciertos sin ayuda	*19*
Aciertos con ayuda	*0*

Par tener una medida estadística de estos resultados, hemos calculado la media de todas las consultas para cada uno de los criterios contemplados, obteniendo los siguientes resultados en tanto por 1.

Media	
Tiempo máximo	*157, 29 s*
Fallos	*0,0925*
Aciertos sin ayuda	*0,8575*
Aciertos con ayuda	*0,05*

Al finalizar las pruebas se pidió a los usuarios que propusieran algunas mejoras a la Web que se resumirán en cuatro, de las cuales dos se han implementado la solución en dos de ellas:

- La dificultad de saber cuándo se acciona el botón para buscar entre el wiki, el Tesauro y la Wikipedia. Esto produjo un cambio en la interfaz para realzar que botón (radio button) pertenecía a que etiqueta separando las etiquetas con una línea de color azul como se muestra en la imagen:

- La claridad de la ayuda: La lectura de la ayuda no les ayudó mucho en algunas ocasiones, por lo que se redactó de nuevo la ayuda.
- La animación que se ejecuta para indicar que se está realizando el proceso de búsqueda da un salto al terminar y pasar a la imagen del logo estático. Este problema se ha intentado reducir, minimizando dicho salto, pero en el diseño final aún lo comete, aunque en menor medida que anteriormente.
- El diseño de la página es muy simple. En parte el objetivo del diseño es que resulte simple para agilizar el proceso de carga de la página, por lo que no hemos atendido a esta sugerencia.

7.8 Evaluación de actualización del índice

Se han realizado varias pruebas para comprobar que el índice se actualiza de forma automática cuando hay cambios en las páginas. Para ello hemos utilizado el sitio Web destinado a pruebas ya que la detección de cambios en los documentos es más sencilla en este entorno cerrado que en el wiki a indizar.

Para realizar esta prueba se ha modificado, añadiendo un espacio en blanco, la página *sistemas_operativos.html* y una palabra a la página *windows.html*. Además, se ha ejecutado los índices con la opción *update* que mantiene la funcionalidad de búsqueda añadiendo la posibilidad de modificar los índices si los documentos cambian. Mientras que el sistema atendía a nuestras peticiones, cada cierto tiempo, accedía a una de las páginas Web indexadas para comprobar si había cambiado. Para hacer esto, el índice comprueba que las palabras de la Web sean las mismas. Si hay alguna diferencia entre la página adquirida y la página almacenada, elimina todas las referencias a esta página en todos los índices, e inserta la nueva página.

En las pruebas realizadas el sistema modificó solamente la página a la que se le cambió el contenido, es decir a *windows.html*, no contemplando el cambio realizado en *sistemas_operativos.html*. Después se realizaron cambios en la página tesauro.html mientras el sistema estaba en funcionamiento, detectando dicho cambio sin reiniciar el sistema.

Posteriormente se movió el fichero *dom.html* a otra ubicación para que al acceder a dicha página, el índice no la pudiera recuperar. Al suceder esto, el índice eliminó la referencia de dicha página, tanto de la base de datos del índice que detectó el fallo, como del resto de índices. Después se volvió a colocar la página en su dirección correcta y cuando el índice revisó todas las páginas y volvió a encontrar un enlace del que no disponía referencias, lo incorporó a la base de datos.

8 Conclusiones

Los objetivos que se habían marcado al inicio del proyecto se han cumplido satisfactoriamente. Se ha diseñado y construido un buscador para el wiki de uno de los grupos de investigación sobre Ingeniería del Software y Recuperación de Información del Departamento de Informática de la Universidad Carlos III de Madrid. El buscador tiene como principales características:

- Supera en prestaciones al buscador que incorpora la herramienta DokuWiki, con la que se ha desarrollado el wiki, tanto en precisión de búsqueda como en capacidad para ordenar los resultados según su relevancia.
- Proporciona un árbol de sugerencias extraído del tesauro, que ayuda a los usuarios a la navegación y el refinamiento de sus búsquedas.
- Corrige ciertos errores del buscador de DokuWiki, como la omisión de errores en la acentuación, corrección de palabras mal escritas, etc.
- Permite consultar un término en la Wikipedia.
- Tiene una interfaz más intuitiva y fácil de utilizar, proporcionando ayuda en la búsqueda booleana.
- Es un buscador escalable y fácilmente reutilizable en otros entornos
- Gracias a su diseño basado en plugins, es fácilmente mantenible y ampliable.
- Tiene mayor capacidad semántica en la búsqueda gracias al tesauro y a los sinónimos.
- Los tiempos de indización y búsqueda han sido optimizados sin necesidad de cargar completamente los índices en la memoria RAM, utilizando las capacidades multihilo de los nuevos procesadores y el uso de caches para reducir comunicaciones, lo que favorece su escalabilidad.
- Permite buscar dentro del propio tesauro las relaciones de algún término.

9 Trabajos futuros

Este proyecto plantea ciertos aspectos que pueden ser mejorados y ampliados en el futuro para aumentar las posibilidades la herramienta y ampliar sus dominios de utilización. A continuación describimos una lista de mejoras que se podrían incluir en el presente proyecto:

- Incorporar un sistema de *clustering* (agrupación de términos) para ayudar al tesauro en la asesoría del usuario, descubriendo otras relaciones que no estén incluidas en el tesauro y proponiéndolas a los usuarios.
- Incorporar la funcionalidad de ampliar el tesauro de forma automática a través de la Web. Esta funcionalidad permitiría ampliar el tesauro de la misma forma que se añaden nuevas urls en la indización. Esta ampliación podría hacerse utilizando técnicas automáticas y manuales, de forma que los propios usuarios puedan, mediante un formulario Web, proponer estas relaciones. El problema de esta aproximación es como defenderse de las proposiciones erróneas o malintencionadas. Debería haber un sistema que compruebe que estas relaciones son correctas. Una posible solución sería utilizando técnicas estadísticas, si varios usuarios introducen un término en el tesauro, se introduce este. Si posteriormente, muchos usuarios creen que esta relación es errónea, se eliminaría.
- Incorporar la posibilidad de cargar los índices en memoria RAM de forma opcional para mejorar los tiempos de recuperación, de forma que para entornos en los que no se requiera un índice muy grande, este se incluya en la memoria RAM, para mejorar los tiempos de recuperación.
- Ampliar los contenidos indizables para que, a demás de páginas Web, pueda realizar búsquedas en pdf, docs, etc., ya que estos tipos de documentos están cada vez más presentes en la Web. Esta ampliación podría favorecer la utilización de este buscador como buscador de escritorio.
- Ampliación de elementos de búsqueda como metadatos HTML, enlaces, etc., para mejorar las posibilidades de búsqueda. Esta ampliación mejoraría su posible utilización en entorno más abiertos distintos a las wikis, como pueden ser sitios Web.
- Implementar un importador de tesauros que soporte los formatos más comunes de creación de tesauros.
- Ajuste de los pesos con algún algoritmo de aprendizaje automático, utilizando por ejemplo Weka.
- Implementar sistemas de búsqueda para elementos no indizables como imágenes, música, videos, utilizando metadatos, etiquetas u otras técnicas.

10 Bibliografía

[1] Portland Pattern Repositori. Repositorio del creador de la wiki: http://c2.com/cgi/wiki?PortlandPatternRepository [consultado 8/06/2008]

[2] Damaris Fuentes Lorenzo. CoolWikiNews, una nueva forma de entender el periodismo electrónico.

[3] Blanca Gil Urdiciain. Orígenes y evolución de los Tesauros en España.

[4] Bruce Eckel. Thinking in Java. 2ª Edición.

[5] Luke Welling Laura Thomson. Desarrollo Web con PHP y MySQL. Anaya.

[6] M.L. Liu. Computación distribuida, Fundamentos y aplicaciones. Pearson.

[7] Core J2EE Patterns: Best Practices and Design Strategies (2nd Edition) (Sun Core Series) (Hardcover)

[8] José Luis Oros. Adobe DreamWeaver CS3 Curso práctico: Editorial RA-MA.

[9]Daniel Borrajo Millán, Jesús González Boticario, Pedro Isasi Viñuela. Aprendizaje Automático. Sainz y Torres.

[10]Ellie Quigley, Marko Gargenta. PHP y Mysql para diseñadores y programadores Web. Anaya Multimedia.

[11] Lee Babin. Introducción a AJAX con PHP. Anaya Multimedia.

[12] Jack Beaird. Diseño Web. Anaya multimedia.

[13] Introducción a CSS. Javier Eguíluz Pérez. Libro online:
http://www.librosweb.es/css/index.html [consultado 8/06/2008]

[14] Introducción a Javascript. Javier Eguíluz Pérez. Libro online: http://www.librosweb.es/javascript/index.html [consultado 8/06/2008]

[15] Lexicografía computacional y semántica, M Antonia Martí Antonín, Ana Fernández Montraveta Gloria Vázquez García. Ediciones Universidad de Barcelona.

[16] Wikipedia, enciclopedia libre online: http://es.wikipedia.org [consultado 8/06/2008]

[17] DeveloperWorks Interviews: Tim Berners-Lee. Entrevista creador de HTTP http://www.ibm.com/developerworks/podcast/dwi/cm-int082206.txt [consultado 8/06/2008]

[18] N. Noy, D. McGuiness. Ontology Development 101: A Guide to Creating Your First Ontology. http://www.ksl.stanford.edu/people/dlm/papers/ontology-tutorial-noy-mcguiness-abstract.html [consultado 8/06/2008]

[19] Wiki del departamento de ingeniería del software http://163.117.147.74/ie/doku.php?id=que_es_un_tesauro [consultado 8/06/2008]

[20] XEROX PARC. Patrón diseño: Modelo Vista Controlador (MVC) http://heim.ifi.uio.no/~trygver/themes/mvc/mvc-index.html [consultado 8/06/2008]

[21] Sonia Sánchez Cuadrado. Definición de ontología para la construcción de un sistema de organización del conocimiento.

[22] Ivar Jacobson. Language Support for Changeable Large Real Time Systems. *Proceedings of OOPSLA'86*. pp 377-384, Sept 1986.

[23] Chignell, Gwizdka, and Bonder. Overlap in Web Search Results: A Study of Five Search Engines

[24] Dulce Aguilar-Lopez, Ivan Lopez-Arevalo, Victor Sosa-Sosa. Uso de ontologías para la mejora de resultados de motores de búsqueda web El Profesional de la Informacion Issue: Volume 18, Number 1 / January - February 2009

11 Anexos

11.1 Ayuda de usuario

Barra de búsqueda

Escriba en la barra de búsqueda las palabras que representen el artículo a buscar. Por defecto, todas las palabras que se incluyan (excepto las palabras con poca significación) aparecerán en el documento (AND). Si se quiere una búsqueda con operadores lógicos hay que especificarla de la siguiente forma:

- Para marcar las palabras que pueden o no estar en los documentos de búsqueda se debe escribir el carácter | delante de la palabra (OR).
- Para marcar las palabras que no deben aparecer en los documentos de búsqueda, se debe escribir el carácter - delante de la palabra que no quiera que aparezca (NOT).
- Para buscar exactamente un grupo de palabras (excluyendo palabras sin significado) se ponen entre comillas simples (') o dobles (“) las palabras a buscar.

¿Dónde podemos buscar?

Los lugares donde podemos buscar son: el wiki del departamento de ingeniería del software de la universidad Carlos III de Madrid, el tesauro que utiliza el buscador y en la Wikipedia.

Búsqueda avanzada

La **búsqueda avanzada** permite definir los operadores lógicos más fácilmente ya que no es necesario recordar los caracteres que la denotan.

En búsqueda avanzada podemos:

- Buscar con todas las palabras (AND)
- Buscar con alguna de las palabras (OR)
- Buscar sin las palabras (NOT)
- Exactamente las palabras indicadas (Entre comillas)
- Precisión: indica la cantidad de documentos que se tendrán en cuenta en la búsqueda. A mayor precisión, mejores serán los resultados pero se incrementarán el tiempo de respuesta del buscador.
- Buscar en revisiones: Existen páginas del wiki con contenido similar pero desactualizado. Las revisiones se guardan en el wiki por lo que el

buscador las recupera junto con la última versión. Si no desea ver esas revisiones desactive esta opción de búsqueda.

La página de resultados

La página de resultados muestra las páginas del wiki que tienen más relevancia de acuerdo a los términos introducidos por el usuario.

El buscador muestra los resultados obtenidos, ordenados por su relevancia respecto a los términos de búsqueda. Pulsando sobre el título de cada uno de ellos se abrirá una nueva ventana mostrando el artículo del wiki recuperado. Además, el buscador proporciona una jerarquía de términos relacionados (izquierda de la imagen). Si la búsqueda no le satisface o simplemente desea buscar por alguno de los términos propuestos, haga clic sobre alguno de ellos y el buscador buscará ese término en el wiki.

De cada una de las páginas se muestra su título, un fragmento donde aparecen los términos de la búsqueda, la URL del documento y una medida sobre su relevancia.

Si la búsqueda se ha realizado sobre la Wikipedia, aparecerá únicamente un resultado, el primero obtenido por Google.

La búsqueda en el tesauro

Si ha elegido la opción de **buscar en el tesauro** del buscador, este mostrará la lista completa de términos relacionados con los propuestos por el usuario.

Con estos términos podemos seguir buscando en el tesauro o podemos buscarlos en el wiki. En cada término propuesto hay un enlace para buscar el término en el wiki o para seguir buscando en el tesauro. En la interfaz se indica el tipo de relación que hay entre los términos obtenidos y los buscados. Hay 4 relaciones:

- Sinónimos: (USE)
- Hiperónimo: termino más general (BT)
- Hipónimo: termino menos general (NT)
- Antónimo: (UF)

Corrección de errores

Si la búsqueda no obtiene resultados, las palabras introducidas en la consulta que no han sido encontradas en el wiki, son corregidas por el buscador con palabras de ortografía similar. El sistema ofrece cinco palabras similares a la introducida que no produjo resultados. Pulsando sobre alguna de estas palabras, el término de la búsqueda se sustituirá por el propuesto por el buscador y se podrá lanzar una nueva búsqueda con los términos corregidos.

Este sistema además de ser más cómodo para el usuario, permite realizar búsquedas sin conocer exactamente como se escribe la palabra que se está buscando.

Añadir URL

Si se detecta por parte de un usuario que hay una página que el motor no ha indexado, se puede añadir la URL pulsando en el enlace de dicho nombre que se encuentra en la página principal.

11. 2 Instalación

- Ejecute el instalador de la base de datos del Manager en el computador que albergará el proceso Manager. Se recomienda utilizar la MySQL Administrator incorporado dentro de las GUI-Tools de MySQL.
- Ejecute el instalador de la base de datos del índice. Si dispone de más de un índice ejecútelo en cada máquina donde se encuentre un índice. Si desea tener dos o más índices en una máquina, tendrá que renombrar el nombre de la base de datos y los ficheros de configuración de dicho índice.
- Configurar el fichero bat del Manager con el puerto y el fichero de configuración del Manager. Ejemplo 1300 Manager/managerConfiguration.cfg
- Configurar el fichero bat de los índices con el puerto y la máquina donde se lanzó el Manager y el nombre del índice, así como el fichero de configuración del índice. Incorporar como último parámetro la modalidad de índice. Hay 3 modalidades:
    - boot: carga inicial de la página. Aproximadamente ente 2 y 4 horas dependiendo de la potencia de la máquina.
    - índice: para buscar documentos con el índice ya construido.
    - *update*: permite buscar y a la vez actualizar el índice.
    - Ejemplo:
        - localhost 1300 indexAgent01 Indexator\Indexator1.cfg índice.

- Lanzar el manager.
- Lanzar cada uno de los índices en modo boot.
- Esperar a que finalice.
- Lanzar el índices en modo *update* o *índice*
- Lanzar el buscador.
- Instalar la Web en una máquina con servidor Web y PHP 5 instalado. La Web no accede a MySQL luego no es necesario que este esté instalado en la máquina donde se ejecute la interfaz Web.

11. 3 Ayuda de administrador

Los ficheros de configuración se encuentran en las siguientes direcciones:

- Buscador.cfg: fichero de configuración para el proceso buscador.
- Útil/palabrasVacias.txt: fichero donde se guardan las palabras vacías.
- Thesaurus/tesis-sonia.tes o WE_xml.xml: fichero donde se guarda dos tesauros.
- Manager/managerConfiguration.cfg: fichero de configuración del manager
- Indexator/indexator1.cfg: configuración del índice 1. 2 Para el índice 2, etc.

11.4 Tesauro.

A continuación se incorpora el código de un segundo tesauro aportado por Eva Carbonero después de realizar la validación. Dicho tesauro es más completo que el utilizado para la validación y se incluye aquí como anexo. En este tesauro se incluyen cuatro tipos de relaciones**:**

- Relación de hiperonímia: BT
- Relacion de hiponímia: NT
- Relación de sinonímia: SN, UF
- Relación de término relacionado: RT

```
@@@
TERM     a.k.a. Classification Software
NODE     463
STATUS   M
BT       Gestión tesauros
STAGE    P
CO       A
CD       04/29/2009
MD       05/04/2009
RT       Tesauros
@@@
TERM     Actividades
NODE     464
STATUS   M
STAGE    P
CO       A
```

CD 05/06/2009
MD 05/06/2009
NT Text mining
NT Stemming
NT Resumen automático
NT Recuperación de información
NT Procesamiento de voz
NT Integración
NT Indización
NT Indexación
NT ERP
NT EDI
NT Desambiguación
NT Data mining
@@@
TERM Adlib
NODE 465
STATUS M
BT Gestión tesauros
STAGE P
CO A
CD 04/29/2009
MD 05/04/2009
@@@
TERM AGROVOC
NODE 466
STATUS M
BT Ejemplos
STAGE P
CO A
CD 04/29/2009
MD 04/29/2009
SN Tesauro multilingue para el ámbito de la agricultura y alimentación
@@@
TERM Alfabética
NODE 467
STATUS M
BT Presentación
STAGE P
CO A
CD 04/29/2009
MD 04/29/2009
RT Notaciones
@@@
TERM AllDifferent
NODE 468
STATUS M
BT Caracteristicas igualdad-desigualdad
STAGE P
CO A
CD 04/30/2009
MD 04/30/2009
SN Un número de individuos puede ser establecido para ser mutuamente distintos
UF all diferent
UF all different
UF alldiferent
@@@
TERM Alt
NODE 469
STATUS M

BT Contenedores
STAGE P
CO A
CD 04/29/2009
MD 04/29/2009
SN Contenedor de RDF para referirse a un grupo de miembros, de los que son alternativos
@@@
TERM Altova Semantic Works
NODE 470
STATUS M
BT Gestión ontologías
STAGE P
CO A
CD 05/04/2009
MD 05/04/2009
RT Ontologías
@@@
TERM Altova XMLSpy
NODE 471
STATUS M
BT Editores XML
STAGE P
CO A
CD 05/04/2009
MD 05/04/2009
RT DTD
RT XML
RT XML Schema
@@@
TERM Amicus Thesaurus
NODE 472
STATUS M
BT Gestión tesauros
STAGE P
CO A
CD 04/29/2009
MD 05/04/2009
RT Tesauros
@@@
TERM ANSI
NODE 473
STATUS M
BT Organismos
STAGE P
CO A
CD 05/06/2009
MD 05/06/2009
SN American National Standards Institute. Organización privada que supervisa el desarrollo de normas de consenso voluntario en EE.UU.
UF American National Standards Institute
RT ANSI ASCX12
RT Estándares
@@@
TERM ANSI ASCX12
NODE 474
STATUS M
BT Estándares
STAGE P
CO A
CD 05/06/2009

MD 05/06/2009
SN American National Standards Institute Accredited Standards Committee X12. Estándar oficial de los EE.UU. para el desarrollo y mantenimiento de EDI (Electronic Data Interchange)
UF American National Standards Institute Accredited Standards Committee X12
UF ASC X12
RT ANSI
RT EDI
RT UN/EDIFACT
@@@
TERM Apollo
NODE 475
STATUS M
BT Gestión ontologías
STAGE P
CO A
CD 05/04/2009
MD 05/04/2009
RT Ontologías
@@@
TERM Árboles
NODE 476
STATUS M
BT Visualización
STAGE P
CO A
CD 04/30/2009
MD 04/30/2009
@@@
TERM ARIADNE
NODE 477
STATUS M
BT Proyectos
STAGE P
CO A
CD 05/05/2009
MD 05/05/2009
RT LOM
@@@
TERM ARQ
NODE 478
STATUS M
BT Lenguajes de recuperación
STAGE P
CO A
CD 05/06/2009
MD 05/06/2009
SN Lenguaje propio del motor de consultas de JENA
RT Jena
@@@
TERM Asimétrico
NODE 479
STATUS M
BT Cualidades
STAGE P
CO A
CD 05/07/2009
MD 05/07/2009
RT Simétrico
@@@
TERM Asociación de linguística computacional

NODE 480
STATUS M
BT Organismos
STAGE P
CO A
CD 05/07/2009
MD 05/07/2009
UF ACL
RT Informática
RT Linguística
@@@
TERM Association
NODE 481
STATUS M
BT Elementos
STAGE P
CO A
CD 04/30/2009
MD 04/30/2009
UF Asociation
@@@
TERM Association role type
NODE 482
STATUS M
BT Elementos
STAGE P
CO A
CD 04/30/2009
MD 04/30/2009
UF asociationroletype
UF associationrole type
UF associationtoletype
@@@
TERM Association type
NODE 483
STATUS M
BT Elementos
STAGE P
CO A
CD 04/30/2009
MD 04/30/2009
UF asociation type
UF asociationtype
@@@
TERM ATOM
NODE 484
STATUS M
BT Lenguajes de sindicación
STAGE P
CO A
CD 05/05/2009
MD 05/05/2009
RT XML
@@@
TERM ATop
NODE 485
STATUS M
BT Gestión Topic Maps
STAGE P
CO A

CD 04/30/2009
MD 05/04/2009
RT Topic maps
@@@
TERM AudioBibliographicRepresentation
NODE 486
STATUS M
BT Esquemas para audio
STAGE P
CO A
CD 05/05/2009
MD 05/05/2009
UF Audio Bibliographjc Representation
@@@
TERM AudioOwner
NODE 487
STATUS M
BT Esquemas para audio
STAGE P
CO A
CD 05/05/2009
MD 05/05/2009
UF Audio owner
@@@
TERM AudioRepresentation
NODE 488
STATUS M
BT Esquemas para audio
STAGE P
CO A
CD 05/05/2009
MD 05/05/2009
UF Audio Representation
@@@
TERM AudioUnit
NODE 489
STATUS M
BT Esquemas para audio
STAGE P
CO A
CD 05/05/2009
MD 05/05/2009
UF Audio Unit
@@@
TERM Bag
NODE 490
STATUS M
BT Contenedores
STAGE P
CO A
CD 04/29/2009
MD 04/29/2009
SN Contenedor de RDF para referirse a un grupo de miembros, donde no es relevante el orden
@@@
TERM base name
NODE 491
STATUS M
BT Elementos
STAGE P

CO A
CD 04/30/2009
MD 04/30/2009
SN La denominación de topic debe realizarse <baseName>
UF basename
@@@
TERM BBCode
NODE 492
STATUS M
BT Lenguajes de marcado
STAGE P
CO A
CD 05/06/2009
MD 05/14/2009
SN Bulletin Board Code. Lenguaje que se emplea en los foros para cambiar y editar los post.
UF Bulletin Board Code
@@@
TERM Bibliometría
NODE 493
STATUS M
BT Cienciometría
STAGE P
CO A
CD 05/07/2009
MD 05/11/2009
SN Parte de la cienciometría que aplica métodos matemáticos y estadísticos a toda la literatura de carácter científico y a los autores que la producen, con el objetivo de estudiar y analizar la actividad científica.
@@@
TERM Bibliotecas
NODE 494
STATUS M
BT Instituciones
STAGE P
CO A
CD 05/07/2009
MD 05/07/2009
RT Documentación
RT Documentalista
@@@
TERM Borradores
NODE 495
STATUS M
BT Documentos
STAGE P
CO A
CD 05/07/2009
MD 05/07/2009
@@@
TERM Buscadores documentos XML
NODE 496
STATUS M
BT Herramientas
STAGE P
CO A
CD 05/04/2009
MD 05/04/2009
UF Buscadores de documentos XML
RT Web semántica
RT XML

```
@@@
TERM    Caracteristicas igualdad-desigualdad
NODE    498
STATUS  M
BT      OWL lite
STAGE   P
CO      A
CD      04/30/2009
MD      04/30/2009
NT      sameAs
NT      equivalentProperty
NT      equivalentClass
NT      differentFrom
NT      AllDifferent
UF      Caracteristicas de igualdad-desigualdad
UF      Características de igualdad y desigualdad
@@@
TERM    Características OWL
NODE    497
STATUS  M
BT      OWL DL
BT      OWL full
STAGE   P
CO      A
CD      04/30/2009
MD      05/14/2009
NT      oneOf
NT      hasValue
NT      disjointWith
NT      Combinaciones booleanas
NT      Cardinalidad
@@@
TERM    Cardinalidad
NODE    499
STATUS  M
BT      Características OWL
STAGE   P
CO      A
CD      04/30/2009
MD      04/30/2009
NT      minCardinality
NT      maxCardinality
NT      complex classes
NT      cardinality
@@@
TERM    cardinality
NODE    500
STATUS  M
BT      Cardinalidad
STAGE   P
CO      A
CD      04/30/2009
MD      04/30/2009
@@@
TERM    CEL
NODE    501
STATUS  M
BT      Razonadores
STAGE   P
CO      A
```

CD 05/04/2009
MD 05/04/2009
RT Ontologías
@@@
TERM Cerebra Engine
NODE 502
STATUS M
BT Razonadores
STAGE P
CO A
CD 05/04/2009
MD 05/04/2009
RT Ontologías
@@@
TERM Cerebra Products
NODE 503
STATUS M
BT Gestión ontologías
STAGE P
CO A
CD 05/04/2009
MD 05/04/2009
RT Ontologías
@@@
TERM Chimaera
NODE 504
STATUS M
BT Ontologías
STAGE P
CO A
CD 05/04/2009
MD 05/04/2009
@@@
TERM Cienciometría
NODE 505
STATUS M
BT Disciplinas
STAGE P
CO A
CD 05/07/2009
MD 05/11/2009
SN Ciencia de medir y analizar la ciencia.
NT Bibliometría
@@@
TERM Clases
NODE 506
STATUS M
BT Componentes
STAGE P
CO A
CD 05/04/2009
MD 05/04/2009
@@@
TERM Class
NODE 507
STATUS M
BT Recursos
STAGE P
CO A
CD 04/29/2009

MD 04/29/2009
NT Individual
RT OWL lite
@@@
TERM CLI
NODE 508
STATUS M
BT Esquemas para imágenes
STAGE P
CO A
CD 05/05/2009
MD 05/05/2009
SN Caller Line Identifier
UF Caller Line Identifier
@@@
TERM Cognatrix
NODE 509
STATUS M
BT Gestión tesauros
STAGE P
CO A
CD 04/29/2009
MD 05/04/2009
RT Tesauros
@@@
TERM Colegios
NODE 510
STATUS M
BT Instituciones
STAGE P
CO A
CD 05/07/2009
MD 05/07/2009
@@@
TERM Combinaciones booleanas
NODE 511
STATUS M
BT Características OWL
STAGE P
CO A
CD 04/30/2009
MD 04/30/2009
NT unionOf
NT intersectionOf
NT complementOf
@@@
TERM complementOf
NODE 512
STATUS M
BT Combinaciones booleanas
STAGE P
CO A
CD 04/30/2009
MD 04/30/2009
UF complement of
UF complement off
UF complementoff
@@@
TERM complex classes
NODE 513

STATUS M
BT Cardinalidad
STAGE P
CO A
CD 04/30/2009
MD 04/30/2009
UF complet classes
UF completclasses
UF complexclasses
@@@
TERM Componentes
NODE 514
STATUS M
BT Ontologías
STAGE P
CO A
CD 05/04/2009
MD 05/04/2009
NT Reglas
NT Propiedades
NT Instancias
NT Clases
@@@
TERM Contenedores
NODE 515
STATUS M
BT RDF
STAGE P
CO A
CD 04/29/2009
MD 04/29/2009
SN Tipos predefinidos para describir elementos
NT Seq
NT Bag
NT Alt
@@@
TERM CSS
NODE 516
STATUS M
BT XML
BT HTML
STAGE P
CO A
CD 05/04/2009
MD 05/04/2009
SN Cascading Style Sheets (hoja de estilos)
UF Cascading Style Sheets
RT CSS Spy
RT CSSED
RT EasyCSS
RT Rapid CSS Editor
RT Style Master
@@@
TERM CSS Spy
NODE 517
STATUS M
BT Editores CSS
STAGE P
CO A
CD 05/06/2009

MD 05/06/2009
RT CSS
@@@
TERM CSSED
NODE 518
STATUS M
BT Editores CSS
STAGE P
CO A
CD 05/06/2009
MD 05/06/2009
SN Editor de CSS para Linux
RT CSS
@@@
TERM Cualidades
NODE 519
STATUS M
STAGE P
CO A
CD 05/07/2009
MD 05/07/2009
NT Técnico
NT Simétrico
NT Hopónimo
NT Hiperónimo
NT Genérico
NT Específico
NT Electrónico
NT Cuantitativo
NT Cualitativo
NT Asimétrico
@@@
TERM Cualitativo
NODE 520
STATUS M
BT Cualidades
STAGE P
CO A
CD 05/07/2009
MD 05/07/2009
RT Cuantitativo
@@@
TERM Cuantitativo
NODE 521
STATUS M
BT Cualidades
STAGE P
CO A
CD 05/07/2009
MD 05/07/2009
RT Cualitativo
@@@
TERM DAML+OIL
NODE 522
STATUS M
BT Lenguajes de marcado
STAGE P
CO A
CD 04/30/2009
MD 05/14/2009

SN Lenguaje de marcado para los recursos de la web semántica. Procede de la unificación de los lenguajes DAML y OIL.
RT OWL
@@@
TERM Data Harmony
NODE 523
STATUS M
BT Gestión tesauros
STAGE P
CO A
CD 04/29/2009
MD 05/04/2009
RT Tesauros
@@@
TERM Data mining
NODE 859
STATUS M
BT Actividades
STAGE P
CO A
CD 05/11/2009
MD 05/11/2009
SN Extracción no trivial de información que reside de manera implícita en los datos.
UF Minería de datos
RT Web semántica
@@@
TERM Data warehouse
NODE 524
STATUS M
BT Datos
STAGE P
CO A
CD 05/06/2009
MD 05/14/2009
SN Colección de datos. Repositorio completo de datos de empresa, donde se almacenan datos estratégicos, tácitos y operativos, al objeto de obtener información estratégico y tácita.
UF Colección de datos
RT Recuperación de información
RT Web semántica
@@@
TERM Datos
NODE 525
STATUS M
STAGE P
CO A
CD 05/06/2009
MD 05/06/2009
NT Data warehouse
@@@
TERM Desambiguación
NODE 526
STATUS M
BT Actividades
STAGE P
CO A
CD 05/06/2009
MD 05/06/2009
@@@
TERM Diccionarios
NODE 528

```
STATUS    M
BT        Vocabularios controlados
STAGE     P
CO        A
CD        05/06/2009
MD        05/06/2009
@@@
TERM      differentFrom
NODE      529
STATUS    M
BT        Caracteristicas igualdad-desigualdad
STAGE     P
CO        A
CD        04/30/2009
MD        04/30/2009
SN        Un individuo puede ser establecido como diferente de otros individuos
UF        diferent from
UF        diferentfrom
UF        Different From
@@@
TERM      Disciplinas
NODE      530
STATUS    M
STAGE     P
CO        A
CD        05/06/2009
MD        05/06/2009
NT        Lógica
NT        Linguística
NT        Informática
NT        Estadística
NT        Documentación
NT        Cienciometría
RT        Profesiones
@@@
TERM      disjointWith
NODE      531
STATUS    M
BT        Características OWL
STAGE     P
CO        A
CD        04/30/2009
MD        04/30/2009
UF        disjoin with
UF        disjoint with
UF        disjoinwith
@@@
TERM      Documentación
NODE      532
STATUS    M
BT        Disciplinas
STAGE     P
CO        A
CD        05/07/2009
MD        05/11/2009
SN        Ciencia del procesamiento de la información.
RT        Bibliotecas
RT        Documentalista
RT        Integración
RT        Linguística computacional
```

RT Páginas web
RT Recuperación de información
RT Resumen automático
RT Sistemas de gestión de bases de datos
@@@
TERM Documentalista
NODE 533
STATUS M
BT Profesiones
STAGE P
CO A
CD 05/07/2009
MD 05/14/2009
SN Profesional experto en la gestión de información, independientemente del formato o ubicación en la que ésta se encuentre.
RT Bibliotecas
RT Documentación
@@@
TERM Documentos
NODE 534
STATUS M
STAGE P
CO A
CD 05/07/2009
MD 05/07/2009
NT Tesis
NT Tesinas
NT Páginas web
NT Manuales
NT Libros de instrucciones
NT Informes
NT Estudios
NT Borradores
@@@
TERM DOM
NODE 535
STATUS M
BT XML
STAGE P
CO A
CD 05/04/2009
MD 05/05/2009
SN Document Object Model (provee un conjunto de objetos para representar un documento XML)
UF Document Object Model
RT HTML
RT W3C
@@@
TERM Domain
NODE 536
STATUS M
BT Property
STAGE P
CO A
CD 04/29/2009
MD 04/30/2009
RT OWL lite
@@@
TERM DOME
NODE 537

STATUS M
BT Gestión ontologías
STAGE P
CO A
CD 05/04/2009
MD 05/04/2009
SN Distributed Ontology Management Environment
RT Ontologías
@@@
TERM DTD
NODE 538
STATUS M
BT XML
STAGE P
CO A
CD 05/04/2009
MD 05/04/2009
SN Document Type Definition (Definición de Tipo de Documento)
UF Document Type Definition
RT Altova XMLSpy
RT Stylus Studio
RT XEmacs
@@@
TERM Dublin Core
NODE 539
STATUS M
BT Vocabularios de Metadatos
STAGE P
CO A
CD 05/05/2009
MD 05/05/2009
UF DC
RT OCLC
@@@
TERM EasyCSS
NODE 540
STATUS M
BT Editores CSS
STAGE P
CO A
CD 05/06/2009
MD 05/06/2009
RT CSS
@@@
TERM EDI
NODE 541
STATUS M
BT Actividades
STAGE P
CO A
CD 05/06/2009
MD 05/06/2009
SN Electronic Data Interchange (Intercambio Electrónico de Datos). El término EDI también se usa para referirse a la implementación y operación de las sistemas y procesos para crear, transmitir y recibir documentos EDI.
NT VAN
UF Electronic Data Interchange
UF Intercambio Electrónico de Datos
RT ANSI ASCX12
RT ERP

RT Naciones Unidas
RT UN/EDIFACT
@@@
TERM Editores CSS
NODE 542
STATUS M
BT Herramientas
STAGE P
CO A
CD 05/06/2009
MD 05/06/2009
NT Style Master
NT Rapid CSS Editor
NT EasyCSS
NT CSSED
NT CSS Spy
@@@
TERM Editores XML
NODE 543
STATUS M
BT Herramientas
STAGE P
CO A
CD 05/04/2009
MD 05/06/2009
NT XEmacs
NT Stylus Studio
NT SLIDE
NT Oxygen XML Editor
NT Microsoft Visual Studio
NT Altova XMLSpy
@@@
TERM EDUCOM
NODE 544
STATUS M
BT Organismos
STAGE P
CO A
CD 05/05/2009
MD 05/06/2009
SN Organización de Educación y Comunicación
RT Estándares
RT LOM
@@@
TERM Ejemplos
NODE 545
STATUS M
BT Tesauros
STAGE P
CO A
CD 04/29/2009
MD 04/29/2009
NT GEMET
NT AGROVOC
@@@
TERM Electrónico
NODE 546
STATUS M
BT Cualidades
STAGE P

```
CO      A
CD      05/07/2009
MD      05/07/2009
@@@
TERM    Elementos
NODE    547
STATUS  M
BT      Topic maps
STAGE   P
CO      A
CD      04/30/2009
MD      04/30/2009
NT      Topic
NT      Scope
NT      Occurrence
NT      base name
NT      Association type
NT      Association role type
NT      Association
@@@
TERM    Encabezamientos de materias
NODE    548
STATUS  M
BT      Vocabularios controlados
STAGE   P
CO      A
CD      05/05/2009
MD      05/05/2009
@@@
TERM    equivalentClass
NODE    549
STATUS  M
BT      Caracteristicas igualdad-desigualdad
STAGE   P
CO      A
CD      04/30/2009
MD      04/30/2009
SN      Dos clases pueden ser establecidas para ser equivalentes
UF      Equivalent
UF      Equivalent class
@@@
TERM    equivalentProperty
NODE    550
STATUS  M
BT      Caracteristicas igualdad-desigualdad
STAGE   P
CO      A
CD      04/30/2009
MD      04/30/2009
SN      Dos propiedades pueden ser establecidas como equivalentes
UF      Equivalent Property
@@@
TERM    ERP
NODE    551
STATUS  M
BT      Actividades
STAGE   P
CO      A
CD      05/06/2009
MD      05/06/2009
```

```
SN      Enterprise Resource Planning (Planificación de Recursos Empresariales)
UF      Enterprise Resource Planning
UF      Planificación de Recursos Empresariales
RT      EDI
@@@
TERM    error
NODE    552
STATUS  M
BT      Esquemas para tesauros
STAGE   P
CO      A
CD      05/04/2009
MD      05/04/2009
SN      Describe un error del proceso por un código y/o una descripción
RT      Tesauros
@@@
TERM    Específico
NODE    553
STATUS  M
BT      Cualidades
STAGE   P
CO      A
CD      05/07/2009
MD      05/07/2009
RT      Genérico
@@@
TERM    Esquemas para audio
NODE    554
STATUS  M
BT      XML Schema
STAGE   P
CO      A
CD      05/05/2009
MD      05/05/2009
NT      TechnicalDescription
NT      AudioUnit
NT      AudioRepresentation
NT      AudioOwner
NT      AudioBibliographicRepresentation
UF      AudioDoc
@@@
TERM    Esquemas para imágenes
NODE    555
STATUS  M
BT      XML Schema
STAGE   P
CO      A
CD      05/05/2009
MD      05/05/2009
NT      XWI
NT      WBMP
NT      OTB
NT      Logos de operador
NT      CLI
@@@
TERM    Esquemas para tesauros
NODE    556
STATUS  M
BT      XML Schema
STAGE   P
```

CO A
CD 05/04/2009
MD 05/04/2009
NT term-description
NT term
NT response
NT properties
NT list
NT hierarchy
NT extended
NT error
@@@
TERM Estadística
NODE 557
STATUS M
BT Disciplinas
STAGE P
CO A
CD 05/07/2009
MD 05/07/2009
@@@
TERM Estadístico
NODE 857
STATUS M
BT Profesiones
STAGE P
CO A
CD 05/11/2009
MD 05/11/2009
@@@
TERM Estándares
NODE 558
STATUS M
STAGE P
CO A
CD 04/28/2009
MD 04/29/2009
NT URI
NT UN/EDIFACT
NT NC-ISO 2788: 2000
NT ISO/IEC FCD 18048
NT ISO/IEC 19503:2005
NT ISO/IEC 13250:2000
NT ID3
NT Exif
NT ANSI ASCX12
UF Estándar
RT ANSI
RT EDUCOM
RT IETF
RT ISO
RT JEITA
RT Naciones Unidas
RT NISO
RT OCLC
RT OMG
RT OWL
RT W3C
@@@
TERM Estudios

NODE 559
STATUS M
BT Documentos
STAGE P
CO A
CD 05/07/2009
MD 05/07/2009
@@@
TERM Exif
NODE 560
STATUS M
BT Estándares
STAGE P
CO A
CD 05/05/2009
MD 05/06/2009
SN Exchangeable Image File Format. Especificación para formatos de archivos de imagen usado por las cámaras digitales.
UF Exchangeable Image File Format
RT JEITA
@@@
TERM extended
NODE 561
STATUS M
BT Esquemas para tesauros
STAGE P
CO A
CD 05/04/2009
MD 05/04/2009
SN Formato opcional de un tesauro específico que describe un solo término
RT Tesauros
@@@
TERM FaCT++
NODE 562
STATUS M
BT Razonadores
STAGE P
CO A
CD 05/04/2009
MD 05/04/2009
RT Ontologías
@@@
TERM FCA-Merge
NODE 563
STATUS M
BT Ontologías
STAGE P
CO A
CD 05/04/2009
MD 05/04/2009
@@@
TERM File
NODE 564
STATUS M
BT URL
STAGE P
CO A
CD 05/05/2009
MD 05/05/2009
SN Apuntan hacia archivos contenidos en el mismo disco que se encuentra el navegador

```
@@@
TERM    FLEX
NODE    565
STATUS  M
BT      Gestión ontologías
STAGE   P
CO      A
CD      05/04/2009
MD      05/04/2009
RT      Ontologías
@@@
TERM    FOAF
NODE    566
STATUS  M
BT      Lenguajes de sindicación
STAGE   P
CO      A
CD      05/05/2009
MD      05/05/2009
SN      Friend of a Friend
UF      Friend of a Friend
RT      RDF
@@@
TERM    F-OWL
NODE    567
STATUS  M
BT      Gestión ontologías
STAGE   P
CO      A
CD      05/04/2009
MD      05/04/2009
RT      Ontologías
@@@
TERM    FTP
NODE    568
STATUS  M
BT      Protocolos
STAGE   P
CO      A
CD      05/05/2009
MD      05/05/2009
SN      File Transfer Protocol
UF      File Transfer Protocol
@@@
TERM    FunctionalProperty
NODE    569
STATUS  M
BT      Identificadores especiales
STAGE   P
CO      A
CD      04/30/2009
MD      04/30/2009
UF      functional property
UF      funtional property
UF      funtionalproperty
@@@
TERM    Fusión
NODE    570
STATUS  M
BT      Ontologías
```

STAGE P
CO A
CD 05/04/2009
MD 05/04/2009
@@@
TERM GEMET
NODE 571
STATUS M
BT Ejemplos
STAGE P
CO A
CD 04/29/2009
MD 04/29/2009
SN General European Multilingual Environmental Thesaurus. Tesauro multilingue de medio ambiente
@@@
TERM Genérico
NODE 572
STATUS M
BT Cualidades
STAGE P
CO A
CD 05/07/2009
MD 05/07/2009
RT Específico
@@@
TERM Gestión ontologías
NODE 573
STATUS M
BT Herramientas
STAGE P
CO A
CD 05/04/2009
MD 05/04/2009
NT WebOnto
NT WebODE
NT TopBraid Composer
NT SymOntoX
NT Protégé
NT OpenKnoME
NT Ontosaurus
NT Ontolingua Server
NT OntoEdit Free and Professional versions
NT ONTO TERM
NT OILEd
NT Jena
NT InferEd
NT IBM Ontology Management System
NT FLEX
NT F-OWL
NT DOME
NT Cerebra Products
NT Apollo
NT Altova Semantic Works
RT Ontologías
RT Web semántica
@@@
TERM Gestión tesauros
NODE 574
STATUS M

BT Herramientas
STAGE P
CO A
CD 04/29/2009
MD 05/04/2009
NT Wordmap
NT WebChoir
NT tmCake
NT Tim Craven-Freeware
NT ThManager
NT Thesmain
NT The Taxonomy Editor
NT Tesaurus Builder
NT TermChoir
NT Term Tree 2000
NT TemaTres
NT TCS-9
NT Synaptica
NT SWOOP
NT STRIDE
NT Star/Thesaurus
NT SIS-TMS
NT SchemaLogic
NT MultiTes
NT Midos Thesaurus
NT LiveLink
NT LinkFactory
NT Data Harmony
NT Cognatrix
NT Amicus Thesaurus
NT Adlib
NT a.k.a. Classification Software
UF Edición y mantenimiento de tesauros
UF Gestión de tesauros
RT Tesauros
@@@
TERM Gestión Topic Maps
NODE 575
STATUS M
BT Herramientas
STAGE P
CO A
CD 04/30/2009
MD 05/04/2009
NT Wandora
NT Topincs
NT Topic Map Designer
NT TMTab
NT TML4Editor
NT The Ceryle Project
NT Simple Topic Maps Management
NT Mapalizer
NT GNOWSYS
NT ATop
UF Edición y gestión de Topic Maps
UF Gestión de Topic Maps
RT Topic maps
@@@
TERM Glosarios
NODE 576

STATUS M
BT Vocabularios controlados
STAGE P
CO A
CD 05/05/2009
MD 05/05/2009
@@@
TERM GNOWSYS
NODE 577
STATUS M
BT Gestión Topic Maps
STAGE P
CO A
CD 04/30/2009
MD 05/04/2009
RT Topic maps
@@@
TERM Gopher
NODE 578
STATUS M
BT Protocolos
STAGE P
CO A
CD 05/05/2009
MD 05/05/2009
@@@
TERM gpath
NODE 579
STATUS M
BT Representación de grafos
STAGE P
CO A
CD 05/05/2009
MD 05/05/2009
@@@
TERM Gráfica
NODE 580
STATUS M
BT Presentación
STAGE P
CO A
CD 04/29/2009
MD 04/29/2009
@@@
TERM Grafos
NODE 581
STATUS M
BT Visualización
STAGE P
CO A
CD 04/30/2009
MD 04/30/2009
@@@
TERM hasValue
NODE 582
STATUS M
BT Características OWL
STAGE P
CO A
CD 04/30/2009

MD 04/30/2009
SN Valores de propiedad
UF Has value
@@@
TERM Herramientas
NODE 583
STATUS M
STAGE P
CO A
CD 04/28/2009
MD 05/04/2009
NT Visualización Topic Maps
NT Sistemas de gestión de bases de datos
NT Representación de grafos
NT Razonadores
NT Gestión Topic Maps
NT Gestión tesauros
NT Gestión ontologías
NT Editores XML
NT Editores CSS
NT Buscadores documentos XML
UF Herramienta
UF Software
@@@
TERM hierarchy
NODE 584
STATUS M
BT Esquemas para tesauros
STAGE P
CO A
CD 05/04/2009
MD 05/04/2009
SN Describe la jerarquía de términos de un término preferido
RT Tesauros
@@@
TERM Hiperónimo
NODE 585
STATUS M
BT Cualidades
STAGE P
CO A
CD 05/07/2009
MD 05/07/2009
RT Hopónimo
@@@
TERM Hopónimo
NODE 586
STATUS M
BT Cualidades
STAGE P
CO A
CD 05/07/2009
MD 05/07/2009
RT Hiperónimo
@@@
TERM HP Labs Semantic Web Programme
NODE 587
STATUS M
BT Proyectos
STAGE P

CO A
CD 05/06/2009
MD 05/06/2009
SN Proyecto de investigación sobre la web semántica llevado a cabo por HP (Hewlett-Packard)
RT Web semántica
@@@
TERM HTML
NODE 588
STATUS M
BT Lenguajes de marcado
STAGE P
CO A
CD 04/29/2009
MD 04/29/2009
SN HyperText Markup Language
NT CSS
RT DOM
RT XHTML
RT XML
@@@
TERM HTTP
NODE 589
STATUS M
BT Protocolos
STAGE P
CO A
CD 05/05/2009
MD 05/05/2009
SN HyperText Transfer Protocol
UF HyperText Transfer Protocol
@@@
TERM HyperGraph
NODE 590
STATUS M
BT Visualización Topic Maps
STAGE P
CO A
CD 04/30/2009
MD 05/04/2009
@@@
TERM HyTime
NODE 591
STATUS M
BT Lenguajes de marcado
STAGE P
CO A
CD 04/30/2009
MD 04/30/2009
UF Hy time
RT SGML
@@@
TERM IBM Ontology Management System
NODE 592
STATUS M
BT Gestión ontologías
STAGE P
CO A
CD 05/04/2009
MD 05/04/2009

RT Ontologías
@@@
TERM ID3
NODE 593
STATUS M
BT Estándares
STAGE P
CO A
CD 05/05/2009
MD 05/05/2009
SN Estándar de facto para incluir metadatos en un contenedor multimedia
@@@
TERM Identificadores especiales
NODE 594
STATUS M
BT OWL lite
STAGE P
CO A
CD 04/30/2009
MD 04/30/2009
NT TransitiveProperty
NT SymmetricProperty
NT inverseOf
NT InverseFunctionalProperty
NT FunctionalProperty
@@@
TERM IEC
NODE 863
STATUS M
BT Organismos
STAGE P
CO A
CD 05/14/2009
MD 05/14/2009
SN International Electrotechnical Commission
UF International Electrotechnical Commission
RT ISO
RT ISO/IEC 13250:2000
RT ISO/IEC 19503:2005
RT ISO/IEC FCD 18048
@@@
TERM IETF
NODE 595
STATUS M
BT Organismos
STAGE P
CO A
CD 05/05/2009
MD 05/06/2009
SN Internet Engineering Task Force
UF Internet Engineering Task Force
RT Estándares
@@@
TERM Impresa
NODE 860
STATUS M
BT Presentación
STAGE P
CO A
CD 05/11/2009

MD 05/11/2009
@@@
TERM IMS
NODE 596
STATUS M
BT Proyectos
STAGE P
CO A
CD 05/05/2009
MD 05/05/2009
SN Instructional Management System
RT LOM
@@@
TERM Indexación
NODE 861
STATUS M
BT Actividades
STAGE P
CO A
CD 05/11/2009
MD 05/11/2009
SN Acción y efecto de indexar (Registrar ordenadamente datos e informaciones, para elaborar su índice)
RT Recuperación de información
@@@
TERM Índices
NODE 597
STATUS M
BT Presentación
STAGE P
CO A
CD 04/29/2009
MD 04/29/2009
@@@
TERM Individual
NODE 598
STATUS M
BT Class
STAGE P
CO A
CD 04/30/2009
MD 04/30/2009
RT OWL lite
@@@
TERM Indización
NODE 599
STATUS M
BT Actividades
STAGE P
CO A
CD 05/07/2009
MD 05/11/2009
SN Proceso de describir o representar el contenido temático de un recurso de información.
RT Recuperación de información
@@@
TERM InferEd
NODE 600
STATUS M
BT Gestión ontologías
STAGE P

CO A
CD 05/04/2009
MD 05/04/2009
RT Ontologías
@@@
TERM Informática
NODE 601
STATUS M
BT Disciplinas
STAGE P
CO A
CD 05/07/2009
MD 05/11/2009
SN Conjunto de conocimientos científicos y técnicas que hacen posible el tratamiento automático de la información por medio de ordenadores.
NT Inteligencia artificial
RT Asociación de linguística computacional
RT Informático
RT Integración
RT Lenguajes de programación
RT Lógica
RT Procesamiento de voz
RT Procesamiento del lenguaje natural
RT Páginas web
RT Recuperación de información
RT Resumen automático
RT Sistemas de gestión de bases de datos
RT Sistemas operativos
RT Traducción automática
@@@
TERM Informático
NODE 602
STATUS M
BT Profesiones
STAGE P
CO A
CD 05/07/2009
MD 05/07/2009
RT Informática
RT Lenguajes de programación
@@@
TERM Informes
NODE 603
STATUS M
BT Documentos
STAGE P
CO A
CD 05/07/2009
MD 05/07/2009
@@@
TERM Ingeniero
NODE 858
STATUS M
BT Profesiones
STAGE P
CO A
CD 05/11/2009
MD 05/11/2009
@@@
TERM Instancias

NODE 604
STATUS M
BT Componentes
STAGE P
CO A
CD 05/04/2009
MD 05/04/2009
@@@
TERM Instituciones
NODE 605
STATUS M
STAGE P
CO A
CD 05/07/2009
MD 05/07/2009
NT Universidades
NT Colegios
NT Bibliotecas
@@@
TERM Integración
NODE 606
STATUS M
BT Actividades
STAGE P
CO A
CD 05/07/2009
MD 05/07/2009
RT Documentación
RT Informática
RT Sistemas de gestión de bases de datos
@@@
TERM Inteligencia artificial
NODE 607
STATUS M
BT Informática
STAGE P
CO A
CD 05/07/2009
MD 05/11/2009
SN Rama de la ciencia informática dedicada al desarrollo de agentes racionales no vivos.
NT Procesamiento del lenguaje natural
UF AI
RT Linguística computacional
@@@
TERM intersectionOf
NODE 608
STATUS M
BT Combinaciones booleanas
STAGE P
CO A
CD 04/30/2009
MD 04/30/2009
UF inserction of
UF inserction off
UF inserctionoff
@@@
TERM Intranet
NODE 609
STATUS M
BT Redes

STAGE P
CO A
CD 05/07/2009
MD 05/07/2009
SN Red interna
@@@
TERM InverseFunctionalProperty
NODE 610
STATUS M
BT Identificadores especiales
STAGE P
CO A
CD 04/30/2009
MD 04/30/2009
UF inverse functional property
UF inverse funtional property
UF inversefunctionalpropperty
@@@
TERM inverseOf
NODE 611
STATUS M
BT Identificadores especiales
STAGE P
CO A
CD 04/30/2009
MD 04/30/2009
SN Una propiedad puede ser establecida para ser la inversa de otra propiedad
UF inserseoff
UF inverse off
@@@
TERM ISO
NODE 612
STATUS M
BT Organismos
STAGE P
CO A
CD 04/29/2009
MD 05/14/2009
SN International Organization for Standarization. ISO y el IEC colaboran conjuntamente en campos de interés mutuo.
UF International Organization for Standarization
RT Estándares
RT IEC
RT ISO/IEC 13250:2000
RT ISO/IEC 19503:2005
RT ISO/IEC FCD 18048
@@@
TERM ISO/IEC 13250:2000
NODE 613
STATUS M
BT Estándares
STAGE P
CO A
CD 04/30/2009
MD 05/06/2009
SN Estándar para los Topic maps
UF Estándar para Topic maps
UF Estándar Topic maps
UF Norma topic maps
RT IEC

```
RT      ISO
RT      Topic maps
@@@
TERM    ISO/IEC 19503:2005
NODE    614
STATUS  M
BT      Estándares
STAGE   P
CO      A
CD      05/05/2009
MD      05/06/2009
SN      Estándar para XMI (XML Metadata Interchange)
RT      IEC
RT      ISO
@@@
TERM    ISO/IEC FCD 18048
NODE    615
STATUS  M
BT      Estándares
STAGE   P
CO      A
CD      05/04/2009
MD      05/06/2009
SN      Estándar para TMQL (Lenguaje de recuperación de Topic Maps)
RT      IEC
RT      ISO
RT      TMQL
RT      Topic maps
@@@
TERM    JAVA
NODE    616
STATUS  M
BT      Lenguajes de programación
STAGE   P
CO      A
CD      05/05/2009
MD      05/05/2009
NT      Sesame
RT      JDOM
RT      Jena
@@@
TERM    JavaScript
NODE    617
STATUS  M
BT      Lenguajes de programación
STAGE   P
CO      A
CD      05/05/2009
MD      05/05/2009
@@@
TERM    JDOM
NODE    618
STATUS  M
BT      XML
STAGE   P
CO      A
CD      05/05/2009
MD      05/05/2009
SN      Aplicación para leer, crear y manipular documentos XML
RT      JAVA
```

RT XML
@@@
TERM JEITA
NODE 619
STATUS M
BT Organismos
STAGE P
CO A
CD 05/05/2009
MD 05/06/2009
SN Asociación Japonesa de Tecnologías Electrónicas y de la Inforamción
RT Estándares
RT Exif
@@@
TERM Jena
NODE 620
STATUS M
BT Gestión ontologías
STAGE P
CO A
CD 05/04/2009
MD 05/06/2009
RT ARQ
RT JAVA
RT Ontologías
RT RDF
RT RDQL
RT Web semántica
@@@
TERM KAON2
NODE 621
STATUS M
BT Razonadores
STAGE P
CO A
CD 05/04/2009
MD 05/04/2009
RT Ontologías
@@@
TERM Lenguajes
NODE 622
STATUS M
STAGE P
CO A
CD 04/29/2009
MD 04/29/2009
NT Lenguajes de sindicación
NT Lenguajes de recuperación
NT Lenguajes de programación
NT Lenguajes de modelado
NT Lenguajes de marcado
NT Lenguajes de descripción
@@@
TERM Lenguajes de descripción
NODE 623
STATUS M
BT Lenguajes
STAGE P
CO A
CD 05/04/2009

MD 05/04/2009
NT WSMF
NT WSDL
NT WS-CDL
NT SWSL
NT SVG
NT RDF
@@@
TERM Lenguajes de marcado
NODE 624
STATUS M
BT Lenguajes
STAGE P
CO A
CD 04/29/2009
MD 04/29/2009
NT YAML
NT XML
NT XHTML
NT WML
NT SOX
NT SLiP
NT SGML
NT RuleML
NT OWL
NT OGDL
NT MathML
NT HyTime
NT HTML
NT DAML+OIL
NT BBCode
RT Web semántica
@@@
TERM Lenguajes de modelado
NODE 625
STATUS M
BT Lenguajes
STAGE P
CO A
CD 05/04/2009
MD 05/04/2009
NT WSMO
NT WSML
NT UML
@@@
TERM Lenguajes de programación
NODE 626
STATUS M
BT Lenguajes
STAGE P
CO A
CD 05/05/2009
MD 05/07/2009
NT VBScript
NT Python
NT PHP
NT Perl
NT JavaScript
NT JAVA
RT Informática

RT Informático
@@@
TERM Lenguajes de recuperación
NODE 627
STATUS M
BT Lenguajes
STAGE P
CO A
CD 05/04/2009
MD 05/04/2009
NT Xquery
NT TOLOG
NT TMQL
NT TMCL
NT SQL
NT SPARQL
NT RDQL
NT ARQ
UF Lenguajes de consulta
@@@
TERM Lenguajes de sindicación
NODE 628
STATUS M
BT Lenguajes
STAGE P
CO A
CD 05/05/2009
MD 05/05/2009
NT RSS
NT FOAF
NT ATOM
@@@
TERM Libros de instrucciones
NODE 629
STATUS M
BT Documentos
STAGE P
CO A
CD 05/07/2009
MD 05/07/2009
@@@
TERM Linguista
NODE 630
STATUS M
BT Profesiones
STAGE P
CO A
CD 05/07/2009
MD 05/07/2009
RT Linguística
@@@
TERM Linguística
NODE 631
STATUS M
BT Disciplinas
STAGE P
CO A
CD 05/07/2009
MD 05/07/2009
NT Terminología

NT Linguística computacional
RT Asociación de linguística computacional
RT Linguista
RT Procesamiento de voz
RT Resumen automático
@@@
TERM Linguística computacional
NODE 632
STATUS M
BT Linguística
STAGE P
CO A
CD 05/07/2009
MD 05/11/2009
SN Aplicación de los métodos de la inteligencia artificial al tratamiento de cuestiones lingüísticas.
NT Traducción automática
RT Documentación
RT Inteligencia artificial
RT Procesamiento del lenguaje natural
RT Stemming
@@@
TERM LinkFactory
NODE 633
STATUS M
BT Gestión tesauros
STAGE P
CO A
CD 04/29/2009
MD 05/04/2009
RT Ontologías
@@@
TERM Linux
NODE 634
STATUS M
BT Sistemas operativos
STAGE P
CO A
CD 05/07/2009
MD 05/07/2009
@@@
TERM list
NODE 635
STATUS M
BT Esquemas para tesauros
STAGE P
CO A
CD 05/04/2009
MD 05/04/2009
SN Una lista de cero o más términos
RT Tesauros
@@@
TERM LiveLink
NODE 636
STATUS M
BT Gestión tesauros
STAGE P
CO A
CD 04/29/2009
MD 05/04/2009

@@@
TERM Lógica
NODE 637
STATUS M
BT Disciplinas
STAGE P
CO A
CD 05/07/2009
MD 05/11/2009
SN Ciencia que opera utilizando un lenguaje simbólico artificial y haciendo abstracción de los contenidos.
RT Informática
RT Recuperación de información
@@@
TERM Logos de operador
NODE 638
STATUS M
BT Esquemas para imágenes
STAGE P
CO A
CD 05/05/2009
MD 05/05/2009
@@@
TERM LOM
NODE 639
STATUS M
BT Vocabularios de Metadatos
STAGE P
CO A
CD 05/05/2009
MD 05/05/2009
SN Learning Object Metadata
UF Learning Object Metadata
RT ARIADNE
RT EDUCOM
RT IMS
RT XML
@@@
TERM Mailto
NODE 640
STATUS M
BT URL
STAGE P
CO A
CD 05/05/2009
MD 05/05/2009
@@@
TERM Manuales
NODE 641
STATUS M
BT Documentos
STAGE P
CO A
CD 05/07/2009
MD 05/07/2009
@@@
TERM Mapalizer
NODE 642
STATUS M
BT Gestión Topic Maps

STAGE P
CO A
CD 04/30/2009
MD 05/04/2009
RT Topic maps
@@@
TERM Mapas
NODE 643
STATUS M
BT Visualización
STAGE P
CO A
CD 04/30/2009
MD 04/30/2009
@@@
TERM MathML
NODE 644
STATUS M
BT Lenguajes de marcado
STAGE P
CO A
CD 05/05/2009
MD 05/05/2009
SN Mathematical Markup Language
UF Mathematical Markup Language
@@@
TERM maxCardinality
NODE 645
STATUS M
BT Cardinalidad
STAGE P
CO A
CD 04/30/2009
MD 04/30/2009
UF max cardinality
@@@
TERM METHONTOLOGY
NODE 646
STATUS M
BT Metodologias de desarrollo
STAGE P
CO A
CD 05/04/2009
MD 05/04/2009
@@@
TERM Metodologias de desarrollo
NODE 527
STATUS M
BT Ontologías
STAGE P
CO A
CD 05/04/2009
MD 05/14/2009
NT On-To-Knowledge
NT METHONTOLOGY
@@@
TERM Microsoft Visual Studio
NODE 647
STATUS M
BT Editores XML

STAGE P
CO A
CD 05/04/2009
MD 05/04/2009
RT XML
RT XML Schema
@@@
TERM Midos Thesaurus
NODE 648
STATUS M
BT Gestión tesauros
STAGE P
CO A
CD 04/29/2009
MD 05/04/2009
RT Tesauros
@@@
TERM minCardinality
NODE 649
STATUS M
BT Cardinalidad
STAGE P
CO A
CD 04/30/2009
MD 04/30/2009
UF Min cardinality
@@@
TERM MSPASS
NODE 650
STATUS M
BT Razonadores
STAGE P
CO A
CD 05/04/2009
MD 05/04/2009
RT Ontologías
@@@
TERM Multidimensional
NODE 651
STATUS M
BT Visualización
STAGE P
CO A
CD 04/30/2009
MD 04/30/2009
@@@
TERM MultiTes
NODE 652
STATUS M
BT Gestión tesauros
STAGE P
CO A
CD 04/29/2009
MD 05/04/2009
RT Tesauros
@@@
TERM NA
NODE 653
STATUS M
BT Notaciones

STAGE P
CO A
CD 04/29/2009
MD 05/04/2009
SN Notas aclarativas
@@@
TERM Naciones Unidas
NODE 654
STATUS M
BT Organismos
STAGE P
CO A
CD 05/06/2009
MD 05/06/2009
UF UN
RT EDI
RT Estándares
RT UN/EDIFACT
@@@
TERM NC-ISO 2788: 2000
NODE 862
STATUS M
BT Estándares
STAGE P
CO A
CD 05/11/2009
MD 05/11/2009
SN Estándar para el establecimiento y desarrollo de tesauros monolingües.
RT Tesauros
RT Web semántica
@@@
TERM News
NODE 655
STATUS M
BT URL
STAGE P
CO A
CD 05/05/2009
MD 05/05/2009
@@@
TERM NISO
NODE 656
STATUS M
BT Organismos
STAGE P
CO A
CD 04/29/2009
MD 05/14/2009
SN National Information Standards Organization. Identifica, desarrolla, controla y publica estándares técnicos para el manejo de la información cambiante.
UF National Information Standards Organization
RT Estándares
@@@
TERM Notaciones
NODE 657
STATUS M
BT Tesauros
STAGE P
CO A
CD 04/29/2009

MD 05/04/2009
NT USE
NT UP
NT TR
NT TG
NT TE
NT NA
RT Alfabética
RT Sistemática
@@@
TERM Occurrence
NODE 658
STATUS M
BT Elementos
STAGE P
CO A
CD 04/30/2009
MD 04/30/2009
UF ocurrence
@@@
TERM OCLC
NODE 659
STATUS M
BT Organismos
STAGE P
CO A
CD 05/05/2009
MD 05/14/2009
SN Online Computer Library Center. Organización sin ánimo de lucro, dedicada a ofrecer servicios bibliotecarios computarizados y de investigación, para facilitar el acceso a la información y reducir los costes. Fundado en 1967, con sede en Dublin, Ohio.
UF Online Computer Library Center
RT Dublin Core
RT Estándares
@@@
TERM OGDL
NODE 660
STATUS M
BT Lenguajes de marcado
STAGE P
CO A
CD 05/05/2009
MD 05/05/2009
SN Ordered Graph Data Language
UF Ordered Graph Data Language
RT Representación de grafos
@@@
TERM OILEd
NODE 661
STATUS M
BT Gestión ontologías
STAGE P
CO A
CD 05/04/2009
MD 05/04/2009
RT Ontologías
@@@
TERM OMG
NODE 662
STATUS M

BT Organismos
STAGE P
CO A
CD 05/05/2009
MD 05/14/2009
SN Object Management Group. Consorcio informático sin ánimo de lucro dedicado al desarrollo de normas para la integración de software en la empresa.
UF Object Management Group
RT Estándares
@@@
TERM oneOf
NODE 663
STATUS M
BT Características OWL
STAGE P
CO A
CD 04/30/2009
MD 04/30/2009
SN Clases enumeradas
UF one of
UF one off
UF oneoff
@@@
TERM ONTO TERM
NODE 664
STATUS M
BT Gestión ontologías
STAGE P
CO A
CD 05/04/2009
MD 05/04/2009
RT Ontologías
@@@
TERM OntoEdit Free and Professional versions
NODE 665
STATUS M
BT Gestión ontologías
STAGE P
CO A
CD 05/04/2009
MD 05/04/2009
RT Ontologías
@@@
TERM On-To-Knowledge
NODE 666
STATUS M
BT Metodologias de desarrollo
STAGE P
CO A
CD 05/04/2009
MD 05/04/2009
@@@
TERM Ontolingua Server
NODE 667
STATUS M
BT Gestión ontologías
STAGE P
CO A
CD 05/04/2009
MD 05/04/2009

```
RT      Ontologías
@@@
TERM    Ontologías
NODE    668
STATUS  M
BT      Vocabularios controlados
STAGE   P
CO      A
CD      04/29/2009
MD      04/29/2009
NT      PROMPT
NT      Metodologias de desarrollo
NT      Fusión
NT      FCA-Merge
NT      Componentes
NT      Chimaera
RT      Altova Semantic Works
RT      Apollo
RT      CEL
RT      Cerebra Engine
RT      Cerebra Products
RT      DOME
RT      F-OWL
RT      FaCT++
RT      FLEX
RT      Gestión ontologías
RT      IBM Ontology Management System
RT      InferEd
RT      Jena
RT      KAON2
RT      LinkFactory
RT      MSPASS
RT      OILEd
RT      ONTO TERM
RT      OntoEdit Free and Professional versions
RT      Ontolingua Server
RT      Ontosaurus
RT      OpenKnoME
RT      OWL
RT      Pellet
RT      Protégé
RT      Protégé 2000
RT      Protégé 3.2.1
RT      RacerPro
RT      Razonadores
RT      Sistemas de organización del conocimiento
RT      SWSL
RT      SymOntoX
RT      Tesauros
RT      TopBraid Composer
RT      Topic maps
RT      Web semántica
RT      WebODE
RT      WebOnto
RT      WS-CDL
RT      WSDL
RT      WSMF
RT      WSML
RT      WSMO
@@@
```

TERM Ontosaurus
NODE 669
STATUS M
BT Gestión ontologías
STAGE P
CO A
CD 05/04/2009
MD 05/04/2009
RT Ontologías
@@@
TERM OpenKnoME
NODE 670
STATUS M
BT Gestión ontologías
STAGE P
CO A
CD 05/04/2009
MD 05/04/2009
RT Ontologías
@@@
TERM Organismos
NODE 671
STATUS M
STAGE P
CO A
CD 04/29/2009
MD 05/06/2009
NT W3C
NT OMG
NT OCLC
NT NISO
NT Naciones Unidas
NT JEITA
NT ISO
NT IETF
NT IEC
NT EDUCOM
NT Asociación de linguística computacional
NT ANSI
@@@
TERM OTB
NODE 672
STATUS M
BT Esquemas para imágenes
STAGE P
CO A
CD 05/05/2009
MD 05/05/2009
SN On-Thr-air Bitmap
UF On-Thr-air Bitmap
@@@
TERM OWL
NODE 673
STATUS M
BT Lenguajes de marcado
STAGE P
CO A
CD 04/29/2009
MD 04/30/2009
SN Ontology Web Language; estándar de la Web semántica

NT Variantes
UF Ontology Web Language
RT DAML+OIL
RT Estándares
RT Ontologías
RT Sistemas de organización del conocimiento
RT W3C
RT Web semántica
@@@
TERM OWL DL
NODE 674
STATUS M
BT Variantes
STAGE P
CO A
CD 04/30/2009
MD 04/30/2009
NT Características OWL
@@@
TERM OWL full
NODE 675
STATUS M
BT Variantes
STAGE P
CO A
CD 04/30/2009
MD 04/30/2009
NT Características OWL
@@@
TERM OWL lite
NODE 676
STATUS M
BT Variantes
STAGE P
CO A
CD 04/30/2009
MD 04/30/2009
NT Identificadores especiales
NT Caracteristicas igualdad-desigualdad
RT Class
RT Domain
RT Individual
RT Range
RT SubClassOf
@@@
TERM Oxygen XML Editor
NODE 677
STATUS M
BT Editores XML
STAGE P
CO A
CD 05/04/2009
MD 05/04/2009
RT XML
@@@
TERM Páginas web
NODE 678
STATUS M
BT Documentos
STAGE P

CO A
CD 05/07/2009
MD 05/11/2009
SN Documento situado en una red informática, al que se accede mediante enlaces de hipertexto.
RT Documentación
RT Informática
@@@
TERM Panckoucke
NODE 679
STATUS M
BT Visualización Topic Maps
STAGE P
CO A
CD 04/30/2009
MD 05/04/2009
@@@
TERM Pellet
NODE 680
STATUS M
BT Razonadores
STAGE P
CO A
CD 05/04/2009
MD 05/04/2009
RT Ontologías
@@@
TERM Perl
NODE 681
STATUS M
BT Lenguajes de programación
STAGE P
CO A
CD 05/06/2009
MD 05/06/2009
@@@
TERM PHP
NODE 682
STATUS M
BT Lenguajes de programación
STAGE P
CO A
CD 05/05/2009
MD 05/05/2009
@@@
TERM Presentación
NODE 683
STATUS M
BT Tesauros
STAGE P
CO A
CD 04/29/2009
MD 04/29/2009
NT Índices
NT Sistemática
NT Impresa
NT Gráfica
NT Alfabética
@@@
TERM Procesamiento de voz

NODE 684
STATUS M
BT Actividades
STAGE P
CO A
CD 05/07/2009
MD 05/07/2009
RT Informática
RT Linguística
@@@
TERM Procesamiento del lenguaje natural
NODE 685
STATUS M
BT Inteligencia artificial
STAGE P
CO A
CD 05/07/2009
MD 05/14/2009
SN Subdisciplina de la Inteligencia Artificial y de la lingüística computacional. El PLN se ocupa de la formulación e investigación de mecanismos eficaces computacionalmente para la comunicación entre personas o entre personas y máquinas por medio de lenguajes naturales.
UF NLP
UF PLN
RT Informática
RT Linguística computacional
RT Web semántica
@@@
TERM Profesiones
NODE 686
STATUS M
STAGE P
CO A
CD 05/07/2009
MD 05/07/2009
NT Terminólogo
NT Linguista
NT Ingeniero
NT Informático
NT Estadístico
NT Documentalista
RT Disciplinas
@@@
TERM PROMPT
NODE 687
STATUS M
BT Ontologías
STAGE P
CO A
CD 05/04/2009
MD 05/04/2009
@@@
TERM properties
NODE 688
STATUS M
BT Esquemas para tesauros
STAGE P
CO A
CD 05/04/2009
MD 05/04/2009
SN Describe las características totales del tesauro

RT Tesauros
@@@
TERM Property
NODE 689
STATUS M
BT RDFS
STAGE P
CO A
CD 04/29/2009
MD 05/04/2009
NT Type
NT SubClassOf
NT Range
NT Domain
@@@
TERM Propiedades
NODE 690
STATUS M
BT Componentes
STAGE P
CO A
CD 05/04/2009
MD 05/04/2009
@@@
TERM Protégé
NODE 691
STATUS M
BT Gestión ontologías
STAGE P
CO A
CD 05/04/2009
MD 05/04/2009
NT Protégé 3.2.1
NT Protégé 2000
RT Ontologías
@@@
TERM Protégé 2000
NODE 692
STATUS M
BT Protégé
STAGE P
CO A
CD 05/04/2009
MD 05/04/2009
UF otégé-2000
UF Protégé2000
RT Ontologías
@@@
TERM Protégé 3.2.1
NODE 693
STATUS M
BT Protégé
STAGE P
CO A
CD 05/04/2009
MD 05/04/2009
UF Protégé versión 3.2.1
RT Ontologías
@@@
TERM Protocolos

NODE 694
STATUS M
STAGE P
CO A
CD 05/05/2009
MD 05/05/2009
NT HTTP
NT Gopher
NT FTP
@@@
TERM Proyectos
NODE 695
STATUS M
STAGE P
CO A
CD 04/29/2009
MD 05/06/2009
NT SWAD-Europe
NT RODA
NT IMS
NT HP Labs Semantic Web Programme
NT ARIADNE
RT SLiP
@@@
TERM PSI
NODE 696
STATUS M
BT Topic maps
STAGE P
CO A
CD 04/30/2009
MD 04/30/2009
SN Published Subject Indicator
UF PSI'S
UF Published Subject Indicator
@@@
TERM Python
NODE 697
STATUS M
BT Lenguajes de programación
STAGE P
CO A
CD 05/06/2009
MD 05/06/2009
RT SLiP
@@@
TERM RacerPro
NODE 698
STATUS M
BT Razonadores
STAGE P
CO A
CD 05/04/2009
MD 05/04/2009
RT Ontologías
@@@
TERM Range
NODE 699
STATUS M
BT Property

STAGE P
CO A
CD 04/29/2009
MD 04/30/2009
RT OWL lite
@@@
TERM Rapid CSS Editor
NODE 700
STATUS M
BT Editores CSS
STAGE P
CO A
CD 05/06/2009
MD 05/06/2009
RT CSS
@@@
TERM Razonadores
NODE 701
STATUS M
BT Herramientas
STAGE P
CO A
CD 05/04/2009
MD 05/04/2009
NT RacerPro
NT Pellet
NT MSPASS
NT KAON2
NT FaCT++
NT Cerebra Engine
NT CEL
RT Ontologías
@@@
TERM RDF
NODE 702
STATUS M
BT Lenguajes de descripción
STAGE P
CO A
CD 04/29/2009
MD 05/14/2009
SN Resource Description Framework; estándar de la Web semántica. Marco para los metadatos elaborado por el W3C.
NT RDFS
NT Contenedores
RT FOAF
RT Jena
RT RSS
RT RuleML
RT Sesame
RT Sistemas de organización del conocimiento
RT SPARQL
RT W3C
RT Web semántica
RT WSMO
RT XML
@@@
TERM RDFS
NODE 703
STATUS M

```
BT      RDF
STAGE   P
CO      A
CD      04/29/2009
MD      04/29/2009
SN      Extensión semántica de RDF
NT      SKOS Mapping
NT      SKOS Extensions
NT      Skos Core
NT      Recursos
NT      Property
UF      Esquema
UF      Schema
RT      Sesame
@@@
TERM    RDQL
NODE    704
STATUS  M
BT      Lenguajes de recuperación
STAGE   P
CO      A
CD      05/06/2009
MD      05/06/2009
SN      RDF Data Query Language
UF      RDF Data Query Language
RT      Jena
@@@
TERM    Recuperación de información
NODE    705
STATUS  M
BT      Actividades
STAGE   P
CO      A
CD      05/07/2009
MD      05/11/2009
UF      Information Retrieval
UF      IR
UF      ROI
RT      Data warehouse
RT      Documentación
RT      Indexación
RT      Indización
RT      Informática
RT      Lógica
RT      Text mining
RT      Web semántica
@@@
TERM    Recursos
NODE    706
STATUS  M
BT      RDFS
STAGE   P
CO      A
CD      04/29/2009
MD      04/29/2009
SN      Las clases definidas por RDFS son descritas por recursos
NT      Resource
NT      Class
@@@
TERM    Redes
```

```
NODE	707
STATUS	M
STAGE	P
CO	A
CD	05/07/2009
MD	05/07/2009
NT	Web semántica
NT	Web 2.0
NT	Web 1.0
NT	Intranet
@@@
TERM	Reglas
NODE	708
STATUS	M
BT	Componentes
STAGE	P
CO	A
CD	05/04/2009
MD	05/04/2009
@@@
TERM	Representación de grafos
NODE	709
STATUS	M
BT	Herramientas
STAGE	P
CO	A
CD	05/05/2009
MD	05/05/2009
NT	xml2ogdl
NT	gpath
RT	OGDL
@@@
TERM	Resource
NODE	710
STATUS	M
BT	Recursos
STAGE	P
CO	A
CD	04/29/2009
MD	04/29/2009
@@@
TERM	response
NODE	711
STATUS	M
BT	Esquemas para tesauros
STAGE	P
CO	A
CD	05/04/2009
MD	05/04/2009
SN	Contiene la respuesta de un servicio del tesauro
RT	Tesauros
@@@
TERM	Resumen automático
NODE	712
STATUS	M
BT	Actividades
STAGE	P
CO	A
CD	05/07/2009
MD	05/11/2009
```

RT Documentación
RT Informática
RT Linguística
@@@
TERM RODA
NODE 713
STATUS M
BT Proyectos
STAGE P
CO A
CD 05/04/2009
MD 05/04/2009
SN Red de conocimiento descentralizado a través de anotaciones
RT Web semántica
@@@
TERM RSS
NODE 714
STATUS M
BT Lenguajes de sindicación
STAGE P
CO A
CD 05/05/2009
MD 05/05/2009
SN Really Simple Syndication
UF Really Simple Syndication
RT RDF
RT XML
@@@
TERM RuleML
NODE 715
STATUS M
BT Lenguajes de marcado
STAGE P
CO A
CD 05/04/2009
MD 05/04/2009
SN Rule Markup Language
RT RDF
RT Web semántica
RT XML
@@@
TERM sameAs
NODE 716
STATUS M
BT Caracteristicas igualdad-desigualdad
STAGE P
CO A
CD 04/30/2009
MD 04/30/2009
SN Dos individuos deben ser establecidos como lo mismo
UF Same As
@@@
TERM SAX
NODE 717
STATUS M
BT XML
STAGE P
CO A
CD 05/04/2009
MD 05/04/2009

SN Simple API for XML (Lenguaje que permite manipular objetos de un documento XML)
UF Simple API for XML
@@@
TERM SchemaLogic
NODE 718
STATUS M
BT Gestión tesauros
STAGE P
CO A
CD 04/29/2009
MD 05/04/2009
@@@
TERM Scope
NODE 719
STATUS M
BT Elementos
STAGE P
CO A
CD 04/30/2009
MD 04/30/2009
@@@
TERM Seq
NODE 720
STATUS M
BT Contenedores
STAGE P
CO A
CD 04/29/2009
MD 04/29/2009
SN Contenedor de RDF para referirse a un grupo de miembros donde es relevante el orden
@@@
TERM Sesame
NODE 721
STATUS M
BT JAVA
STAGE P
CO A
CD 05/05/2009
MD 05/05/2009
SN Framework de Java basado en código abierto
RT RDF
RT RDFS
RT XML
@@@
TERM SGML
NODE 722
STATUS M
BT Lenguajes de marcado
STAGE P
CO A
CD 04/30/2009
MD 04/30/2009
SN Standard Generalized Markup Language
RT HyTime
@@@
TERM Simétrico
NODE 723
STATUS M
BT Cualidades
STAGE P

CO A
CD 05/07/2009
MD 05/07/2009
RT Asimétrico
@@@
TERM Simple Topic Maps Management
NODE 724
STATUS M
BT Gestión Topic Maps
STAGE P
CO A
CD 04/30/2009
MD 05/04/2009
RT Topic maps
@@@
TERM Sistemas de gestión de bases de datos
NODE 725
STATUS M
BT Herramientas
STAGE P
CO A
CD 05/07/2009
MD 05/07/2009
UF SGBD
RT Documentación
RT Informática
RT Integración
@@@
TERM Sistemas de organización del conocimiento
NODE 726
STATUS M
STAGE P
CO A
CD 04/29/2009
MD 05/05/2009
NT Vocabularios controlados
UF KOS
RT Ontologías
RT OWL
RT RDF
RT Skos Core
RT SKOS Extensions
RT SKOS Mapping
RT Tesauros
RT Web semántica
@@@
TERM Sistemas operativos
NODE 727
STATUS M
STAGE P
CO A
CD 05/07/2009
MD 05/07/2009
NT Windows
NT Linux
UF SO
RT Informática
@@@
TERM Sistemática
NODE 728

STATUS M
BT Presentación
STAGE P
CO A
CD 04/29/2009
MD 04/29/2009
RT Notaciones
@@@
TERM SIS-TMS
NODE 729
STATUS M
BT Gestión tesauros
STAGE P
CO A
CD 04/29/2009
MD 05/04/2009
RT Tesauros
@@@
TERM SKOS
NODE 730
STATUS M
BT Vocabularios de Metadatos
STAGE P
CO A
CD 05/05/2009
MD 05/05/2009
SN Simple Knowledge Organisation System
UF Simple Knowledge Organisation System
RT Skos Core
@@@
TERM Skos Core
NODE 731
STATUS M
BT RDFS
STAGE P
CO A
CD 04/29/2009
MD 05/05/2009
RT Sistemas de organización del conocimiento
RT SKOS
@@@
TERM SKOS Extensions
NODE 732
STATUS M
BT RDFS
STAGE P
CO A
CD 04/29/2009
MD 04/29/2009
RT Sistemas de organización del conocimiento
@@@
TERM SKOS Mapping
NODE 733
STATUS M
BT RDFS
STAGE P
CO A
CD 04/29/2009
MD 04/29/2009
RT Sistemas de organización del conocimiento

```
@@@
TERM    SLIDE
NODE    734
STATUS  M
BT      Editores XML
STAGE   P
CO      A
CD      05/06/2009
MD      05/06/2009
SN      Convierte documentos escritos en SLiP (Short Hand Syntax) en documentos XML
RT      XML
@@@
TERM    SLiP
NODE    735
STATUS  M
BT      Lenguajes de marcado
STAGE   P
CO      A
CD      05/06/2009
MD      05/06/2009
SN      Sorta Like Python. Sintaxis alternativa para crear y editar datos XML a mano
UF      Sorta Like Python
RT      Proyectos
RT      Python
@@@
TERM    SOX
NODE    736
STATUS  M
BT      Lenguajes de marcado
STAGE   P
CO      A
CD      05/06/2009
MD      05/06/2009
SN      Simple Outline XML. Lenguaje XML simplificado
UF      Simple Outline XML
RT      XML
@@@
TERM    SPARQL
NODE    737
STATUS  M
BT      Lenguajes de recuperación
STAGE   P
CO      A
CD      05/04/2009
MD      05/06/2009
SN      Protocol and RDF Query Language
RT      RDF
RT      W3C
RT      Web semántica
@@@
TERM    SQL
NODE    738
STATUS  M
BT      Lenguajes de recuperación
STAGE   P
CO      A
CD      05/04/2009
MD      05/04/2009
SN      Structured Query Language
UF      Structured Query Language
```

RT TMQL
RT TOLOG
@@@
TERM Star/Thesaurus
NODE 739
STATUS M
BT Gestión tesauros
STAGE P
CO A
CD 05/04/2009
MD 05/04/2009
RT Tesauros
@@@
TERM StAX
NODE 740
STATUS M
BT XML
STAGE P
CO A
CD 05/05/2009
MD 05/05/2009
SN Streaming API for XML; aplicación para leer y escribir documentos de XML en Java
UF Streaming API for XML
@@@
TERM Stemming
NODE 741
STATUS M
BT Actividades
STAGE P
CO A
CD 05/06/2009
MD 05/14/2009
SN Método para reducir una palabra a su raíz o mejor a un stem o tema.
RT Linguística computacional
RT Terminología
@@@
TERM STRIDE
NODE 742
STATUS M
BT Gestión tesauros
STAGE P
CO A
CD 04/29/2009
MD 05/04/2009
RT Tesauros
@@@
TERM Style Master
NODE 743
STATUS M
BT Editores CSS
STAGE P
CO A
CD 05/06/2009
MD 05/06/2009
RT CSS
@@@
TERM Stylus Studio
NODE 744
STATUS M
BT Editores XML

STAGE P
CO A
CD 05/04/2009
MD 05/06/2009
RT DTD
RT XML
RT XML Schema
@@@
TERM SubClassOf
NODE 745
STATUS M
BT Property
STAGE P
CO A
CD 04/29/2009
MD 04/29/2009
RT OWL lite
@@@
TERM SVG
NODE 746
STATUS M
BT Lenguajes de descripción
STAGE P
CO A
CD 05/05/2009
MD 05/14/2009
SN Scalable Vector Graphics. Lenguaje para describir gráficos vectorales bidimensionales.
UF Scalable Vector Graphics
RT W3C
RT XML
@@@
TERM SWAD-Europe
NODE 747
STATUS M
BT Proyectos
STAGE P
CO A
CD 04/29/2009
MD 05/04/2009
SN Semantic Web Advanced Development for Europe
RT Web semántica
@@@
TERM SWOOP
NODE 748
STATUS M
BT Gestión tesauros
STAGE P
CO A
CD 04/29/2009
MD 05/04/2009
RT Tesauros
@@@
TERM SWSL
NODE 749
STATUS M
BT Lenguajes de descripción
STAGE P
CO A
CD 05/04/2009
MD 05/14/2009

SN Semantic Web Services Language. Lenguaje de descripción de servicios de la Web semántica. Consituido en 2 partes: SWSL-FOL (lógica de primer orden), and SWSL-Rules (reglas en las que se basa este lenguaje).
NT SWSLRules
NT SWSL-FOL
UF Semantic Web Services Language
RT Ontologías
RT Web semántica
@@@
TERM SWSL-FOL
NODE 750
STATUS M
BT SWSL
STAGE P
CO A
CD 05/04/2009
MD 05/14/2009
SN Lógica de primer orden de SWSL
@@@
TERM SWSLRules
NODE 751
STATUS M
BT SWSL
STAGE P
CO A
CD 05/04/2009
MD 05/14/2009
SN Reglas en las que se basa SWSL.
@@@
TERM SymmetricProperty
NODE 752
STATUS M
BT Identificadores especiales
STAGE P
CO A
CD 04/30/2009
MD 04/30/2009
UF symetric property
UF symetricproperty
UF symmetric property
@@@
TERM SymOntoX
NODE 753
STATUS M
BT Gestión ontologías
STAGE P
CO A
CD 05/04/2009
MD 05/04/2009
RT Ontologías
@@@
TERM Synaptica
NODE 754
STATUS M
BT Gestión tesauros
STAGE P
CO A
CD 04/29/2009
MD 05/04/2009
RT Tesauros

```
@@@
TERM    Tablas
NODE    755
STATUS  M
BT      Visualización
STAGE   P
CO      A
CD      04/30/2009
MD      04/30/2009
@@@
TERM    Taxonomías
NODE    756
STATUS  M
BT      Vocabularios controlados
STAGE   P
CO      A
CD      05/05/2009
MD      05/05/2009
@@@
TERM    TCS-9
NODE    757
STATUS  M
BT      Gestión tesauros
STAGE   P
CO      A
CD      04/29/2009
MD      05/04/2009
RT      Tesauros
@@@
TERM    TE
NODE    758
STATUS  M
BT      Notaciones
STAGE   P
CO      A
CD      04/29/2009
MD      04/29/2009
SN      Término Específico; relación jerárquica
@@@
TERM    TechnicalDescription
NODE    759
STATUS  M
BT      Esquemas para audio
STAGE   P
CO      A
CD      05/05/2009
MD      05/05/2009
UF      Technical Description
@@@
TERM    Técnico
NODE    760
STATUS  M
BT      Cualidades
STAGE   P
CO      A
CD      05/07/2009
MD      05/07/2009
@@@
TERM    TemaTres
NODE    761
```

STATUS M
BT Gestión tesauros
STAGE P
CO A
CD 04/29/2009
MD 05/04/2009
RT Tesauros
@@@
TERM term
NODE 762
STATUS M
BT Esquemas para tesauros
STAGE P
CO A
CD 05/04/2009
MD 05/04/2009
SN Describe brevemente un término por su nombre
RT Tesauros
@@@
TERM Term Tree 2000
NODE 763
STATUS M
BT Gestión tesauros
STAGE P
CO A
CD 04/29/2009
MD 05/04/2009
RT Tesauros
@@@
TERM TermChoir
NODE 764
STATUS M
BT Gestión tesauros
STAGE P
CO A
CD 04/29/2009
MD 05/04/2009
RT Tesauros
@@@
TERM term-description
NODE 765
STATUS M
BT Esquemas para tesauros
STAGE P
CO A
CD 05/04/2009
MD 05/04/2009
SN Describe más completamente un término que un term
RT Tesauros
@@@
TERM Terminología
NODE 766
STATUS M
BT Linguística
STAGE P
CO A
CD 05/07/2009
MD 05/11/2009
SN Conjunto de términos o vocablos propios de determinada profesión, ciencia o materia.
RT Stemming

RT Terminólogo
@@@
TERM Terminólogo
NODE 767
STATUS M
BT Profesiones
STAGE P
CO A
CD 05/07/2009
MD 05/07/2009
RT Terminología
@@@
TERM Tesauros
NODE 768
STATUS M
BT Vocabularios controlados
STAGE P
CO A
CD 04/29/2009
MD 04/29/2009
NT Presentación
NT Notaciones
NT Ejemplos
RT a.k.a. Classification Software
RT Amicus Thesaurus
RT Cognatrix
RT Data Harmony
RT error
RT extended
RT Gestión tesauros
RT hierarchy
RT list
RT Midos Thesaurus
RT MultiTes
RT NC-ISO 2788: 2000
RT Ontologías
RT properties
RT response
RT SIS-TMS
RT Sistemas de organización del conocimiento
RT Star/Thesaurus
RT STRIDE
RT SWOOP
RT Synaptica
RT TCS-9
RT TemaTres
RT term
RT Term Tree 2000
RT term-description
RT TermChoir
RT Tesaurus Builder
RT The Taxonomy Editor
RT Thesmain
RT ThManager
RT Tim Craven-Freeware
RT tmCake
RT Topic maps
RT Wordmap
@@@
TERM Tesaurus Builder

NODE 769
STATUS M
BT Gestión tesauros
STAGE P
CO A
CD 04/29/2009
MD 05/04/2009
RT Tesauros
@@@
TERM Tesinas
NODE 770
STATUS M
BT Documentos
STAGE P
CO A
CD 05/07/2009
MD 05/07/2009
RT Tesis
@@@
TERM Tesis
NODE 771
STATUS M
BT Documentos
STAGE P
CO A
CD 05/07/2009
MD 05/07/2009
RT Tesinas
@@@
TERM Text mining
NODE 772
STATUS M
BT Actividades
STAGE P
CO A
CD 05/06/2009
MD 05/14/2009
SN Extracción de información relevante, procedente de textos. Método de extracción de información de textos e inferencia de relaciones que no aparecen de forma implicita en esa informacion.
UF Minería del texto
UF Textmining
RT Recuperación de información
RT Web semántica
@@@
TERM TG
NODE 773
STATUS M
BT Notaciones
STAGE P
CO A
CD 04/29/2009
MD 04/29/2009
SN Término Genérico; Relación jerárquica
@@@
TERM The Ceryle Project
NODE 774
STATUS M
BT Gestión Topic Maps
STAGE P

```
CO      A
CD      04/30/2009
MD      05/04/2009
RT      Topic maps
@@@
TERM    The Taxonomy Editor
NODE    775
STATUS  M
BT      Gestión tesauros
STAGE   P
CO      A
CD      04/29/2009
MD      05/04/2009
RT      Tesauros
@@@
TERM    Thesmain
NODE    776
STATUS  M
BT      Gestión tesauros
STAGE   P
CO      A
CD      04/29/2009
MD      05/04/2009
RT      Tesauros
@@@
TERM    Thinkgraph
NODE    777
STATUS  M
BT      Visualización Topic Maps
STAGE   P
CO      A
CD      04/30/2009
MD      05/04/2009
@@@
TERM    ThManager
NODE    778
STATUS  M
BT      Gestión tesauros
STAGE   P
CO      A
CD      04/29/2009
MD      05/04/2009
RT      Tesauros
@@@
TERM    Tim Craven-Freeware
NODE    779
STATUS  M
BT      Gestión tesauros
STAGE   P
CO      A
CD      04/29/2009
MD      05/04/2009
RT      Tesauros
@@@
TERM    TM4Web
NODE    780
STATUS  M
BT      Visualización Topic Maps
STAGE   P
CO      A
```

```
CD      04/30/2009
MD      05/04/2009
@@@
TERM    tmCake
NODE    781
STATUS  M
BT      Gestión tesauros
STAGE   P
CO      A
CD      04/29/2009
MD      05/04/2009
RT      Tesauros
@@@
TERM    TMCL
NODE    782
STATUS  M
BT      Lenguajes de recuperación
STAGE   P
CO      A
CD      05/04/2009
MD      05/04/2009
SN      Topic Maps Constraint Language
UF      Topic Maps Constraint Language
RT      Topic maps
@@@
TERM    TML4Editor
NODE    783
STATUS  M
BT      Gestión Topic Maps
STAGE   P
CO      A
CD      04/30/2009
MD      05/04/2009
RT      Topic maps
@@@
TERM    TMQL
NODE    784
STATUS  M
BT      Lenguajes de recuperación
STAGE   P
CO      A
CD      05/04/2009
MD      05/04/2009
SN      Topic Maps Query Language
UF      Topic Maps Query Language
RT      ISO/IEC FCD 18048
RT      SQL
RT      Topic maps
@@@
TERM    TMTab
NODE    785
STATUS  M
BT      Gestión Topic Maps
STAGE   P
CO      A
CD      04/30/2009
MD      05/04/2009
RT      Topic maps
@@@
TERM    TMView
```

NODE 786
STATUS M
BT Visualización Topic Maps
STAGE P
CO A
CD 04/30/2009
MD 05/04/2009
@@@
TERM TOLOG
NODE 787
STATUS M
BT Lenguajes de recuperación
STAGE P
CO A
CD 05/04/2009
MD 05/04/2009
SN Topic map query language
UF Topic map query language
RT SQL
RT Topic maps
@@@
TERM TopBraid Composer
NODE 788
STATUS M
BT Gestión ontologías
STAGE P
CO A
CD 05/04/2009
MD 05/04/2009
RT Ontologías
@@@
TERM Topic
NODE 789
STATUS M
BT Elementos
STAGE P
CO A
CD 04/30/2009
MD 05/05/2009
NT Topic types
UF Elemento principal
UF Topic map elemento principal
@@@
TERM Topic Map Designer
NODE 790
STATUS M
BT Visualización Topic Maps
BT Gestión Topic Maps
STAGE P
CO A
CD 04/30/2009
MD 05/04/2009
RT Topic maps
@@@
TERM Topic maps
NODE 791
STATUS M
BT Vocabularios controlados
STAGE P
CO A

CD 04/30/2009
MD 04/30/2009
NT Visualización
NT PSI
NT Elementos
RT ATop
RT Gestión Topic Maps
RT GNOWSYS
RT ISO/IEC 13250:2000
RT ISO/IEC FCD 18048
RT Mapalizer
RT Ontologías
RT Simple Topic Maps Management
RT Tesauros
RT The Ceryle Project
RT TMCL
RT TML4Editor
RT TMQL
RT TMTab
RT TOLOG
RT Topic Map Designer
RT Topincs
RT Wandora
@@@
TERM Topic types
NODE 792
STATUS M
BT Topic
STAGE P
CO A
CD 04/30/2009
MD 04/30/2009
SN Definen relaciones clase-instancia
UF topictype
@@@
TERM Topincs
NODE 793
STATUS M
BT Gestión Topic Maps
STAGE P
CO A
CD 04/30/2009
MD 05/04/2009
RT Topic maps
@@@
TERM TR
NODE 794
STATUS M
BT Notaciones
STAGE P
CO A
CD 04/29/2009
MD 04/29/2009
SN Término Relacionado; relación asociativa
@@@
TERM Traducción automática
NODE 795
STATUS M
BT Linguística computacional
STAGE P

CO A
CD 05/07/2009
MD 05/11/2009
SN Área de la lingüística computacional que investiga el uso de software para traducir texto o habla de un lenguaje natural a otro.
UF Machine Translation
UF MT
UF TA
RT Informática
@@@
TERM TransitiveProperty
NODE 796
STATUS M
BT Identificadores especiales
STAGE P
CO A
CD 04/30/2009
MD 04/30/2009
UF tansitivepropperti
UF transitive property
UF transitive propperty
@@@
TERM Type
NODE 797
STATUS M
BT Property
STAGE P
CO A
CD 04/29/2009
MD 04/29/2009
@@@
TERM UML
NODE 798
STATUS M
BT Lenguajes de modelado
STAGE P
CO A
CD 05/05/2009
MD 05/05/2009
SN Lenguaje de Modelado Unificado
UF Lenguaje de Modelado Unificado
@@@
TERM UN/EDIFACT
NODE 799
STATUS M
BT Estándares
STAGE P
CO A
CD 05/06/2009
MD 05/06/2009
SN United Nations/Electronic Data Interchange For Administration, Commerce, and Transport. Estándar de EDI (Electronic Data Interchange) desarrollado por las Naciones Unidas
UF Electronic Data Interchange For Administration, Commerce, and Transport.
RT ANSI ASCX12
RT EDI
RT Naciones Unidas
@@@
TERM unionOf
NODE 800
STATUS M

BT Combinaciones booleanas
STAGE P
CO A
CD 04/30/2009
MD 04/30/2009
UF union of
UF union off
UF unionoff
@@@
TERM Universidades
NODE 801
STATUS M
BT Instituciones
STAGE P
CO A
CD 05/07/2009
MD 05/07/2009
@@@
TERM UP
NODE 802
STATUS M
BT Notaciones
STAGE P
CO A
CD 04/29/2009
MD 04/29/2009
SN Usado Por; relación de equivalencia
@@@
TERM URI
NODE 803
STATUS M
BT Estándares
STAGE P
CO A
CD 04/29/2009
MD 05/11/2009
SN Uniform Resource Identifier; identificador uniforme de recurso.
NT URL
@@@
TERM URL
NODE 804
STATUS M
BT URI
STAGE P
CO A
CD 04/29/2009
MD 04/29/2009
SN Uniform Resource Locator; localizador uniforme de recurso
NT News
NT Mailto
NT File
@@@
TERM USE
NODE 805
STATUS M
BT Notaciones
STAGE P
CO A
CD 04/29/2009
MD 04/29/2009

```
SN      Relación de equivalencia
@@@
TERM    VAN
NODE    806
STATUS  M
BT      EDI
STAGE   P
CO      A
CD      05/06/2009
MD      05/06/2009
SN      Value Added Networks (Red de Valor Añadido)
UF      Red de Valor Añadido
UF      Value Added Networks
@@@
TERM    Variantes
NODE    807
STATUS  M
BT      OWL
STAGE   P
CO      A
CD      04/30/2009
MD      04/30/2009
NT      OWL lite
NT      OWL full
NT      OWL DL
@@@
TERM    VBScript
NODE    808
STATUS  M
BT      Lenguajes de programación
STAGE   P
CO      A
CD      05/05/2009
MD      05/05/2009
SN      Visual Basic Script Edition
UF      Visual Basic
UF      Visual Basic Script Edition
@@@
TERM    Visualización
NODE    809
STATUS  M
BT      Topic maps
STAGE   P
CO      A
CD      04/30/2009
MD      04/30/2009
NT      Árboles
NT      Tablas
NT      Multidimensional
NT      Mapas
NT      Grafos
@@@
TERM    Visualización Topic Maps
NODE    810
STATUS  M
BT      Herramientas
STAGE   P
CO      A
CD      05/04/2009
MD      05/04/2009
```

NT Topic Map Designer
NT TMView
NT TM4Web
NT Thinkgraph
NT Panckoucke
NT HyperGraph
@@@
TERM Vocabularios controlados
NODE 811
STATUS M
BT Sistemas de organización del conocimiento
STAGE P
CO A
CD 04/29/2009
MD 04/30/2009
NT Topic maps
NT Tesauros
NT Taxonomías
NT Ontologías
NT Glosarios
NT Encabezamientos de materias
NT Diccionarios
@@@
TERM Vocabularios de Metadatos
NODE 812
STATUS M
STAGE P
CO A
CD 05/04/2009
MD 05/05/2009
NT SKOS
NT LOM
NT Dublin Core
UF Metadatos
@@@
TERM W3C
NODE 813
STATUS M
BT Organismos
STAGE P
CO A
CD 04/29/2009
MD 05/14/2009
SN World Wide Web Consortium. es un consorcio internacional donde las organizaciones miembro, personal a tiempo completo y el público en general, trabajan conjuntamente para desarrollar estándares Web con el objetivo de sacar el máximo potencial a través del desarrollo de protocolos y pautas.
UF World Wide Web Consortium
RT DOM
RT Estándares
RT OWL
RT RDF
RT SPARQL
RT SVG
RT Web semántica
RT XLink
RT XPointer
RT XSL-FO
RT XSLT
@@@

TERM Wandora
NODE 814
STATUS M
BT Gestión Topic Maps
STAGE P
CO A
CD 04/30/2009
MD 05/04/2009
RT Topic maps
@@@
TERM WBMP
NODE 815
STATUS M
BT Esquemas para imágenes
STAGE P
CO A
CD 05/05/2009
MD 05/05/2009
SN Wireless BotMap
UF Wireless BotMap
@@@
TERM Web 1.0
NODE 816
STATUS M
BT Redes
STAGE P
CO A
CD 05/07/2009
MD 05/07/2009
@@@
TERM Web 2.0
NODE 817
STATUS M
BT Redes
STAGE P
CO A
CD 05/07/2009
MD 05/11/2009
UF Web social
@@@
TERM Web semántica
NODE 818
STATUS M
BT Redes
STAGE P
CO A
CD 04/29/2009
MD 05/11/2009
UF SW
UF Web 3.0
RT Buscadores documentos XML
RT Data mining
RT Data warehouse
RT Gestión ontologías
RT HP Labs Semantic Web Programme
RT Jena
RT Lenguajes de marcado
RT NC-ISO 2788: 2000
RT Ontologías
RT OWL

```
RT      Procesamiento del lenguaje natural
RT      RDF
RT      Recuperación de información
RT      RODA
RT      RuleML
RT      Sistemas de organización del conocimiento
RT      SPARQL
RT      SWAD-Europe
RT      SWSL
RT      Text mining
RT      W3C
RT      WS-CDL
RT      WSDL
RT      WSML
RT      WSMO
RT      XML
@@@
TERM    WebChoir
NODE    819
STATUS  M
BT      Gestión tesauros
STAGE   P
CO      A
CD      04/29/2009
MD      05/04/2009
@@@
TERM    WebODE
NODE    820
STATUS  M
BT      Gestión ontologías
STAGE   P
CO      A
CD      05/04/2009
MD      05/04/2009
RT      Ontologías
@@@
TERM    WebOnto
NODE    821
STATUS  M
BT      Gestión ontologías
STAGE   P
CO      A
CD      05/04/2009
MD      05/04/2009
RT      Ontologías
@@@
TERM    Windows
NODE    822
STATUS  M
BT      Sistemas operativos
STAGE   P
CO      A
CD      05/07/2009
MD      05/07/2009
@@@
TERM    WML
NODE    823
STATUS  M
BT      Lenguajes de marcado
STAGE   P
```

CO A
CD 05/05/2009
MD 05/05/2009
SN Wireless Markup Language
UF Wireless Markup Language
RT XML
@@@
TERM Wordmap
NODE 824
STATUS M
BT Gestión tesauros
STAGE P
CO A
CD 04/29/2009
MD 05/04/2009
RT Tesauros
@@@
TERM WS-CDL
NODE 825
STATUS M
BT Lenguajes de descripción
STAGE P
CO A
CD 05/04/2009
MD 05/14/2009
SN Web Services Description Language. Lenguaje basado en XML que describe de igual a igual de colaboración de las partes.
UF Web Services Choreography Description Language
RT Ontologías
RT Web semántica
RT XML
@@@
TERM WSDL
NODE 826
STATUS M
BT Lenguajes de descripción
STAGE P
CO A
CD 05/04/2009
MD 05/14/2009
SN Web Services Description Language. Formato XML para describir servicios web.
UF Web Services Description Language
RT Ontologías
RT Web semántica
@@@
TERM WSMF
NODE 827
STATUS M
BT Lenguajes de descripción
STAGE P
CO A
CD 05/04/2009
MD 05/14/2009
SN Marco de Modelado de Servicios Web, para transformar la web en en un dispositivo de computación distribuida.
UF Marco de Modelado de Servicios Web
RT Ontologías
@@@
TERM WSML
NODE 828

STATUS M
BT Lenguajes de modelado
STAGE P
CO A
CD 05/04/2009
MD 05/04/2009
SN Web Service Modeling Language
NT WSML-Rule
NT WSML-Full
NT WSML-Flight
NT WSML-DL
NT WSML-Core
UF Web Service Modeling Languag
RT Ontologías
RT Web semántica
RT WSMO
@@@
TERM WSML-Core
NODE 829
STATUS M
BT WSML
STAGE P
CO A
CD 05/04/2009
MD 05/04/2009
@@@
TERM WSML-DL
NODE 830
STATUS M
BT WSML
STAGE P
CO A
CD 05/04/2009
MD 05/04/2009
@@@
TERM WSML-Flight
NODE 831
STATUS M
BT WSML
STAGE P
CO A
CD 05/04/2009
MD 05/04/2009
@@@
TERM WSML-Full
NODE 832
STATUS M
BT WSML
STAGE P
CO A
CD 05/04/2009
MD 05/04/2009
@@@
TERM WSML-Rule
NODE 833
STATUS M
BT WSML
STAGE P
CO A
CD 05/04/2009

```
MD      05/04/2009
@@@
TERM    WSMO
NODE    834
STATUS  M
BT      Lenguajes de modelado
STAGE   P
CO      A
CD      05/04/2009
MD      05/04/2009
SN      Web Service Modeling Language
UF      Web Service Modeling Language
RT      Ontologías
RT      RDF
RT      Web semántica
RT      WSML
RT      XML
@@@
TERM    XEmacs
NODE    835
STATUS  M
BT      Editores XML
STAGE   P
CO      A
CD      05/04/2009
MD      05/04/2009
RT      DTD
RT      XML
RT      XML Schema
@@@
TERM    XFORMS
NODE    836
STATUS  M
BT      XML
STAGE   P
CO      A
CD      05/05/2009
MD      05/05/2009
SN      Lenguaje de marcas para formularios  Web
@@@
TERM    XHTML
NODE    837
STATUS  M
BT      Lenguajes de marcado
STAGE   P
CO      A
CD      05/05/2009
MD      05/05/2009
SN      Extensible Hypertext Markup Language
UF      Extensible Hypertext Markup Language
RT      HTML
RT      XML
@@@
TERM    XLink
NODE    838
STATUS  M
BT      XML
STAGE   P
CO      A
CD      05/04/2009
```

MD 05/06/2009
SN Lenguaje de vínculos XML
UF Lenguaje de vínculos XML
RT W3C
@@@
TERM XMI
NODE 839
STATUS M
BT XML
STAGE P
CO A
CD 05/05/2009
MD 05/05/2009
SN XML Metadata Interchange
UF XML Metadata Interchange
@@@
TERM XML
NODE 840
STATUS M
BT Lenguajes de marcado
STAGE P
CO A
CD 04/29/2009
MD 04/30/2009
SN Extensible Markup Language
NT XSL
NT XScore
NT XQL
NT XPointer
NT XMP
NT XML Schema
NT XMI
NT XLink
NT XFORMS
NT StAX
NT SAX
NT JDOM
NT DTD
NT DOM
NT CSS
UF Extensible Markup Language
RT Altova XMLSpy
RT ATOM
RT Buscadores documentos XML
RT HTML
RT JDOM
RT LOM
RT Microsoft Visual Studio
RT Oxygen XML Editor
RT RDF
RT RSS
RT RuleML
RT Sesame
RT SLIDE
RT SOX
RT Stylus Studio
RT SVG
RT Web semántica
RT WML
RT WS-CDL

```
RT      WSMO
RT      XEmacs
RT      XHTML
RT      Xquery
RT      YAML
@@@
TERM    XML Schema
NODE    841
STATUS  M
BT      XML
STAGE   P
CO      A
CD      05/04/2009
MD      05/04/2009
NT      Esquemas para tesauros
NT      Esquemas para imágenes
NT      Esquemas para audio
RT      Altova XMLSpy
RT      Microsoft Visual Studio
RT      Stylus Studio
RT      XEmacs
@@@
TERM    xml2ogdl
NODE    842
STATUS  M
BT      Representación de grafos
STAGE   P
CO      A
CD      05/05/2009
MD      05/05/2009
@@@
TERM    XMP
NODE    843
STATUS  M
BT      XML
STAGE   P
CO      A
CD      05/05/2009
MD      05/05/2009
SN      Plataforma Extensible de Metadatos
NT      XMP Rights Managemente Schema
NT      XMP Media Managemente Schema
NT      XMP Basic Schema
UF      Plataforma Extensible de Metadatos
@@@
TERM    XMP Basic Schema
NODE    844
STATUS  M
BT      XMP
STAGE   P
CO      A
CD      05/05/2009
MD      05/05/2009
@@@
TERM    XMP Media Managemente Schema
NODE    845
STATUS  M
BT      XMP
STAGE   P
CO      A
```

CD 05/05/2009
MD 05/05/2009
@@@
TERM XMP Rights Managemente Schema
NODE 846
STATUS M
BT XMP
STAGE P
CO A
CD 05/05/2009
MD 05/05/2009
@@@
TERM XPath
NODE 847
STATUS M
BT XSL
STAGE P
CO A
CD 05/04/2009
MD 05/05/2009
SN XML Path Language (lenguaje que sirve para acceder a partes concretas de un documento XML)
UF XML Path Language
@@@
TERM XPointer
NODE 848
STATUS M
BT XML
STAGE P
CO A
CD 05/04/2009
MD 05/04/2009
SN Lenguaje de punteros XML
UF Lenguaje de punteros XML
RT W3C
@@@
TERM XQL
NODE 849
STATUS M
BT XML
STAGE P
CO A
CD 05/04/2009
MD 05/04/2009
SN XML Query Language (Lenguaje de consulta que permitirá hacer búsquedas sobre documentos)
UF XML Query Language
@@@
TERM Xquery
NODE 850
STATUS M
BT Lenguajes de recuperación
STAGE P
CO A
CD 05/05/2009
MD 05/05/2009
SN Permite obtener datos de documentos XML
RT XML
@@@
TERM XScore

NODE 851
STATUS M
BT XML
STAGE P
CO A
CD 05/05/2009
MD 05/05/2009
SN Extensible Store Language
@@@
TERM XSL
NODE 852
STATUS M
BT XML
STAGE P
CO A
CD 05/04/2009
MD 05/04/2009
SN Extensible Stylesheet Language (Lenguaje Extensible de Hojas de Estilo)
NT XSLT
NT XSL-FO
NT XPath
UF Extensible Stylesheet Language
@@@
TERM XSL-FO
NODE 853
STATUS M
BT XSL
STAGE P
CO A
CD 05/05/2009
MD 05/05/2009
SN Extensible Stylesheet Language Formatting Objects
UF Extensible Stylesheet Language - Formatting Objects
RT W3C
@@@
TERM XSLT
NODE 854
STATUS M
BT XSL
STAGE P
CO A
CD 05/05/2009
MD 05/05/2009
SN Extensible Stylesheet Language Transformations
UF Extensible Stylesheet Language Transformations
RT W3C
@@@
TERM XWI
NODE 855
STATUS M
BT Esquemas para imágenes
STAGE P
CO A
CD 05/05/2009
MD 05/05/2009
SN XML FOR WIRELESS IMAGES
UF XML FOR WIRELESS IMAGES
@@@
TERM YAML
NODE 856

STATUS M
BT Lenguajes de marcado
STAGE P
CO A
CD 05/05/2009
MD 05/14/2009
SN Yet Ain't Markup Language. Lenguaje de marcado similar a XML, pero con menor implantación.
UF Yet Ain't Markup Language
RT XML