TESIS DEFENDIDA POR

Angel Almaraz Mota

Y APROBADA POR EL SIGUIENTE COMITÉ

M. C. José Luis Briseño Cervantes
Co-Director del Comité

Dr. Gabriel Alejandro López Morteo
Co-Director del Comité

Dr. Pedro Gilberto López Mariscal
Miembro del Comité

Dr. Hugo Homero Hidalgo Silva
Miembro del Comité

Dr. Ricardo Arturo Chávez Pérez
Miembro del Comité

Dr. Pedro Gilberto López Mariscal
Coordinador del programa de posgrado en Ciencias de la Computación

Dr. David Hilario Covarrubias Rosales
Director de Estudios de Posgrado

6 de Noviembre de 2008

CENTRO DE INVESTIGACIÓN CIENTÍFICA Y DE EDUCACIÓN SUPERIOR DE ENSENADA

PROGRAMA DE POSGRADO EN CIENCIAS
EN CIENCIAS DE LA COMPUTACIÓN

BÚSQUEDAS FEDERADAS SIGNIFICATIVAS EN REPOSITORIOS DE OBJETOS DE APRENDIZAJE

TESIS

que para cubrir parcialmente los requisitos necesarios para obtener el grado de

MAESTRO EN CIENCIAS

Presenta:

ANGEL ALMARAZ MOTA

Ensenada, Baja California, México, Noviembre de 2008.

RESUMEN de la tesis de **Angel Almaraz Mota**, presentada como requisito parcial para la obtención del grado de MAESTRO EN CIENCIAS en CIENCIAS DE LA COMPUTACIÓN. Ensenada, Baja California. Noviembre de 2008.

BÚSQUEDAS FEDERADAS SIGNIFICATIVAS EN REPOSITORIOS DE OBJETOS DE APRENDIZAJE

Resumen aprobado por:

_____ _____

Dr. Gabriel Alejandro López Morteo — Co-Director de Tesis

M.C. José Luis Briseño Cervantes — Co-Director de Tesis

Actualmente la información se encuentra distribuida en diferentes sitios, donde cada uno de éstos puede estar regido por diversos mecanismos de acceso que restringen la consulta de la información por parte de los usuarios. Esta restricción origina que los usuarios conozcan a la perfección o de una manera general, tanto los sistemas que almacenan la información, como la ubicación de dichos sistemas y la manera en la que se encuentran estructurados los datos a los que pretenden acceder. El conjunto de instrucciones antes mencionadas es llamado búsqueda federada, el cual se refiere a un mecanismo que provee acceso a información residente en lugares heterogéneos y distribuidos, proveyendo de una interfaz unificada que interactúa con estos lugares disparando procesos concurrentes y presentando el resultado como un solo proceso. Dicho proceso de búsqueda es llevado a cabo en una federación, la cual, en el contexto de computación, es definida como un conjunto de sistemas de información capaz de proveer interoperabilidad, total o parcial, entre los elementos de dicho conjunto con el propósito de alcanzar un objetivo en común.

En el presente trabajo, se define una arquitectura de federación basada en mediadores capaz de llevar a cabo el proceso de búsqueda federada, utilizando estándares y protocolos de comunicación que permiten la interoperabilidad entre los distintos repositorios de información, enfocándose específicamente a la búsqueda y recuperación de los metadatos de objetos de aprendizaje (OA). El objeto de aprendizaje es un componente que asiste en el aprendizaje del usuario, dicho objeto esta formado por un componente digital descrito por metadatos. Los metadatos son elementos que proveen un primer acercamiento con el objeto de aprendizaje, además de fungir como descriptores del objeto al cual están asociados; entre la información que contienen los metadatos se encuentra la ubicación física del OA, permitiendo de esta manera, la localización y el posible acceso al mismo.

Las pruebas realizadas a los mecanismos y procesos implementados bajo la arquitectura propuesta demuestran que, en caso de existir metadatos cuyos elementos descriptores permitan determinar si son relevantes o no respecto a la cadena inicial de

búsqueda, éstos son recuperados y entregados en una lista ordenada donde en los primeros lugares se encuentran los metadatos mas relevantes; esta afirmación es corroborada en base a que en la mayoría de las consultas realizadas, los metadatos recuperados que son más relevantes respecto a esa cadena, son presentados en los primeros lugares de la lista generada. El criterio de relevancia utilizado es la cantidad de ocurrencia de palabras de búsqueda en el metadato, es decir, mientras más veces aparezca la cadena de búsqueda en dicho metadato, entonces, éste será más relevante respecto a esa cadena.

Palabras Clave: Federación, Búsquedas Federadas, Sistemas de Información, Objetos de Aprendizaje, Repositorios.

ABSTRACT of the thesis presented by **Angel Almaraz Mota** as a partial requirement to obtain the MASTER OF SCIENCE degree in COMPUTER SCIENCES. Ensenada, Baja California, México. November 2008.

SIGNIFICANT FEDERATED SEARCH IN LEARNING OBJECTS REPOSITORIES

Currently, information is distributed in different places, which can be ruled by different mechanisms that restrict the access to the information. This restriction requires that the users know exactly or in a general way, the systems that store the information, the location of these systems and how the data is structured. These set of instructions are known as a federated search, referring to the mechanisms that provide the access to the distributed information residing in different places, providing a unified interface that interacts with these locations, firing concurrent processes and presenting the results as a single process. The search process is conducted in a federation, which, in the computing context, is defined as a set of information systems that are able to provide total or partial interoperability between the elements of this set with the aim of reaching a common goal.

In this work, a federation architecture based on mediators capable of carrying out the federated search process, using standards and communications protocols that enable interoperability between different repositories is defined. It specifically focuses on the search and recovery of learning objects (LO) metadata. The learning object is a component that assists the user in learning that object and it is made up of a digital component described by metadata. Metadata are elements that provide an initial insight of the learning object, in addition to serving as descriptors of the object to which they are associated with.; among the information contained by the metadata, is the physical location of the LO, allowing in this way, to find its location and possibly accessing it.

The tests carried out to the mechanisms and the implemented processes under the proposed architecture, show that if there are metadata with descriptor elements capable of determining its relevance with respect the initial query search, they are retrieved and delivered in an ordered list where the first places are the most relevant metadata; this statement is corroborated on the basis that on the majority of queries, the retrieved metadata which is most relevant to these queries, are presented in the top of the generated list. The relevant criteria used, was the number of occurrences of query terms in the metadata, that is, the more times that appears the query term in the metadata, then it will be more relevant to that query..

Keywords: Federation, Federated Search, Information Systems, Learning Objects, Repositories.

A ti Angelito:

Por motivarme a seguir cada mañana, por las palabras de aliento que me das en tu idioma cuando platico contigo y por llenar mi vida. Este logro es por ti.

Que Dios te bendiga hijo mío.

A mis padres:

Mamá: ¡¡¡ eres lo máximo, simplemente la mejor !!!

Papá: ¡¡¡ Lo logramos !!! ¡Te quiero mucho mi viejo!

A mis hermanos:

Cecilia y Edson: ¡los quiero mucho hermanos! Son lo '*in*'.

Agradecimientos

A Dios por ayudarme y permitirme haber llegado a un logro más en mi vida.

A mi co-director de tesis José Luis Briseño Cervantes, por sus comentarios, opiniones, consejos y por siempre estar dispuesto a ayudarme y apoyarme asi como a tener una platica amena en cualquier momento. Es usted grande profe.

A mi co-director de tesis Gabriel Alejandro López Morteo, gracias por tus comentarios, sugerencias, opiniones, consejos, apoyo, regaños tan sutiles que no se percibían como tal (pero que dolían en lo más hondo), en fin… Muchas gracias por todo Doc, eres una persona digna de admirar. Gracias por permitirme conocerte.

A mi comité de tesis, por sus comentarios y observaciones que ayudaron a la realización de este trabajo.

Al *superpoderosísimo* círculo de las intrigas o al consultorio y sus pacientes o como quieran decirle, ustedes saben a lo que me refiero. Gracias por su amistad y por permitirme conocerlos. Alfonso, Antonio, César, Daniela, David, Gamaliel, Gustavo, Jehovani, José Luis, Luis, Pablo, Raúl, Rolando, Salvador, junto a ustedes he pasado momentos muy angustiosos pero que al final valieron la pena (aparecen por orden alfabético para que no haya sentimentalismos).

A la *chilango-banda*: Emmanuelle, Leonardo, Rodolfo, René y Lore. Gracias por su ayuda, por su amistad y en fin… ¡Gracias por todo Brothers!

Qué bueno que fui a Mexicali, ¿verdad? Gracias por todo y ser quien eres Denny.
¡Te quiero mucho!

A los nuevos amigos: Nancy, Tavo y… creo que son todos. Gracias por aguantar mi fiesta en el cubo. Tavo gracias por ser tan *pro*, Nancy gracias por permitirme conocerte y por siempre tener un sonrisa en tu rostro.

Al Centro de Investigación Científica y de Educación Superior de Ensenada (CICESE) por permitirme haber pertenecido a tan prestigiosa institución.

Y finalmente al Consejo Nacional de Ciencia y Tecnologia (Conacyt), *¡por la lana papá!*
¡¡¡ No te acabes Beca Conacyt, no te acabes !!!

Ensenada, Baja California, México
Noviembre de 2008

Angel Almaraz Mota

CONTENIDO

Página

CONTENIDO (continuación)

Página

CONTENIDO (continuación)

Página

LISTA DE FIGURAS

LISTA DE FIGURAS (continuación)

Figura Página

LISTA DE TABLAS

Capítulo I

Introducción

I.1. Introducción

La búsqueda de conocimiento en la actualidad se basa, fundamentalmente, en el uso de la Web, originando que existan aplicaciones capaces de buscar información en el espacio virtual del WWW (World Wide Web).

Lamentablemente no toda la información encontrada es de interés para quien la busca, o la que se encuentra no es fácilmente accesible, debido a que ésta es de carácter privado; por lo que es necesario utilizar otro tipo de búsqueda que sea capaz de acceder a la información que se encuentra escondida en la Web.

Las herramientas de búsqueda actuales basadas en Web como Google o Altavista, funcionan en base a listados almacenados en sus respectivas bases de datos. Dichos listados permiten que los buscadores realicen una exploración previa en cada uno de los elementos que conforman la lista con el propósito de capturar información referente a los mismos y de esta manera generar sus respectivos metadatos (en los cuales se realiza la indexación para las búsquedas provenientes de los usuarios finales de los buscadores). La idea de la búsqueda federada utilizando buscadores Web, implica realizar una búsqueda en diversos buscadores Web e integrar los resultados como uno solo; sin embargo, dada la gran cantidad de información existente y que parte de ella pueda existir en recursos que no estén

disponibles vía Web o que pertenezcan a colecciones privadas, la totalidad de la información no puede ser indexada por los buscadores Web. Debido a lo anterior, es necesario emplear más de un buscador con el propósito de abarcar una mayor cantidad del universo de contenido disponible en el Web originando que el usuario deba tener acceso a cada sistema de búsqueda que desee utilizar, así como el conocer las singularidades de cada sistema empleado. Aunado a esto existe el detalle de que cada sistema emplea metadatos, indexa sus documentos y presenta sus resultados de manera distinta.

La búsqueda federada es un mecanismo encargado de proveer un acceso unificado a sistemas de información distribuidos y heterogéneos, permitiendo el acceso a información que se encuentra en un formato diferente o incompatible al medio de búsqueda; en otras palabras, la búsqueda federada permite que los diversos sistemas de información sean capaces de interoperar entre ellos y de esta manera poder acceder a información residente en los mismos que de cualquier otra manera no pudiera ser consultada.

El concepto de federación existe en diversos contextos; sin embargo, en el entorno de computación puede ser comprendido como una integración de sistemas de cómputo heterogéneos y distribuidos, en donde la característica de la federación es la cooperación entre sistemas independientes permitiendo una integración controlada y, en ocasiones limitada. En cada unos de estos sistemas, se encuentra almacenada la información que no puede ser indexada por los buscadores Web tradicionales, debido a la incompatibilidad de los formatos de dicha información con los procesos de indexación del buscador, así como de la disponibilidad en línea de la misma.

Los lugares donde existe la información escondida son los diversos repositorios de información distribuidos en una región o incluso a nivel mundial. Existen repositorios de diferente naturaleza, como los repositorios informáticos que cuentan con una gran capacidad y velocidad para buscar, clasificar, analizar, relacionar y distribuir grandes volúmenes de información que permiten tomar decisiones estratégicas efectivas para las organizaciones basadas en conocimientos críticos; y los repositorios de acceso abierto para

documentos científicos y técnicos publicados y no publicados en librerías, en las áreas de Ciencias de la Información, tecnología, y áreas relacionadas.

I.2. Investigación previa

La búsqueda federada en repositorios de información para el WWW, empezó en 1998 cuando WebFeat (2007) tomó la idea simple de permitir a diversas bibliotecas buscar en alguna o en todas sus bases de datos al mismo tiempo, mediante una interfaz de usuario simple y común, y convirtió esta idea en un producto.

Desde entonces, con una gran cantidad de información disponible en línea así como información que no puede ser accedida mediante buscadores Web y dada la popularidad de otros motores de búsqueda, la necesidad de los productos de búsqueda federada continúa en crecimiento.

I.2.1. Iniciativas de búsqueda federada

Existen iniciativas que se han enfocado en la búsqueda federada; cada iniciativa presenta componentes que los caracterizan así como ciertos componentes que son similares entre todas ellas. A continuación se mencionan algunas iniciativas de búsqueda federada, presentando ciertas características sobre las mismas con el propósito de poder comprender las diferencias y similitudes entre las mismas.

RDN Subject Portals Project (SPP). El cual desarrolló una funcionalidad de portal para cinco de los ocho temas de la red de descubrimiento de recursos, con el objetivo de proveer acceso más fácil a recursos de información de alta calidad utilizando perfiles de usuario y servicios adicionales. Este proyecto fue administrado por UKOLN, donde fue terminada su segunda fase en Agosto de 2004 (JISC, 2002).

Middleware for Distributed Cognition (MDC). Este proyecto desarrolló un medio para académicos y estudiantes para buscar un rango de catálogos de referencia y después construyó una lista de lectura que podía ser impresa, cargada en un sitio Web o almacenada en línea. El proyecto fue desarrollado por la universidad de Oxford y la universidad de Edinburgh. MDC busca a través de diferentes tipos de bases de datos (como Z39.50) y consulta diferentes tipos de metadatos, tales como Dublin Core y LOM. El software fue diseñado de manera que pueda ser incrustado en ambientes de aprendizaje y fue terminado en Noviembre de 2004 (JISC, 2004).

Resource List Toolkit. Este proyecto produjo una herramienta de desarrollo de software para mediar en la reutilización de listas de recursos. Estas listas serían creadas y almacenadas en sistemas distribuidos y heterogéneos, tales como herramientas de aprendizaje electrónico, portales, sistemas de listas de recursos dedicados y repositorios. El proyecto fue liderado por la Universidad de Edinburgh y terminado en Octubre de 2005 (JISC, 2005).

Accessing and Storing Knowledge (ASK). Este proyecto tiene como propósito el permitir a los usuarios realizar una búsqueda federada sobre repositorios que implementan los protocolos Z39.50, SRU y SRW. Algunos de los repositorios incluidos en este proyecto son DSpace, ePrints y Fedora, además el usuario puede personalizar la búsqueda al definir las características de la misma, tales como el tipo de documentos a buscar y los lugares en los cuales buscar. El proyecto pretende utilizar un gran tipo de recursos que incluyen imágenes, documentos, listas de lectura y el componente IMS CP. Este proyecto fue coordinado por los servicios de cómputo de la Universidad de Oxford y terminado en Mayo de 2007. (JISC, 2007).

Las iniciativas mencionadas anteriormente proveen de una interfaz unificada de acceso a diversos sistemas de información heterogéneos y distribuidos, además de que son capaces de presentar los resultados de cada uno de ellos como si fuera uno solo; sin embargo, cada iniciativa presenta características que lo diferencian entre las otras por ejemplo: la creación

de perfiles de usuario del proyecto SPP, mediante éstos es posible realizar una búsqueda especializada basándose en los intereses del usuario final; la personalización de la búsqueda del proyecto ASK determinando los lugares en los cuales realizar la misma, así como el tipo de resultados a recuperar; entre otras.

Estos proyectos implementan el protocolo Z, el cual se encarga de establecer una conexión entre los clientes y servidores permitiendo de esta manera la recuperación de los registros o documentos; sin embargo, la implementación de este protocolo es un tanto complicada debido a la complejidad de las especificaciones y requerimientos que requiere el mismo. Debido a que el protocolo Z se encarga específicamente de la recuperación de los documentos, debe existir otro proceso que permita hacer un previo descubrimiento de los mismos para determinar cuáles documentos obtener. Este proceso es soportado por el protocolo OAI-PMH, el cual es de interés en el presente trabajo y es el encargado de la consulta y recuperación de metadatos soportando de esta manera dicho descubrimiento previo.

Así mismo, en las fuentes de información de estos proyectos no se dan detalles sobre la arquitectura sobre la cual están implementados, por lo que se pueden hacer suposiciones sobre las ya mencionadas lo que lleva a la realización de un estudio con la finalidad de determinar qué arquitectura se adopta de mejor manera a las necesidades de integración que se deseen.

Como ya se mencionó, estos proyectos utilizan protocolos de recuperación de documentos como el Z39.50 o SRU. Además de que algunos de ellos los utilizan en asociación con los estándares de metadatos DC y LOM, permitiendo que los repositorios que implementen dichos protocolos puedan interoperar con la finalidad de proveer acceso a la información que reside en ellos; sin embargo, las iniciativas mencionadas al utilizar repositorios como Fedora, recuperan información que existe en cualquier idioma, lo que puede ser una desventaja para países de habla hispana como el nuestro. Además de que no determinan si los resultados recuperados por estos proyectos son significativos para el usuario, lo que

hace notar la falta de un mecanismo capaz de medir la relevancia de los mismos en base a la búsqueda inicial realizada por el usuario final.

I.2.2. Características de la búsqueda federada

Las iniciativas mencionadas concuerdan en la integración de repositorios heterogéneos y distribuidos, así como el proveer un mecanismo de acceso unificado a los mismos. Estas acciones son características de la búsqueda federada lo que puede suponerse como las principales características de la misma; sin embargo, Pesch (2006) considera que existen tres fases que deben ser incluidas en la búsqueda federada:

1. Se debe adquirir información sobre los contenidos de cada recurso (descripción del recurso).
2. Se debe seleccionar un conjunto de recursos para búsqueda (selección de recurso).
3. Después de que los resultados han sido regresados de los recursos seleccionados, los mismos deben ser combinados en una simple lista (recolección de recursos).

A su vez, Si (2006) menciona que el mecanismo de búsqueda federada puede ser comprendido en cinco componentes:

1. Descubrimiento de recursos: identificar fuentes de información que contengan información escondida.
2. Iniciación de interfaz: provee una API (del inglés Application Programming Interface - Interfaz de Programación de Aplicaciones) estándar de peticiones de interfaces de búsqueda y extrae los datos de las páginas resultantes de cada recurso escondido disponible.
3. Representación de recursos: hay diferentes formas de representar los recursos como por ejemplo descripciones del contenido de fuentes de información escondida por las palabras y sus ocurrencias, tamaño estimado de la fuente de información,

perfiles de efectividad de la recuperación de motores de búsqueda, tiempo del respuesta del motor de búsqueda, y así existen infinidad de formas de representación. Para diferentes motores de búsqueda es muy importante la manera en la cual es representada la información.

4. Selección de recursos: existen algoritmos que tomando como referencia una cadena de texto, eligen un pequeño conjunto de recursos de información que sean más apropiados a la cadena dada por el usuario.
5. Unión de resultados: cuando se seleccionan las fuentes de información, se pueden crear varias listas clasificadas, pero no es conveniente mostrar estas listas al usuario sino que se deben unir en una sola lista para poder mostrar esta última al usuario final.

De las propuestas anteriores podemos notar que los componentes mencionados por Si (2006) son una descripción a mayor detalle de las fases consideradas por Pesch (2006). Los tres primeros componentes pueden ser englobados en la fase de descubrimiento de recursos propuesta por Pesch (2006), mientras que los dos componentes restantes son muy similares a las últimas dos fases. Basado en lo anterior, la definición de la búsqueda federada adoptada en este trabajo, puede ser descrita mediante el conjunto de fases propuesto por Pesch (2006).

I.2.3. Colecciones y bibliotecas digitales y repositorios de objetos de aprendizaje

Al concentrar recursos digitales de información en un sitio Web, se van formando colecciones con la intención de hacerlas disponibles para quienes se interesen por su consulta. Sin embargo, en muchos casos no es sencillo recuperar los contenidos de dichas colecciones, en algunas ocasiones porque no existe un orden, en otras porque la organización hecha no es intuitiva, incluso se llegan a encontrar colecciones en las que no hay registro de lo que contienen y deben hacerse inspecciones exhaustivas para encontrar

algún contenido útil. También es común encontrar largos listados de contenidos que no permiten búsquedas, en los que el usuario debe inspeccionar cada recurso para determinar los que le son útiles y los que no.

Las colecciones de recursos digitales son catalogadas dependiendo de la manera en la que son organizados dichos recursos, recibiendo distintos nombres y difiriendo en las funciones que provee el repositorio.

Las bibliotecas digitales son un conjunto de recursos electrónicos con capacidades técnicas asociadas para creación, búsqueda y uso de información. En este sentido las bibliotecas son una extensión y mejora de almacenamiento de información y sistemas de recuperación que manipulan datos digitales en cualquier formato (texto, imagen o sonido), además existen en redes distribuidas. El contenido de las bibliotecas digitales incluye datos, metadatos que describen varios aspectos de los datos (representación, creador, propietario), y metadatos que consisten de ligas o relaciones a otros datos o metadatos, ya sean internos o externos a la biblioteca digital (Borgman, 1999).

Las bibliotecas digitales basan el contenido de sus repositorios en objetos de información, que se refieren a todo tipo de objeto que provea información, como imágenes, videos, animaciones y multimedios. Además hacen uso de las telecomunicaciones y particularmente de Internet para facilitar el acceso a sus contenidos (Leiner, 1998). Para que una biblioteca digital pueda gestionar los recursos digitales es necesario que éstos se encuentren bien organizados y claramente identificados, para esto hacen uso de los metadatos los cuales fungen como descriptores del recurso al cual están asociados.

Los objetos de información antes mencionados deben contar con la característica de la reutilización con el propósito de facilitar su uso por otros sistemas que los requieran, además que la información contenida en dichos objetos debe ser de carácter pedagógico para de esta manera apoyar a la enseñanza y el aprendizaje de los usuarios del objeto. Estos puntos, hacen que dichos objetos sean considerados como objetos de aprendizaje (OA).

Los OA son elementos para la instrucción, aprendizaje o enseñanza basada en computadora descritos por metadatos. La reutilización es una bondad que deben poseer estos objetos por lo que es necesario un lugar destinado para su almacenamiento y clasificación con la finalidad de facilitar posteriormente su mantenimiento, localización y el posible acceso de otros sistemas a los mismos; este lugar es conocido como repositorio de objetos de aprendizaje.

Los metadatos son un conjunto de atributos o elementos necesarios para describir un recurso. Por medio de éstos, se tiene un primer acercamiento con el objeto de aprendizaje, conociendo sus principales características. El metadato es escrito en lenguaje XML (acrónimo para Extensible Markup Language) y la utilización de los mismos facilita la indexación de los objetos de aprendizaje (López, 2005).

Los archivos de las universidades y los museos han hecho movimientos significativos para adoptar y utilizar los esquemas de metadatos basados en XML para una descripción bibliográfica (Reese, 2005).

Arencibia (2006) considera a los almacenes de datos como repositorios de información los cuales tienen como objetivos principales: el garantizar que sean conocidos los autores de los componentes contenidos en el repositorio, facilitar el contacto entre ellos, favorecer la discusión de los trabajos contenidos en dicho repositorio y contribuir al aumento de las citas hechas sobre los autores

La variedad de contenidos de información existentes da origen a que existan varios tipos de repositorios, diferentes entre ellos, pero con la finalidad de compartir los componentes que cada uno contiene. Este proceso es conocido como interoperabilidad, la cual es soportada por protocolos e iniciativas que dan bases para la comunicación entre repositorios

heterogéneos y distribuidos. Algunas iniciativas son: la Open Archives Initiative (OAI[1]) y la Open Knowledge Initiative (OKI[2]).

Cada sistema conocido como repositorio puede pertenecer a un conjunto de los mismos, todos ellos con información similar, y de esta manera formar Federaciones de Redes Computacionales (CNF, Computational Network Federations, por sus siglas en inglés).

Las federaciones de redes computacionales proveen una capa de abstracción que unifica diferentes computadoras arbitrarias conectadas en un solo recurso ubicuo, dinámico y homogéneo, mediante la virtualización de recursos computacionales y de comunicación desde la perspectiva de servicios, desarrolladores o usuarios finales.

Una CNF habilita un conjunto arbitrario de sistemas heterogéneos, los cuales están conectados por cualquier tipo de red para formar un sistema distribuido virtual dinámico que coopera para ejecutar una aplicación o funcionar como una plataforma de servicios de aplicación generalizada para los usuarios finales (Breg y Polychronopoulos, 2005).

Las iniciativas de búsqueda federada mencionadas, las características que presenta la búsqueda federada, así como los repositorios de objetos de aprendizaje permiten darnos cuenta del amplio campo de estudio que abarca este proceso, ya que considera la integración de sistemas de información heterogéneos y distribuidos con la finalidad de interoperar y así compartir y permitir el acceso a la información contenida en los mismos; sin embargo, este proceso de integración no es tan sencillo dado que existen dificultades que deben ser soportadas tales como: los permisos de acceso del usuario, el acceso a los repositorios y el acceso y utilización de los objetos contenidos en dichos repositorios. Además de estas limitantes, se encuentran los procesos internos de la búsqueda federada como son: la selección de los repositorios de búsqueda, la recuperación de los objetos

[1] http://www.openarchives.org/

[2] http://www.okiproject.org/

contenidos en los mismos y la correcta presentación de los resultados al usuario final; por lo que la búsqueda federada se complica aún más.

I.3. Descripción del problema

Al utilizar alguna herramienta de búsqueda federada de objetos de aprendizaje, como la de ARIADNE, la búsqueda se dispara a través de palabras clave al igual que cualquier buscador genérico. El resultado es una lista con los títulos encontrados en los cuales se puede tener acceso al campo de descripción del contenido educativo. Cada elemento es una liga que lleva al recurso educativo o bien a los metadatos del objeto, dependiendo de la información disponible en cada repositorio, y de la disponibilidad de los recursos, ya que algunos no pueden estar disponibles en línea. Una limitante que es muy notable en el caso de ARIADNE, es que el sistema regresa recursos digitales en cualquier idioma, lo cual representa un problema para su empleo en países de habla hispana, como el nuestro.

Resulta claro que en un buscador genérico, se centraliza la información de cada recurso y se generan los metadatos examinándolo de manera manual o automática. Para realizar la indexación de estos metadatos existe mucha información sobre métodos para asignar pesos y categorizar cada recurso, con el objeto de que los resultados presentados correspondan al interés del usuario dependiendo de la información de búsqueda que proporcione. Sin embargo, en la búsqueda federada de repositorios, al estar los índices distribuidos, no es evidente que se pueda proporcionar la misma funcionalidad que en los buscadores genéricos. Al considerar repositorios heterogéneos y distribuidos se permite suponer que la información contenida en cada uno de ellos puede estar presente en formatos incompatibles entre los mismos, es decir, cada repositorio puede contar con un formato de información propio que no puede ser interpretado por un mecanismo de búsqueda ajeno al utilizado por dicho repositorio, algo similar puede ocurrir con las estructuras de datos e índices utilizados por cada repositorio. Además, la cantidad de información que engloben los índices de los repositorios puede ser más grande que las capacidades físicas del mismo, lo que propiciaría

que sea necesario particionar la totalidad de la información en varios repositorios originando que el contenido se encuentre disperso entre los mismos. Esta distribución de información implica realizar una búsqueda que incluya, no solo un repositorio, sino la totalidad de repositorios en los que fue almacenada la información y que permita el acceso a cada uno de ellos; sin embargo, no todo el contenido distribuido en estos repositorios puede ser relevante para cierta búsqueda, por lo que es necesario delimitar hacia cuáles repositorios direccionar dicha búsqueda, lo que se traduce en una respuesta significativa para el usuario que realizó la consulta inicial.

Debido a lo anterior, para los objetivos de algunas de las líneas de investigación que se siguen en el grupo EDUMAT-TI (2000), se considera necesario realizar un análisis de los mecanismos de búsqueda de recursos digitales en repositorios distribuidos; esto, con el fin de establecer propuestas que conduzcan al desarrollo de una búsqueda más precisa en cuanto a la calidad de los resultados obtenidos, es decir, realizar una búsqueda que recupere resultados significativos para el usuario de una manera semejante a la que se realiza en buscadores genéricos con bases de datos centralizadas. Entendiendo por mecanismos a todo aquello que comprenda interfaces, procesos y algoritmos que habiliten la búsqueda federada de recursos digitales educativos. Todo esto, enfocado a bases de datos (repositorios de objetos de aprendizaje) en español, y de preferencia orientado a aquellas que se emplean en el país; dado que la gran cantidad de repositorios existentes contienen objetos de aprendizaje redactados en un idioma diferente, lo que resulta en una recuperación de dichos objetos orientada a usuarios familiarizados con el idioma del objeto en cuestión.

I.4. Objetivos

Los objetivos del presente trabajo de tesis son los presentados a continuación.

I.4.1. Objetivo General

Proponer una arquitectura para la implementación de un mecanismo para realizar búsquedas federadas de manera óptima en repositorios distribuidos de objetos de aprendizaje respecto a un criterio de relevancia en base a la cadena de búsqueda.

I.4.2. Objetivos Específicos

- Realizar un análisis de las interfaces, procesos y algoritmos que puedan ser empleados en búsquedas federadas de objetos de aprendizaje en repositorios distribuidos de habla hispana.
- Definir un criterio de optimización de recuperación de objetos de aprendizaje, basado en el estado del análisis de las interfaces, procesos y algoritmos.
- Evaluar las interfaces, procesos y algoritmos susceptibles de ser implementados para la búsqueda federada en repositorios distribuidos de habla hispana.
- Definir el mecanismo a utilizar para la implementación de la arquitectura.
- Realizar la implementación de la arquitectura.

I.5. Metodología de la investigación

La metodología llevada a cabo en este trabajo, es la que se cita a continuación. Dichas etapas son descritas de manera general y son la pauta a seguir para el cumplimiento de los objetivos planteados en el presente documento.

- Revisar la literatura. En esta etapa de la investigación, se analiza toda la literatura referente a la búsqueda federada con el fin de obtener bases reales para el cumplimiento de los objetivos de la tesis. También se busca conocer el tema a mayor profundidad para obtener conocimientos necesarios para el desarrollo de las etapas subsecuentes.

- Definir la federación y sus características. Esta etapa es una continuación de la revisión de literatura, pero enfocándose específicamente al concepto de federación, con el propósito de determinar características, componentes, funcionamiento y todo lo que englobe el término.
- Elementos que conforman la federación y tipos de federación. Una vez que se ha determinado la definición que satisfaga las necesidades en el ámbito de federación para el presente trabajo, se procederá a definir todos y cada uno de los componentes de la misma, así como los tipos y las propuestas de federación existentes para continuar delimitando el tema así como con las etapas de desarrollo siguientes.
- Determinar las arquitecturas de federaciones existentes. Una vez que se ha obtenido la definición adoptada de federación, sus características y componentes mínimos necesarios para su funcionamiento, se procede a encontrar arquitecturas existentes que cumplan con el funcionamiento de una federación.
- Análisis de las arquitecturas encontradas. Cuando se han determinado las arquitecturas que cumplan con el funcionamiento de una federación, se procederá al análisis de las mismas con el fin de determinar el funcionamiento, comunicación, interoperabilidad y demás puntos que conciernen a una arquitectura de federación.
- Proponer arquitectura a realizar. Después del análisis de las arquitecturas encontradas y de obtener sus características y componentes, se tendrán suficientes bases para determinar que la arquitectura cumpla con los objetivos y tareas plasmadas en la investigación a desarrollar.
- Implementación de la arquitectura propuesta. Una vez que se ha determinado la arquitectura a utilizar, se procederá a definir la manera en la cual se puede implementar dicha arquitectura, tomando en cuenta los componentes e infraestructura con los que se cuenta.
- Evaluar arquitectura propuesta. Después de haber implementado la arquitectura determinada y que la misma sea operacional, se procederá a evaluarla mediante pruebas que involucren una recuperación idónea de metadatos basándose en una cadena de búsqueda enviada por el usuario.

- Interpretar resultados y concluir. Tomando en cuenta los resultados obtenidos de las pruebas realizadas, se procede con un análisis de los mismos, con el fin de determinar el funcionamiento de la arquitectura implementada para posteriormente dar conclusiones sobre el trabajo de investigación realizado y proponer actividades futuras a desarrollar.

I.6. Contenido de la tesis

En el Capítulo II de la presente tesis se puede encontrar información referente a los sistemas de bases de datos, ya que de éstos es el punto de partida por el cual iniciamos en el proceso de integración de componentes, en específico, de bases de datos. Del proceso de integración mencionado, son resultantes los sistemas de información federados, los cuales son la pauta para la generación de una federación y que son descritos en el Capítulo III. Un tipo especial de sistemas de información federados, son los sistemas de información basados en mediadores, los cuales son tratados en el Capítulo IV. En el Capítulo V, se trata información referente a los objetos de aprendizaje, los cuales son los componentes a los cuales se desea acceder para su manipulación y consulta. El diseño y descripción de la arquitectura propuesta, son presentados en el Capítulo VI, los cuales permiten la creación de la federación así como la integración de repositorios de objetos de aprendizaje. La evaluación y resultados obtenidos sobre los mecanismos implementados en la arquitectura, son presentados en el Capítulo VII. Finalmente, el Capítulo VIII, presenta las conclusiones, aportaciones y recomendaciones para trabajo futuro.

Capítulo II

Sistemas de Bases de Datos

El término federación, puede ser aplicado en distintos contextos, sin embargo en el ámbito de este trabajo el mismo es utilizado para hacer referencia a un conjunto de sistemas de cómputo heterogéneos y distribuidos con el propósito de compartir todos o algunos de los recursos que cada sistema posee. En otras palabras, una federación tiene como objetivo la cooperación entre sistemas independientes.

Cada uno de estos sistemas que conforman la federación, puede ser conocido como sistema de información, es decir, son los lugares donde se encuentran alojados los datos y la información que puede satisfacer, mediante mecanismos de acceso y recuperación, la necesidad de información de un usuario en particular.

Dado que la información puede ser datos simples almacenados con cierta estructura, los mismos pueden ser alojados en un sistema de base de datos donde cada sistema puede almacenar datos de distinta índole y con diversas características.

En este capítulo se presenta la definición de un sistema de base de datos así como la clasificación de los mismos, de la cual, parte el concepto inicial de bases de datos federadas. Esta clasificación de sistemas de bases de datos funciona como un primer acercamiento a las características y funcionalidad de una federación, para posteriormente profundizar en la arquitectura de referencia de bases de datos federadas, la cual provee un

marco de trabajo para comprender de mejor manera las partes que involucran a una federación.

II.1. Sistemas MultiBase de Datos

Un sistema de base de datos consiste de un software, llamado sistema administrador de base de datos (DBMS, por sus siglas en ingles), y de una o más bases de datos que administra. Estos sistemas pueden ser centralizados o distribuidos.

Los sistemas de bases de datos centralizados consisten de un DBMS simple y centralizado que administra una sola base de datos en el mismo sistema computacional. El sistema de bases de datos distribuido consiste de un DBMS simple y distribuido que administra múltiples bases de datos. Las bases de datos pueden residir en un sistema computacional simple o en múltiples sistemas que pueden ser diferentes en hardware, software y soporte de comunicación.

Estos sistemas de bases de datos distribuidos pueden ser conocidos como Sistemas de Información Compartida (Ince, 2000), Sistemas de MultiBase de Datos o Sistemas MultiBase de Datos Heterogéneos (Sheth y Larson, 1990).

El Sistema de Información Compartida es una serie de computadoras interconectadas por algún tipo de red de comunicación (Ince, 2000), con el fin de compartir información residente e integrada de alguna forma en cada una de esas computadoras, y permitir a los usuarios observar a los sistemas de bases de datos como si fueran un solo sistema centralizado.

Un Sistema MultiBase de Datos (MDBS, por sus siglas en ingles) soporta operaciones en múltiples Sistemas de Base de Datos (SBD). Cada SBD es manejado por un sistema manejador de base de datos (DBMS, por sus siglas en ingles). Un SBD en un MDBS puede

ser centralizado o distribuido y puede residir en la misma computadora o en múltiples computadoras conectadas por un subsistema de comunicación. Un MDBS es llamado homogéneo si todos los DBMS son iguales; si son diferentes entonces es llamado un MDBS heterogéneo.

Sheth y Larson (1990) proponen la taxonomía mostrada en la Figura 1 para comparar las arquitecturas de diversos esfuerzos de investigación y desarrollo en relación a sistemas multibase de datos heterogéneos y homogéneos, así como centralizados y distribuidos.

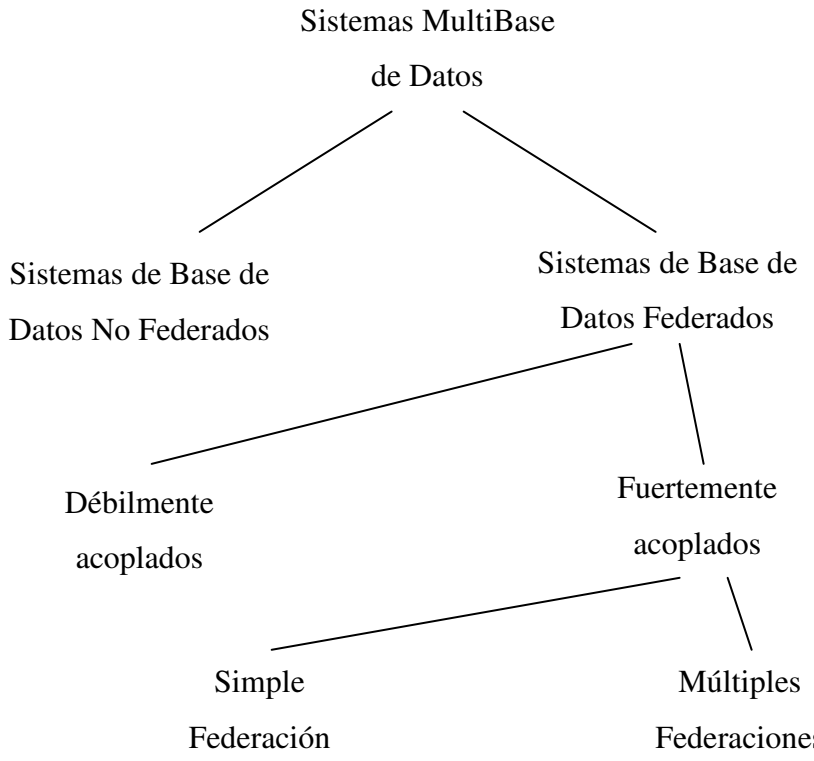

Figura 1. Taxonomía de los Sistemas MultiBase de Datos propuesta por Sheth y Larson

Un Sistema MultiBase de Datos puede ser clasificado en dos tipos basados en la autonomía de los sistemas de bases de datos, los cuales son: Sistemas de Base de Datos no Federado y Sistemas de Base de Datos Federado.

II.1.1. Sistemas de Base de Datos No Federado

Un Sistema de Base de Datos no Federado es una integración de DBMS que no son autónomos. Esto significa que los SBD al participar en una federación pierden su autonomía y cualquier operación debe hacerse sobre la base de datos global. Un sistema de este tipo no distingue entre usuarios locales y usuarios no locales. Un tipo particular de sistema de base de datos no federado en el cual todas las bases de datos están completamente integradas para proveer un esquema global simple puede ser llamado Sistema MultiBase de Datos unificado. Esto lógicamente parece a los usuarios como un Sistema de Base de Datos distribuido.

II.2. Sistema de Base de Datos Federado

Los Sistemas de Base de Datos Federados (FDBS, por sus siglas en ingles) son sistemas completamente autónomos y no dependen de un esquema global de datos para procesar las consultas distribuidas; este tipo de consultas son muy complejas dada la cantidad de bases de datos independientes que están interconectadas y que poseen reglas propias de optimización de consulta, detección de tiempo y concurrencias.

Calegari *et al.* (2005) menciona que un Sistema Federado Distribuido de Bases de Datos "es una colección de sistemas de computo (usualmente sistemas de bases de datos) independientes, cooperativos, posiblemente heterogéneos y autónomos", permitiendo compartir todos o algunos de sus datos o información.

Un FDBS consiste de SBDs que son autónomos, participan en una federación para permitir compartición parcial y controlada de sus datos. El concepto de autonomía implica que los SBDs tienen control sobre los datos que ellos manejan. Ellos cooperan para permitir diversos grados de integración. No hay control centralizado en una arquitectura federada

debido a que los SBDs (y sus administradores de bases de datos) controlan el acceso a sus datos.

El término federación existe en varios conceptos, pero enfocado a bases de datos, se entiende por federación a un conjunto de entidades (sistemas de cómputo) independientes que cooperan para lograr un objetivo en particular; cada una de estas entidades puede decidir si se comunica o no con otra o varias de las mismas. En el contexto de bases de datos no existe un modelo ideal o simple para una federación, pero existe una característica clave de la misma para que pueda ser considerada como tal: cooperación entre sistemas independientes.

Para permitir la compartición controlada de datos mientras preserva la autonomía de los SBDs y continuar con la ejecución de aplicaciones existentes, un FDBS soporta dos tipos de operaciones: local y global (o federación). Esta división de operaciones globales y locales es una característica esencial de un FDBS. Las operaciones globales involucran acceso a los datos usando un sistema manejador de base de datos federado y puede involucrar manejar datos por múltiples SBDs. Los SBDs deben dar permisos de acceso a los datos que ellos manejan. Las operaciones locales son sometidas a un SBD directamente. En la mayoría de los ambientes los FDBS son heterogéneos, y entonces los SBD también lo son.

II.2.1. Propiedades de un Sistema Federado

Un sistema federado debe cumplir con tres propiedades fundamentales para que pueda ser considerado como tal: Autonomía, Heterogeneidad y Distribución. Dichas propiedades permiten poder clasificar a los sistemas federados.

II.2.1.1. Autonomía

La autonomía de un sistema de base de datos se refiere al control independiente y separado que tiene sobre sí mismo; se pueden distinguir varios tipos de autonomía: diseño, comunicación, ejecución y asociación.

- La autonomía de diseño se refiere a que un sistema de bases de datos elija su propio diseño con respecto a algún asunto que incluye información propia, es decir, los datos, su representación, interpretación semántica, restricciones para manejar datos, funcionalidad del sistema, asociación y compartición con otros sistemas, así como la implementación. Este tipo de autonomía es la principal causa que propicia la heterogeneidad de un sistema de base de datos.
- La autonomía de comunicación se da cuando un sistema de bases de datos puede decidir de manera independiente, con qué otros sistemas se comunica; los sistemas con esta autonomía son capaces de decidir cuándo y cómo responder a una petición.
- La autonomía de ejecución es la habilidad de ejecutar operaciones locales sin interferencia de operaciones externas y decidir el orden en el cual ejecutar dichas operaciones. De manera operacional, un sistema de información ejerce su autonomía tratando las operaciones externas de la misma manera que como si fueran locales.
- La autonomía de asociación se refiere a la decisión de un sistema de bases de datos sobre compartir y qué tanta información y recursos van a ser compartidos. Esto incluye la decisión de asociarse o desasociarse de una o más federaciones.

La autonomía de asociación no debe ser soportada completamente, es decir, si un sistema de bases de datos tiene la libertad de unirse y desunirse de la federación cuando quiera, la federación debe ser creada de una manera tal que su existencia no dependa de ningún sistema de bases de datos. Dicha unión y desunión debe ser hecha de común acuerdo por las dos partes y no tomada unilateralmente por el sistema de información. En cuanto a la autonomía de ejecución, ésta no puede ser tomada completamente puesto que seria

conveniente que un sistema de bases de datos notifique sobre el estado de las tareas que él esta ejecutando a la federación, para permitir una administración más simple y eficiente de transacciones globales.

II.2.1.2. Heterogeneidad

La literatura tiene muchas clasificaciones de heterogeneidad en diferentes niveles de detalle (Sheth y Larson, 1990; Busse *et al.*, 1999; Wiederhold, 1993) pero de acuerdo con Sheth y Larson (1990), la heterogeneidad de los sistemas de bases de datos es muy variada originando que los mismos puedan ser diferentes por: la plataforma en que están funcionando, su estructura, la manera en que los datos son almacenados, los lenguajes de consulta que cada sistema utiliza, las restricciones de acceso y, la más importante o más difícil de descifrar, la diferencia semántica. Esta diferencia implica que aunque el nombre del atributo sea el mismo en diferentes sistemas de información, puede tener un significado completamente diferente en cada uno de ellos.

Para comprender mejor la diferencia semántica tomemos el siguiente ejemplo en el cual consideramos un atributo llamado COSTO_COMIDA de la relación RESTAURANTE en una base de datos uno (BD1), que describe el costo promedio de una comida por persona sin el cargo de la propina; consideremos una segunda base de datos (BD2) que contiene el mismo nombre de atributo pero describe el costo promedio de una comida incluyendo la propina. Aunque ambos atributos tienen las mismas propiedades sintácticas, sus definiciones son diferentes originando la heterogeneidad.

II.2.1.3. Distribución

Además de la autonomía y heterogeneidad, existe el problema de la distribución de las fuentes de datos. Debido a que en estos días la mayoría de las computadoras están

conectadas mediante algún tipo de red, especialmente Internet, es natural el pensar en la gran combinación de aplicaciones y fuentes de datos ubicadas en diferentes sitios, pero capaces de comunicarse mediante la red.

La distribución de estas fuentes de información, la cual no sólo es física sino también lógica, genera la necesidad de tener mecanismos flexibles de integración y comunicación remota para la interconexión de las propias fuentes (Calegari *et al.*, 2005).

Un FDBS puede ser categorizado como débilmente acoplado o fuertemente acoplado basado en la idea de quién maneja la federación y cómo es que los componentes son integrados.

II.2.2. Sistemas de Bases de Datos Federados Débilmente Acoplados

Un FDBS es débilmente acoplado si la responsabilidad de crear y mantener la federación recae en el usuario y no hay control por parte del sistema federado y sus administradores. Litwin *et al.* (1990) se refiere a este mismo concepto como multibases de datos o bases de datos interoperables. Ellos asumen que los usuarios necesitan acceder a múltiples datos sin el beneficio de un esquema global y que el componente esencial de un sistema de este tipo es el lenguaje usado para manejar las bases de datos participantes. Otro requerimiento importante es que el usuario debe ser capaz de formular manipulaciones multibase de datos no procedimental en la ausencia de un esquema global. El usuario es responsable de comprender la semántica de los objetos en los esquemas de exportación y resolver la heterogeneidad de los DBMS y de la semántica.

El lenguaje multibase de datos debe permitir a los usuarios definir y manipular una colección de bases de datos autónomas en una forma no procedimental, es decir, sin un solo procedimiento genérico para acceder a los mismos. Tal lenguaje necesita características que no son parte de lenguajes de bases de datos, esto debido a que los DBMS clásicos fueron

desarrollados para una sola base de datos. El objetivo del lenguaje multibase de datos es crear mecanismos que puedan simultáneamente ejecutar consultas que involucren a varios SBDs.

II.2.3. Sistemas de Bases de Datos Federados Fuertemente Acoplados

Una Federación es fuertemente acoplada si su administrador(es) tiene la responsabilidad de crear y mantener la federación y el control de acceso a los SBDs. Una federación esta compuesta por una integración selectiva y controlada de sus componentes. La actividad de desarrollar un FDBS fuertemente acoplado consiste en la creación de un esquema federado sobre el cual las operaciones (consultas y/o actualizaciones) son ejecutadas.

Un FDBS fuertemente acoplado puede tener uno o más esquemas federados. Un FDBS fuertemente acoplado se dice que tiene una federación sencilla si permite la creación y manejo de solamente un esquema federado. Tener un esquema federado sencillo ayuda a mantener la uniformidad en la interpretación semántica de los datos integrados. Un FDBS fuertemente acoplado se dice que tiene una federación múltiple si permite la creación y manejo de múltiples federaciones. Las restricciones involucran a múltiples SBDs sin embargo, pueden ser difíciles de imponer.

Un FDBS fuertemente acoplado provee localización, duplicación y transparencia de distribución. Esto es llevado a cabo al desarrollar un esquema federado que integra múltiples esquemas de exportación. Las transparencias son manejadas por los mapeos entre el esquema federado y los esquemas de exportación, y un usuario de la federación puede hacer consultas, a través de un lenguaje de consultas clásico al esquema federado con la ilusión de que se esta accediendo a un solo sistema (Sheth y Larson, 1990).

Debido a que un esquema federado es creado al integrar todos los esquemas de exportación y soporta además los requerimientos de datos de todos los usuarios, puede llegar a ser

demasiado grande y, por tanto, difícil de crear y mantener. Estas federaciones son cómodas para los usuarios de la federación, ya que no necesitan conocer los esquemas de todos los SBDs, sino solo el esquema federado.

Una arquitectura de referencia, como la descrita a continuación, es necesaria para clarificar varios puntos y elecciones entre los SBDs.

II.3. Arquitectura de Referencia de Bases de Datos Federadas

Una arquitectura de referencia provee un marco de trabajo (Framework) para poder entender, categorizar y comparar diferentes opciones arquitecturales para desarrollar sistemas de bases de datos federadas.

Los componentes básicos de la arquitectura de referencia son:

- Datos: los datos básicos y la información administrada por un sistema de base de datos.
- Base de datos: es un repositorio de datos estructurados de acuerdo a un modelo de datos.
- Mandos: peticiones para acciones especificas que son generadas por un usuario o procesador.
- Procesadores: módulos de software que manipulan mandos y datos.
- Esquemas: descripciones de datos administrados por uno o mas DBMS. Son objetos de esquema y sus interrelaciones. Los objetos de esquema son típicamente definiciones de clases, tipos de entidades y tipos de relaciones en el modelo entidad-relación.
- Mapeos (Mappings): funciones que correlacionan los esquemas de objetos en un esquema, a un esquema de objetos en otro esquema.

Una característica para elegir estos componentes es que esconden detalles de implementación que no son relevantes para entender las diferencias entre las arquitecturas (Figura 2). Además de que la mayoría de los sistemas federados de bases de datos, centralizados y distribuidos pueden ser expresados usando estos componentes básicos.

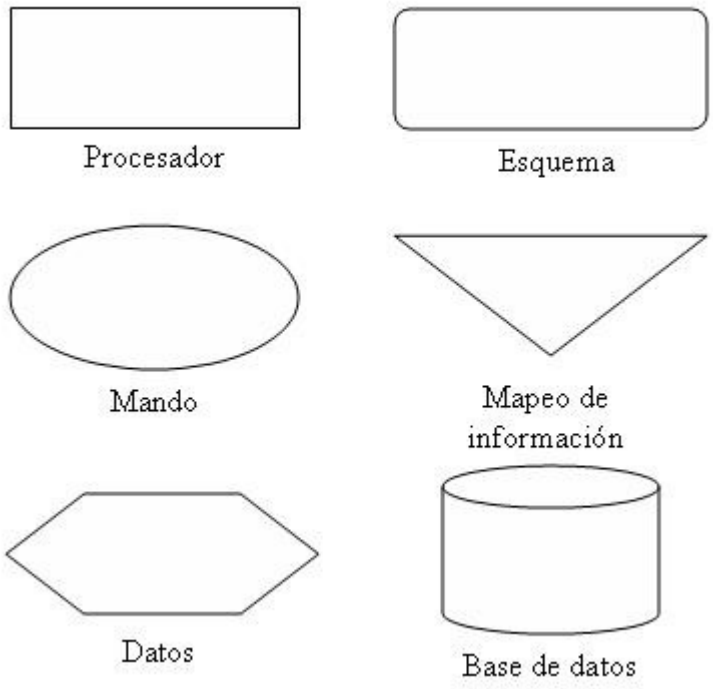

Figura 2. Componentes básicos del sistema de la arquitectura de referencia.

Dos componentes básicos, procesadores y esquemas, juegan roles importantes en la definición de varias arquitecturas. Los procesadores son módulos de software de aplicación de un DBMS. Los esquemas son componentes de aplicación específicos que definen contenidos y estructura de bases de datos.

II.3.1. Tipos de procesadores

Recordemos que los procesadores son módulos de software de aplicación independiente de un DBS, por lo que la clasificación de éstos es (Roantree *et al.*, 2001):

- Procesadores de transformación: traducen los mandos de un lenguaje a otro lenguaje, o traducen los datos de un formato a otro formato. Proveen un tipo de independencia de datos llamado modelo de transparencia de datos en el cual la estructura de datos y mandos usados por un procesador son escondidos a otro procesador. Esconden diferencias de lenguajes de consulta y formato de datos.
- Procesador de filtro: restringe los mandos y datos asociados que pueden ser pasados a otro procesador (para cada procesador existe un mapeo que describe las restricciones en los mandos y los datos). Estas restricciones pueden estar incrustadas en el código del procesador o ser especificadas en una estructura de datos dada.
- Procesador de construcción: particiona y/o replica una operación enviada por un procesador, en operaciones que son aceptadas por dos o más procesadores distintos. También une los datos producidos por varios procesadores en un conjunto simple de datos para que otro procesador los utilice.
- Procesador de acceso: acepta mandos y produce datos ejecutando aquellos contra la base de datos. Puede aceptar mandos de varios procesadores e intercalar el procesamiento de esos mandos.

II.3.2. Esquemas

Los procesadores antes mencionados, son utilizados para poder otorgar la funcionalidad que caracteriza a la federación, pero estos procesadores necesitan información extra, la cual es producida mediante los llamados esquemas. Como se mencionó anteriormente, los esquemas son descripciones de datos administrados por uno o más DBMS; consisten de objetos de esquema y sus interrelaciones.

Los esquemas son utilizados como objetos que describen información en diferentes niveles de la federación, donde cada procesador los utiliza para poder ejecutar las operaciones que requiera el usuario de la federación. La creación y el manejo de los esquemas, sirven como base para crear la arquitectura de referencia para bases de datos federadas, ya sea

centralizadas o distribuidas. Los esquemas son de distinto tipo, dependiendo de la arquitectura en la que se encuentren así como del nivel que ocupen la misma y de la información que contienen, lo que determina cuáles son los procesos y componentes que pueden acceder a dicha información.

II.4. Arquitectura de tres niveles

La arquitectura de referencia es una extensión de la propuesta por ANSI/X3/SPARC Study Group on Database Systems. Primero maneja el esquema de tres niveles estándar para bases de datos centralizadas, posteriormente se hace una extensión de éste para los requerimientos de distribución, autonomía y heterogeneidad de una FDBS. Estos tres niveles son:

- Esquema conceptual: consiste de objetos que proveen una descripción a nivel lógico o conceptual de la base de datos (estructuras), y las relaciones entre esas estructuras. Describe todos los datos que sean de interés.
- Esquema interno: describe características físicas de estructuras de datos lógicas en el esquema conceptual. Estas características incluyen información sobre la ubicación de los registros en dispositivos de almacenamiento físico, ubicación y tipos de índices y representación física de relaciones entre registros lógicos.
- Esquema externo: la mayoría de los usuarios no requieren acceso a todos los datos en la base de datos por lo que el acceso a todos los esquemas de objetos en el esquema conceptual puede ser restringido (cada usuario o clase de usuarios puede requerir acceso a solo una porción de la base de datos). Este subconjunto de base de datos que puede ser accedida por un usuario o clase de usuarios es un esquema externo. Como cada usuario o clase de usuario puede requerir acceso a diferentes porciones de la base de datos, cada usuario o clase de usuario necesitará un diferente esquema externo.

La manera en la que la arquitectura de tres niveles funciona es la siguiente: el procesador de filtro usa la información en el esquema externo para controlar qué datos pueden ser accedidos por determinados usuarios. El procesador de transformación traduce mandos expresados usando los objetos del esquema conceptual, en mandos usando objetos del esquema interno. Un procesador de acceso ejecuta los mandos para recuperar datos de un medio físico (base de datos) (Figura 3).

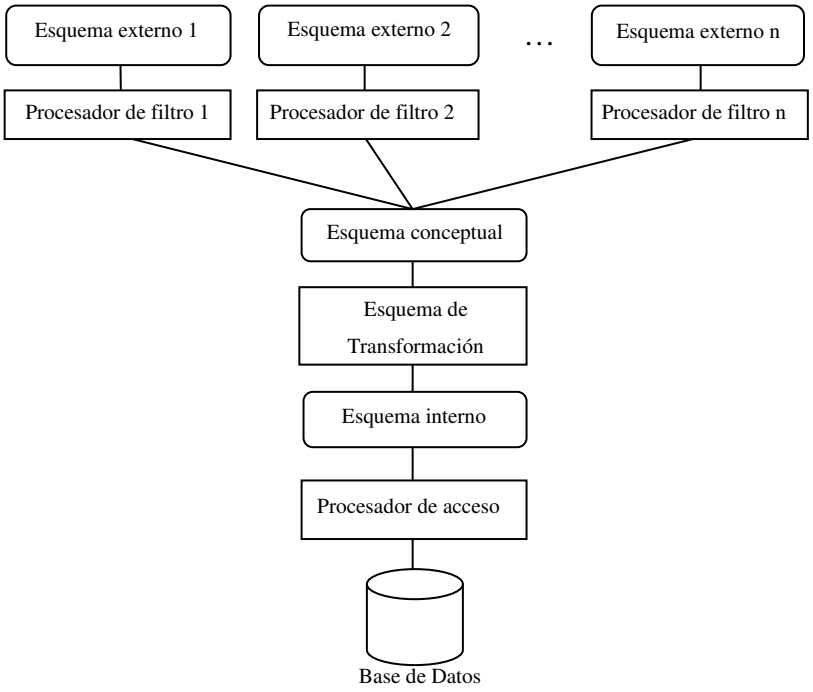

Figura 3. Arquitectura de un DBMS centralizado (de tres niveles).

II.5. Arquitectura de cinco niveles

La arquitectura anterior es adecuada para describir un DBMS centralizado, sin embargo es inadecuada para describir a una FDBS. La arquitectura de tres niveles debe ser extendida, como se mencionó anteriormente, de manera que se puedan soportar las tres características de una federación: distribución, heterogeneidad y autonomía; de esta manera se crea la arquitectura de cinco niveles (Figura 4) que incluye lo siguiente:

- Esquema local: es el esquema conceptual de un sistema de base de datos. Es expresado en el modelo de datos nativo de un DBMS.
- Esquema de componente: es derivado de traducir esquemas locales en un modelo de datos llamado canónico o modelo común de datos (CDM, por sus siglas en inglés). Hay dos razones para definir esquemas de componente en un CDM, una de ellas es porque describen los diferentes esquemas locales usando una simple representación y, la otra, es porque la semántica que está perdida en un esquema local puede ser adherida a su esquema de componente. La transformación de un esquema local a un esquema de componente genera los mapeos entre los objetos de esquema de componente y los objetos de esquema local. Los procesadores de transformación usan estos mapeos para transformar mandos en un esquema de componente, a mandos en su correspondiente esquema local, y de esta manera correlacionar esquemas de objetos en un nivel con esquemas de objetos en otro nivel de la arquitectura, soportando de esta manera la heterogeneidad.
- Esquema de exportación: representa el subconjunto de esquema de componente que está disponible a la federación. El propósito de este esquema es facilitar el control y administración de la autonomía de asociación. Un procesador de filtro puede ser usado para proveer el control de acceso como se especifica en el esquema de exportación, limitando las operaciones disponibles que pueden ser enviadas al componente de esquema correspondiente. El procesador de filtro y el esquema de exportación, soportan la autonomía.
- Esquema federado: es una integración de múltiples esquemas de exportación. Incluye información sobre distribución de datos que es generada cuando se integran los esquemas de exportación. El procesador de construcción transforma los mandos del esquema federado en mandos de uno o más esquemas de exportación. Estos procesadores y este esquema soportan la distribución.
- Esquema externo: define un esquema para un usuario y/o aplicación o para una clase de usuarios/aplicaciones. Las razones para usar este esquema son:
    - Personalización: un esquema federado puede ser muy grande, complejo y difícil de cambiar. El esquema externo puede ser usado para especificar un

subconjunto de información del esquema federado, que es relevante a los usuarios del esquema externo.

- Restricciones adicionales de integridad: pueden ser especificadas en el esquema externo.
- Control de acceso: los esquemas de exportación proveen control de acceso con respecto a los datos manejados por una base de datos; de manera similar el esquema externo provee control de acceso con respecto a los datos manejados por un FDBS.

Los usuarios finales de la federación necesitan recuperar datos de múltiples sistemas de bases de datos, lo que implicaría que los datos deben ser presentados en un modelo de formato común. Este formato unificado es llamado canónico o modelo común de datos (CDM, por sus siglas en ingles) el cual tiene un lenguaje asociado llamado lenguaje de mandos interno. El CDM permite dar soporte al esquema local y al esquema externo, los cuales están expresados en diferentes modelos de datos.

Todos los mandos en los esquemas de componente, federados y de exportación, usan este lenguaje de mandos interno para la creación de las reglas o modelos semánticos a seguir para la creación del CDM; ello es una idea subjetiva puesto que estas reglas semánticas deben ser realizadas por el desarrollador durante el diseño, integración y traducción, es decir, depende del diseñador el poder expresar el lenguaje que puede ser basado en objetos o en datos semánticos.

El funcionamiento de la arquitectura de cinco niveles sería expresado de la siguiente forma: cada base de datos tiene un esquema local en su propio lenguaje, así como cada base de datos tiene su propio lenguaje de consulta. Este esquema es transformado por un procesador de transformación en un esquema con una representación canónica, o CDM, para toda la federación (también transforma las cadenas de consulta en el lenguaje canónico de la federación a el lenguaje de la base de datos), generando un esquema de componente. Cada fuente define la parte de información a la que la federación tendrá acceso. Basado en este

control de acceso, un procesador de filtro genera un esquema de exportación. De estos esquemas de exportación de cada fuente, la federación, valiéndose de un procesador de construcción, genera el esquema federado. Después de otro control de acceso, llevado a cabo por otro procesador de filtro, este esquema es transformado en un esquema externo, al cual es el que acceden los usuarios de la federación.

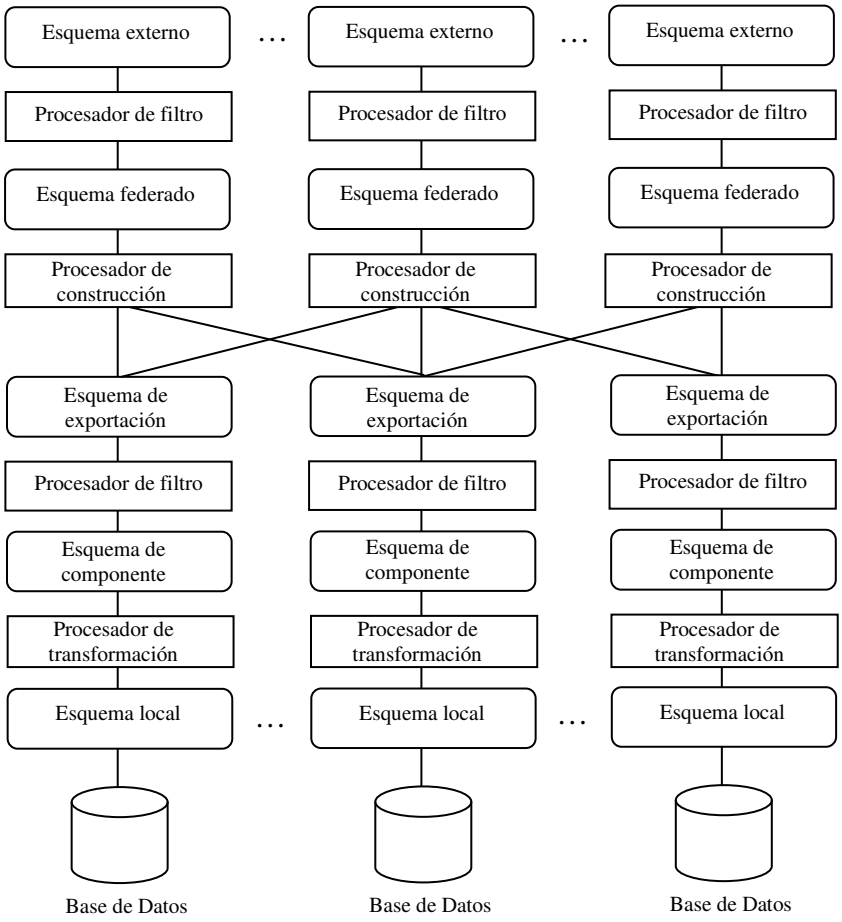

Figura 4. Arquitectura de referencia para un Sistema de Base de Datos Federado (de cinco niveles).

II.6. Resumen

En el presente capitulo, se presentó información referente a los Sistemas MultiBases de datos, los cuales son un tipo de sistemas de bases de datos con ciertas propiedades que los caracterizan. Se profundizó en una parte de dichos sistemas, los cuales son los Sistemas

Federados de Bases de Datos (FDBS) los cuales, a su vez, cuentan con ciertas características que los hacen únicos en el entorno de Sistemas MultiBase de Datos. Se menciona la clasificación que posee un FDBS y características de la misma.

Se definió el término Federación, enfocándose a bases de datos, así como también se presentó una arquitectura de FDBS, la cual sirve como referencia para mostrar el funcionamiento y posible creación de un sistema federado. Dicha arquitectura es resultado de una extensión realizada previamente para bases de datos centralizadas y puede variar en los niveles que presenta, dependiendo de las necesidades que deba satisfacer.

Capítulo III

Sistemas de Información Federados

En el capítulo anterior se hizo mención sobre los sistemas de bases de datos, los cuales, como su nombre lo indica, se enfocan específicamente en bases de datos. Sin embargo, la necesidad de información del usuario no puede ser satisfecha con un simple dato, la mayoría de las veces es necesario un conjunto de los mismos.

Este conjunto de datos puede ser representado como un documento de texto plano, un documento con formato enriquecido, una imagen, un archivo multimedia, etc., originando que no todos los sistemas capaces de satisfacer la necesidad de información del usuario sean específicos de bases de datos, es decir, son necesarios sistemas que almacenen imágenes, archivos multimedia, documentos de texto, etc.

Debido a lo anterior, es necesario llamar a los sistemas que conforman la federación, con un nombre que describa de mejor manera su funcionamiento, de este punto se obtiene el término sistemas de información.

En este capítulo se menciona la definición de un sistema de información, su clasificación así como las características de los mismos, permitiendo un primer acercamiento a la arquitectura adoptada en el presente trabajo mencionando conceptos necesarios para su mejor comprensión.

III.1. Sistemas de información

Los datos que residen propiamente en cada Sistema de Base de Datos que conforman una federación, es información de utilidad para cada usuario que acceda a la misma, sin embargo, la característica de heterogeneidad de la federación indica que no todos los sistemas que forman parte de la misma son bases de datos, es decir, cada entidad que esta en la federación es diferente y por ende con organización, almacenamiento y estructura de datos diferentes; es por esto que se adopta el término Sistema de Información para referirnos a cualquier sistema capaz de proveer información de distinta índole no enfocándose a una sola estructura de datos en particular.

Busse *et. al.* (1999) mencionan que un sistema de información es aquella entidad que permite, de alguna manera, el acceso a los datos que están almacenados en algún lugar del sistema. Así mismo menciona que si los sistemas de información que están intercomunicados, son autónomos, entonces éstos son llamados Sistemas de Información Federados.

III.2. Clasificación de los sistemas de información.

Basándose en las propiedades de distribución y heterogeneidad de las federaciones, Busse *et al.* (1999) clasifican a los sistemas de información en:

- Un sistema simple de información (también llamado monolítico o centralizado) el cual funciona en una aplicación monolítica en una computadora. Ofrece una o más interfaces para su contenido. Estos sistemas de información pueden ser:
    - Sistemas de bases de datos los cuales usan un DBMS para almacenar y administrar sus datos. En particular está basado en un modelo de datos; los datos están estructurados de acuerdo a un esquema y son accesibles a través de un lenguaje de consulta.
    - Sistemas sin bases de datos tales como sistemas de archivos, colecciones de documentos, colecciones de archivos planos, etc. Estos sistemas usualmente

no están basados en un modelo estándar de datos y usualmente no ofrecen un lenguaje de consulta.

- En un sistema distribuido de información, los datos están físicamente distribuidos sobre múltiples sitios los cuales están conectados mediante algún tipo de red de comunicación.
- Un sistema de información heterogéneo es una colección de sistemas de información que difiere en aspectos sintácticos o lógicos como plataforma de hardware, modelo de datos o semántica.

Así mismo mencionan que de este último punto se desprende el término de Sistema de Información Federado, puesto que entra en la clasificación de sistema heterogéneo.

III.3. Sistemas de Información Federados

De acuerdo con Busse *et al.* (1999), un Sistema de Información Federado (FIS, por sus siglas en ingles) está formado por sistemas de información distintos y autónomos, los cuales son los participantes de la federación. Un FIS es típicamente construido de un conjunto de componentes heterogéneos, autónomos y distribuidos; sin embargo, si los componentes son altamente homogéneos aún hablaremos de un FIS. En este sentido, aun no esta claro el grado de heterogeneidad que deben tener los componentes para cambiar de un Sistema de Información Distribuido a un FIS; sin embargo, podemos decir que el primero debe obedecer a un componente global es decir, cada sistema de información que lo conforma debe obedecer a instrucciones generadas por un sistema global; mientras que en un FIS, cada sistema de información decide si obedece o no a ese tipo de peticiones además de que deben someterse a las reglas de la federación para su integración en la misma.

Los sistemas de información son utilizados como una solución para los problemas de negocio, ciencia y administración, aunque también presentan una característica como requerimiento para el desarrollo futuro: capacidad de integración de datos. Esta integración

debe ser eficiente y modular con el fin de facilitar la evolución y mantenimiento del sistema de información federado.

La integración de los datos permite que, en caso de que existan nuevas fuentes de información o nuevos datos en los sistemas ya integrados a la federación, el acceso a ellos no sea tratado como un nuevo problema, sino que sean tratados de la misma manera que la información residente en cada una de las entidades que conforman la federación.

III.3.1. Tipos de componentes

Los FIS difieren en los tipos de componentes que pueden integrar. Los FIS pueden habilitar o deshabilitar la integración de componentes estructurados, semi-estructurados o sin estructura. Las fuentes estructuradas presentan un esquema pre-definido. Todos los datos son definidos a través del esquema al que pertenecen. Mas aún, el esquema dictamina el formato de todos los datos, los datos que no caen en este esquema no pueden ser integrados al conjunto de datos.

Una fuente de datos semi-estructurada también cuenta con una estructura de datos, pero esta estructura que no está pre-definida conforme a un esquema estricto (Buneman, 1997). Cada dato simple debe contar con su propia definición semántica. En un punto dado, la suma de todas las etiquetas de una fuente de datos semi-estructurada puede ser considerada como un esquema; sin embargo, este esquema puede potencialmente cambiar cada vez que un nuevo dato es agregado, mientras que en las fuentes estructuradas un cambio de esquema no ocurre frecuentemente.

Las fuentes de datos no estructurados, no cuentan con alguna estructura, tales como los documentos de texto.

III.3.2. Tipos de integración semántica.

Como ya se mencionó previamente, la integración permite encapsular fuentes de datos para el usuario con el fin de poder presentar la información que reside en los FIS de una manera

lógica y entendible. Busse *et al.* (1999) clasifica a los diferentes tipos de integración semántica como:

- Colección: Los datos de los componentes son recolectados sin cambio y sin datos equivalentes en diferentes colecciones.
- Fusión: la integración de los datos del componente es hecha por una simple extracción; pero en contraste a la integración de colección, la fusión de objetos es ejecutada para identificar entidades semánticamente equivalentes[3] provenientes de diferentes fuentes (Papakonstantinou *et al.*, 1996).
- Abstracción: los datos federados se basan en datos extraídos de los componentes. La necesidad de abstracción es en general causada por conflictos semánticos; abarca funciones de agregación de datos, entidades de reclasificación o incluso más procesos complejos racionales. Llevar a cabo el proceso de abstracción durante la integración implica que las operaciones de escritura no son posibles, debido a que las operaciones son difíciles de especificar.
- Complementación: Los datos no son solo derivados de aquellos que residen en los componentes, sino que algunos otros son agregados, los cuales describen el contenido o contexto semántico de los datos. Tal integración es usada para manejar componentes semánticos implícitos.

III.3.3. Transparencia.

La transparencia es, para el usuario final, considerada como el último logro de la integración. Un sistema de información perfectamente integrado da la ilusión a los usuarios de interactuar con un solo sistema de información central, consistente, homogéneo y con apariencia de ejecución local. Existen ciertos tipos de transparencia[4]:

---

[3] Dos objetos de datos son semánticamente equivalentes, si describen el mismo concepto del mundo real.

[4] La transparencia en este contexto denota el concepto de invisibilidad del problema; si la ubicación es transparente, entonces el usuario final no puede verlo (o al menos no necesita hacerlo).

- Transparencia de ubicación: los usuarios no necesitan conocer la ubicación física de información. Esto comprende la ubicación del sistema de cómputo y el nombre de la fuente de datos.
- Transparencia de esquema: los usuarios no necesitan conocer las diferentes denotaciones que las entidades o atributos tienen en diferentes fuentes de datos. Dado un escenario puramente relacional, los mismos no deben preocuparse por las diferentes relaciones y nombres de atributos, es decir, todos los conflictos lógicos son enmascarados.
- Transparencia del lenguaje: los usuarios no necesitan preocuparse de diferentes mecanismos de consulta y lenguajes. Esto comprende el lenguaje de consulta e implícitamente el modelo de datos y el mecanismo de acceso, es decir, ya sea que las cadenas de consulta sean ejecutadas por medio de un lenguaje de consulta declarativo como SQL[5] o por algunos métodos de aplicación iniciados vía RPC[6].

La heterogeneidad lógica puede ser soportada por la transparencia de esquema, mientras que la transparencia del lenguaje permite ocultar la heterogeneidad de interfaz. La transparencia de ubicación esta relacionada a la heterogeneidad técnica.

III.3.4. Estrategias de desarrollo de FIS.

Idealmente, una base de datos centralizada es desarrollada de la siguiente manera: primero, un análisis del problema es hecho para obtener la especificación de requerimientos. Esos requerimientos son expresados como vistas. Dado que diferentes requerimientos de diferentes usuarios de la aplicación planeada llevan a diferentes vistas, un esquema de base de datos homogéneo no se obtiene inmediatamente sino que necesita ser derivado de las vistas a través de la integración de las mismas.

---

[5] Acrónimo de Structured Query Language (Lenguaje de Consulta Estructurado) y es un estándar de lenguaje de consulta de ANSI (American National Standards Institute) para acceder y manipular sistemas de bases de datos.

[6] Acrónimo de Remote Procedure Call (Llamada de Procedimiento Remoto) y consiste en un técnica para construir aplicaciones distribuidas basadas en el modelo cliente-servidor.

El desarrollo de federaciones fuertemente ligadas también enfrenta el problema de encontrar un esquema global sustentable. Sin embargo, en contraste con una base de datos central, los sistemas de información federados no están diseñados desde cero, sino construidos en lo alto de sistemas existentes. Tomando esta propiedad en cuenta, se identifican dos estrategias para el diseño conceptual del esquema global (Leser, 2000; Sheth y Larson, 1990):

- Bottom-up. Los esquemas de las fuentes de datos son considerados como vistas que deben ser integradas en un esquema uniforme, homogéneo y global. La integración del esquema es mas complejo que la integración de las vistas dado que posiblemente se tienen que integrar esquemas muy grandes en vez de vistas simples así como considerar los datos actuales en las fuentes y no solo los esquemas.
- Top-down. El esquema global se diseña de la misma manera que como se diseñaría un esquema de una base de datos central. Es decir, se inicia de los requerimientos globales, resultando en vistas globales, y se termina con un esquema homogéneo. Los esquemas de las fuentes no son considerados en este proceso, pero deben ser relacionados al esquema global en un paso subsecuente.

III.3.4.1. Top-Down

El punto de inicio para este tipo de desarrollo es un requerimiento de nueva información que puede ser satisfecho por algunas fuentes de datos conocidas. Dado que las fuentes de datos aisladas son ineficientes y consumen tiempo, emerge el requerimiento para acceso integrado.

El acceso integrado inicia de la necesidad de nueva información a través del análisis de requerimientos. Este análisis resulta en la especificación de un esquema que trae consigo todas las estructuras de información necesarias para satisfacer los requerimientos. A pesar que es necesario conocer, en esta etapa, si hay fuentes de datos que pueden proveer los

datos actuales en tiempo de ejecución, sus estructuras, semántica y lenguajes son ignorados en primer lugar.

Una vez que el esquema global es completado, las fuentes de datos están altamente conectadas en el sistema. Cada fuente es considerada en aislamiento; su contenido es descrito relativo al esquema global usando una especificación de lenguaje de correspondencia. Las correspondencias son después usadas para descomponer y traducir cadenas de búsqueda.

Considerar las fuentes en aislamiento tiene ventajas en relación con el mantenimiento del sistema; particularmente permite una conexión/desconexión fácil de fuentes de datos. Los cambios en la estructura de las fuentes de datos pueden ser manejados únicamente cambiando las descripciones de correspondencia, manteniendo el esquema global sin afectación.

Sin embargo, el uso de la estrategia Top-Down tiene como consecuencia una menor coherencia del sistema; muchos sistemas son construidos en un contexto Web pero tales fuentes no están consientes de su integración en un contexto más grande, como por ejemplo el proyecto Agrega (Canabal-Barreiro y Sarasa-Cabezuelo, 2007) el cual es creado en un contexto Web pero con un acceso restringido a los sistemas que lo conforman . Esas fuentes no pueden informar al sistema de integración sobre cambios en su estructura (Leser, 2000; Guri-Rosenblit, 2002).

La estrategia Top-Down presenta muchas ventajas en escenarios donde las fuentes son cambiantes frecuentemente, son removidas o que nuevas sean agregadas, que el esquema de integración es infactible o muy caro o si los requerimientos globales en sí son muy cambiantes; esto es debido principalmente a que el esquema de integración es extremadamente vulnerable a cualquier cambio. De cualquier manera, la estrategia de desarrollo Top-Down resulta en una integración con bajo acoplamiento a comparación de la estrategia Bottom-Up.

III.3.4.2. Bottom-Up

Esta estrategia inicia con el requerimiento de proveer acceso integrado a un conjunto dado de fuentes de datos. Este conjunto esta predefinido y no se espera que cambie frecuentemente. El campo de acción del sistema de información federado es esencialmente la unión semántica del contenido de esas fuentes. El primer problema es la ejecución de esta unión y su representación en el esquema. Esto es llevado a cabo por la integración de esquema (Eder y Frank, 1994).

La integración de esquema es hecha en cuatro pasos: en el primero (pre-integración), todos los esquemas son transformados en un modelo de datos común; en el segundo paso (comparación de esquema), los esquemas son comparados entre ellos para encontrar y claramente identificar las correspondencias y conflictos existentes. Los conflictos que pueden ser resueltos re-estructurando los esquemas de la fuente son llevados a cabo en el tercer paso (adaptación de esquema); el cuarto paso finalmente une todos los esquemas en un solo esquema integrado. El esquema resultante debe ser: completo, correcto, mínimo y entendible.

Hay ocasiones donde una estrategia bottom-up es la elección correcta, especialmente si la integración es llevada a cabo como primer paso hacia la migración, es decir, si la federación es construida para soportar aplicaciones usando el esquema integrado mientras que aún se mantienen viejas aplicaciones vivas que utilizan los esquemas originales.

La estrategia Bottom-Up genera un sistema semánticamente bien integrado debido a que los componentes deben ser conocidos completamente antes del proceso de integración. Cualquier cambio posterior en la configuración conlleva a un nuevo proceso de integración (Klischewski, 2003).

En la estrategia top-down, el diseñador especifica correspondencias verticales de esquemas, es decir, correspondencias entre esquemas globales y de fuentes. En la estrategia bottom-up, el diseñador define correspondencias horizontales de esquemas, es decir,

correspondencias entre diferentes esquemas de fuente, los cuales son usados para derivar el esquema global. El procesamiento de cadenas de consulta requiere la existencia de correspondencias verticales, las cuales deben ser derivadas de las correspondencias horizontales en un paso extra.

La elección de cuál estrategia usar normalmente no puede ser elegida libremente, sino que está determinada por la situación en la que el nuevo desarrollo es buscado. Si, por ejemplo, se requiere acceso global de escritura, se deben considerar esquemas de fuente cuidadosamente en el diseño del sistema; esto puede ser llevado a cabo cómodamente por la estrategia bottom-up. Por otro lado, si las fuentes entran y salen de la federación frecuentemente, entonces la estrategia top-down es superior puesto que ofrece mayor flexibilidad.

Un problema particular en la construcción de un FIS es la necesidad de actualizar datos en las fuentes de datos a través del esquema global. Dichas actualizaciones solo son posibles si la conexión entre los esquemas global y de componente son muy estrechas, es decir, si las actualizaciones pueden ser propagadas de manera única (Busse *et al.*, 1999). La estrategia Top-Down raramente ofrece esta posibilidad y así mismo la estrategia Bottom-Up no mantiene correspondencias únicas.

III.3.5. Tipos de Sistemas de Información Federados.

Busse *et al.* (1999) proponen una clasificación de sistemas de información federados, tomando como referencia a su vez a Sheth y Larson (1990), así mismo Saavedra (2003) toma las ideas expuestas por Busse *et al.* (1999) y Sheth y Larson (1990) pero menciona que la integración de los diferentes esquemas de las bases de datos sigue siendo un tema de investigación activo. Menciona también que se ha producido una redefinición de los conceptos comúnmente usados en el contexto de integración de esquemas heterogéneos.

El primer cambio importante es el llamar a estos sistemas "Sistemas de información federados" en lugar de "Sistemas de bases de datos federadas", ya que muchas veces las fuentes no serán bases de datos. Así mismo menciona que existe un cambio en su clasificación.

La clasificación de Busse *et al.* (1999), y que Saavedra (2003) retoma, se divide en tres tipos: sistemas de información débilmente acoplados, sistemas de bases de datos federadas y sistemas de información basados en mediadores. La Tabla I describe brevemente las características de dicha clasificación:

Tabla I. Características de los Tipos de Sistemas de Información Federados.

	S. I. Débilmente Acoplados	**Bases de Datos Federadas**	**S. I. basados en Mediadores**
Tipos de heterogeneidad solucionados	Técnicos y Lenguaje	Todos excepto heterogeneidad de restricciones; dificultades en integración de heterogeneidades de esquema.	Todos
Transparencia en consulta	Lenguaje	Localización, esquema y, parcialmente lenguaje	Localización, esquema y lenguaje
Tipo de componentes	Estructurados	Estructurados	Estructurados
Métodos de acceso	Lenguaje de consulta	Lenguaje de consulta	Cualquiera
Restricciones de acceso	No	No	Si
Acceso de escritura	Si	Si	No
Acoplamiento	Débil	Fuerte	Fuerte
Tipos de integración semántica	Colecciones	Colecciones y fusiones	Colecciones, fusiones y a veces abstracciones
Metadatos necesarios	Técnicos, infraestructura	Lógicos, técnicos, semánticos	Lógicos, técnicos, semánticos
Bottom-Up vs. Top-Down	No aplica	Bottom-Up	Top-Down
Capacidad de evolución	Alta	Baja	Alta

De las características presentadas en la tabla anterior se puede formar la taxonomía de Sistemas de Información Federados mostrada en la Figura 5.

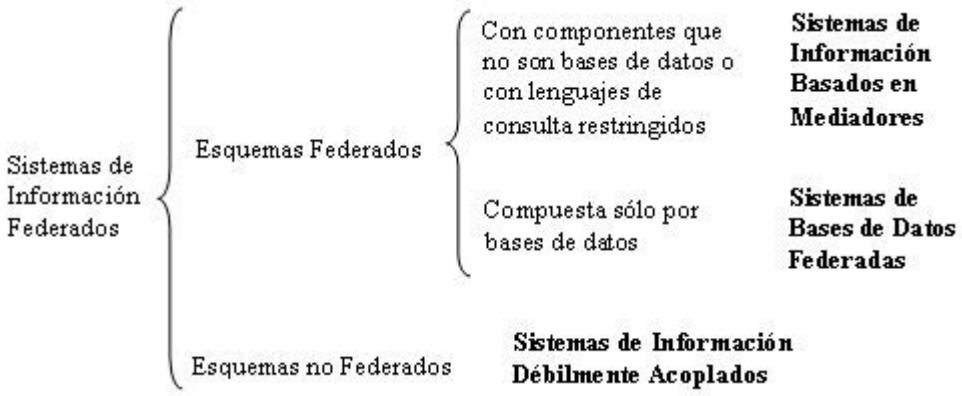

Figura 5. Clasificación de los Sistemas de Información Federados.

A continuación se describen dichos sistemas de información federados, presentando características sobre los mismos como su funcionamiento y algunos componentes de los mismos.

III.3.5.1. Sistemas de Información Débilmente Acoplados

Los sistemas de información débilmente acoplados no ofrecen un esquema federado, en lugar de esto, ofrecen un lenguaje de consulta multibase de datos para acceder a los componentes. Esto tiene la ventaja de que los componentes no ceden autonomía para participar en la federación; sin embargo, el usuario debe conocer el componente que desea acceder y por ende el elemento particular en el esquema de componente con su propio lenguaje de consulta.

Al proveer un lenguaje de consulta uniforme, la heterogeneidad técnica y de lenguaje es soportada por los sistemas de información federados. Todos los conflictos lógicos tienen que ser resueltos por el usuario.

Dado que no existe un esquema global, los cambios en los esquemas de componentes no afectan al sistema. Sin embargo, la falta de integración lógica implica que existan varias dependencias entre aplicaciones y componentes del sistema trayendo consigo problemas semánticos de los componentes.

III.3.5.2. Sistemas de Bases de Datos Federadas

Estos sistemas proveen funcionalidad clásica de un sistema de bases de datos. Esto incluye acceso de lectura y escritura para administración de los datos. El termino “base de datos” indica la relación con un sistema clásico de base de datos y como tal los componentes de un sistema de base de datos son fuentes estructuradas, las cuales son accedidas mediante lenguajes de consulta.

A diferencia de los sistemas débilmente acoplados, los sistemas de bases de datos federadas ceden cierta autonomía, por ejemplo, notificación sobre cambios y calendarización de información para transacciones de administración global.

Los sistemas de bases de datos federados ofrecen localización y transparencia de esquema completos a sus usuarios; sin embargo presentan una arquitectura estática con problemas en su evolución debido a la dependencia en el proceso de integración de esquema que no permite facilidad en la agregación y desagregación de componentes a la federación.

III.3.5.3. Sistemas de Información Basados en Mediadores.

Los sistemas de Información Basados en Mediadores (MBIS, por sus siglas en ingles) son sistemas fuertemente acoplados, por lo que un esquema federado es usado para proveer acceso integrado a la información de los distintos componentes (heterogeneidad semántica). A diferencia de los sistemas de bases de datos federadas, los MBIS proveen acceso de solo lectura.

Los sistemas de información basados en mediadores pueden considerar diversos mecanismos de integración como abstracción o agregación. Generalmente, un mediador no soluciona estos aspectos pero debe ocuparse al menos de alguno. Este tipo de sistemas presentan características descritas con mayor detalle en el Capítulo IV, que los hacen factibles de ser implementados en escenarios donde exista diversas fuentes de información heterogéneas, como el caso de estudio del presente trabajo, debido a que es posible dar soporte a la misma con la creación de un nuevo componente que funcione como interprete entre la propia fuente y la federación.

III.4. Resumen

En este capítulo se habló sobre los sistemas de información, los cuales son entidades que pueden o no estar asociados a una o más federaciones. Dichos sistemas de información son nombrados así debido a que los datos que dan a conocer a los demás miembros de la federación, no son solo datos como tal, o documentos de texto, sino que pueden ser cualquier tipo de elementos que sean de utilidad para el usuario que los ha solicitado.

Estos sistemas de información son catalogados en sistemas de información monolíticos, distribuidos o heterogéneos, donde las diferencias entre ellos caen en la ubicación física de la información que poseen y la cantidad de fuentes que tienen asociadas.

A su vez, los sistemas de información heterogéneos pueden ser conocidos como Sistemas de Información Federados, los cuales, a su vez, son catalogados en Sistemas de Información Débilmente Acoplados, Sistemas de Bases de Datos Federadas y Sistemas de Información Basados en Mediadores.

Los Sistemas de Información Federados permiten integración de componentes, y otorgan transparencia en la ubicación de las fuentes de información así como en el acceso a los

mismos. El desarrollo de los sistemas de información federados puede ser llevado a cabo utilizando dos estrategias: Bottom-Up y Top-Down.

Las estrategias de desarrollo difieren en que, básicamente, la última surge de una necesidad de información y a partir de ésta se procede con la integración de los componentes. La estrategia Bottom-Up se realiza cuando lo que se desea integrar son fuentes de información que ya existen y las cuales se desean unificar, lo que implica un sistema semánticamente bien integrado.

Los Sistemas de Información Débilmente Acoplados ofrecen un lenguaje de consulta multibase de datos para acceder a los componentes. Los Sistemas de Bases de Datos Federadas se enfocan a las bases de datos y como tal presenta todos los beneficios y funcionamiento de los sistemas de bases de datos.

Los Sistemas de Información Basados en Mediadores son sistemas que utilizan un módulo de software que permite un acceso integrado a distintos componentes. Estos sistemas son los que, con base en sus características, se consideran de utilidad para el caso de estudio tratado en el presente documento. Los MBIS son tratados de una manera mas detallada, en el siguiente capítulo.

Capítulo IV

Sistemas de Información Basados en Mediadores

En el presente capítulo se profundiza en un tipo de sistema de información: el sistema de información basado en mediadores. Se definen los componentes de los cuales consta este tipo de sistemas de información, así como las características de los mismos.

Se hace énfasis en la definición del término mediador, el cual es un componente imprescindible y del cual toman el nombre este tipo de sistemas, también se mencionan conceptos generales sobre la arquitectura de los mismos debido a que la arquitectura adoptada en este trabajo está basada en mediadores.

IV.1. Mediador

La necesidad de un acceso unificado e integrado a múltiples fuentes de información ha estimulado la investigación en lo que a mediadores se refiere. De manera muy general, un mediador es una fuente de información que pretende proveer de una interfaz uniforme a un número de fuentes de información en una capa inferior.

Los usuarios realizan una consulta al mediador el cual consulta a su vez a las fuentes de información en la capa inferior; este proceso involucra una selección de fuentes a ser consultadas. Esta tarea es llevada a cabo basada en lo que mediador “conoce” de las fuentes. Finalmente, el mediador tiene que combinar apropiadamente los resultados obtenidos y entregar la respuesta final al usuario.

Varios autores (Busse *et al.*, 1999; Tzitzikas *et al.*, 2001; Tzitzikas *et al.*, 2005) concuerdan en que Wiederhold (1992) fue quien introdujo el termino mediador y desde entonces ha sido utilizado en muchas publicaciones sobre proyectos y técnicas de integración de datos. En general, un mediador es un componente de software que debe ser ligero, flexible y reusable, el cual que media entre el usuario y la fuente física de datos.

Wiederhold (1992) define a un mediador como "un modulo de software que aprovecha el conocimiento codificado sobre algunos conjuntos o subconjuntos de datos para crear información para aplicaciones de una capa superior". Esta definición supone una arquitectura de federación de tres capas, en donde la capa inferior son las fuentes de datos, la capa superior es la interfaz para el usuario y la capa intermedia es el mediador.

Tzitzikas *et al.* (2001) y Tzitzikas *et al.* (2005) mencionan que un mediador es una fuente secundaria que puede sobrellevar las heterogeneidades que pueden existir entre dos o más fuentes de información así como proveer un acceso unificado a las mismas. Cita que un mediador tiene un número de articulaciones a las fuentes de información. Una articulación a una fuente es un conjunto de relaciones entre los términos de un mediador y los términos de una fuente de información. Dichas relaciones son definidas por el diseñador del mediador en el momento en que se esta llevando a cabo el diseño, y las mismas son almacenadas en el mediador.

Busse y Pons (2001) citan que los mediadores proveen un acceso de solo lectura homogéneo a un conjunto de fuentes de información heterogéneas; citan que un mediador provee de un punto de acceso de solo lectura homogéneo a una base de información dinámicamente cambiante de sistemas de información autónomos, distribuidos y heterogéneos. El mediador utiliza un esquema virtual global para la integración de las fuentes de datos. Básicamente se crea un esquema global que involucra información referente a las fuentes de datos y al cual el mediador accede.

En resumen, las definiciones propuestas por Tzitkas *et. al.* (2001) y Tzitkas *et. al.* (2005), Busse *et. al.* (1999) y, Busse y Pons (2001) son muy parecidas a la propuesta por Wiederhold (1992) por lo que, para fines de este trabajo, la definición que se adopta es la propuesta por Busse y Pons (2001).

Los mediadores son un componente que permite el acceso a fuentes de información heterogéneas y además otorga un punto de acceso a los mismos; este conjunto de componentes recibe el nombre de Sistemas de Información Basados en Mediadores (MBIS, por sus siglas en ingles).

De manera general, un Sistema de Información Basado en Mediadores es un sistema de información federado fuertemente acoplado que provee acceso de solo lectura a un conjunto de fuentes de información heterogéneas, el cual presenta las características de una federación y la funcionalidad de los mediadores.

IV.2. Sistema de Información Basado en Mediadores

Un Sistema de Información Basado en Mediadores (MBIS) es definido como una federación fuertemente acoplada, la cual provee un acceso unificado de solo lectura a fuentes de información heterogéneas y distribuidas.

La definición que Busse *et al.* (1999) propone para un MBIS es: “Un sistema de información basado en mediadores es un sistema que ofrece un mecanismo de acceso de solo lectura, homogéneo y virtual a una colección dinámicamente cambiante de fuentes de información heterogéneas, autónomas y distribuidas...”.

En esencia, la definición de Busse *et al.* (1999) concuerda con la propuesta por Leser (2000) quien a su vez añade que las partes que componen un MBIS son: wrappers[7], mediadores y las fuentes de información en sí: "Un sistema de información basado en mediadores (MBIS) es una FIS fuertemente acoplada que provee acceso de solo lectura a un conjunto de fuentes heterogéneas dinámicamente cambiantes" (Figura 6).

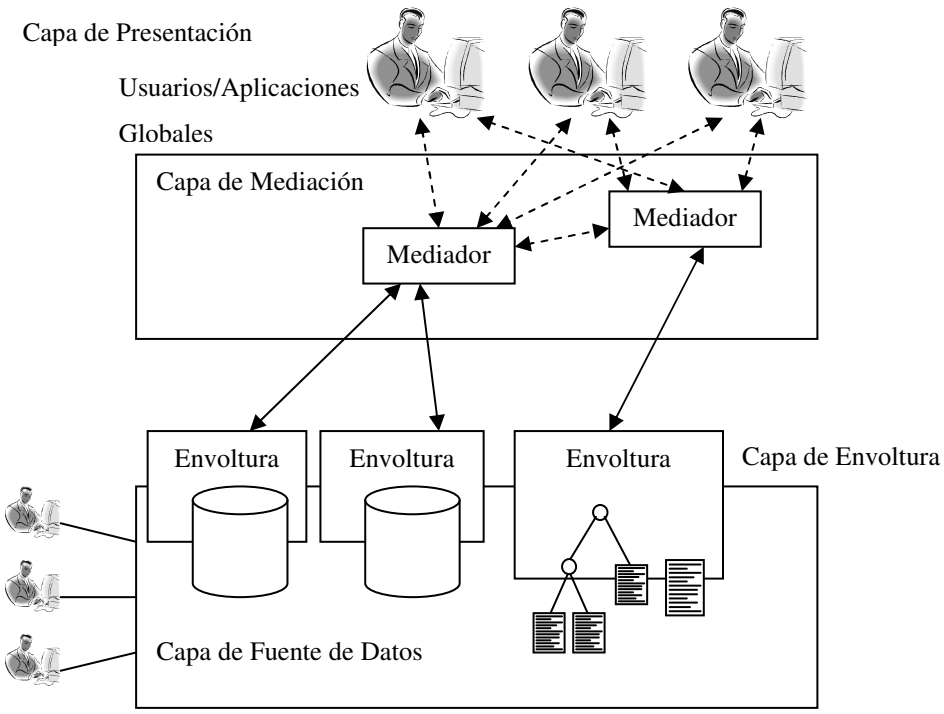

Figura 6. Arquitectura de un Sistema de Información Basado en Mediadores

Los componentes de un MBIS son envoltura, mediadores y las fuentes de datos. Un mediador traduce las consultas de los usuarios en combinaciones equivalentes de cadenas de consulta en las envolturas, en los cuales se tratan los problemas semánticos y estructurales. Otros tipos de heterogeneidad, concernientes a los protocolos de comunicación o representación sintáctica de los datos, deben ser resueltos antes de que la

[7] Término en inglés que significa "envoltorio o envoltura". En el contexto de un MBIS, se interpreta como un componente que encapsula la fuente de información o datos física permitiendo la comunicación entre el mediador y dicha fuente.

fuente sea integrada al MBIS. Para este fin, las fuentes cuentan con envolturas, las cuales ofrecen una interfaz que se adapta a los requerimientos de los mediadores.

Lesser (2000) menciona que existen dos tipos de MBIS: heterogéneos y homogéneos. En los MBIS homogéneos, todos los mediadores utilizan el mismo modelo de datos y lenguaje de consulta. Por lo tanto, un mediador es utilizable por otro en el MBIS correspondiente. En contraste, los MBIS heterogéneos contienen diferentes tipos de mediadores, por lo tanto, las envolturas deben proveer diferentes interfaces para diferentes mediadores, y los mediadores pueden necesitar ser encapsulados para ser utilizables por otros mediadores.

IV.2.1. Envolturas

Una envoltura transforma los datos representados en el modelo de datos de su respectiva fuente encapsulada, en una representación en el modelo de datos del mediador. Es decir, traduce consultas del lenguaje del mediador en consultas ejecutables por la fuente. Las envolturas son capaces de describir sus capacidades y de otorgar su información al medidor con el propósito de facilitar la tarea de selección al mediador. Usualmente es responsabilidad del mediador generar consultas que son ejecutadas por una sola envoltura.

Existen dos partes en la envoltura que son de especial interés para el mediador:

- Cada envoltura cuenta con un esquema propio (relacional). Los datos producidos por la envoltura se adhieren a este esquema, es decir, todos los datos son estructurados de acuerdo a este esquema.
- Cada envoltura es capaz de responder al menos algunas consultas predefinidas explicitas, lo que resulta en un conjunto de duplas consistentes de atributos con valores discretos.

El mediador solo utiliza un conjunto de cadenas predefinidas para responder a las consultas del usuario. Los esquemas de la envoltura son importantes para la selección de tales cadenas. Las envolturas presentan ciertas características en cuanto a su construcción:

- Las envolturas son específicas de la fuente. Su esquema es elegido basado en los datos almacenados en la fuente encapsulada y en la interfaz que la envoltura utiliza para acceder a la misma.
- La ubicación física de la envoltura no está determinada. Las envolturas pueden ser construidos en el lugar donde se encuentre la fuente de datos, donde este ubicado el mediador o en un tercer sitio.
- Aunque las envolturas son específicas de la fuente, pueden ser reutilizados. Por ejemplo, una envoltura para un RDBMS[8] puede ser utilizado para otras fuentes que utilicen el mismo RDBMS.
- Una fuente de datos puede ser accedida por medio de diferentes envolturas en un MBIS. Razones para esto pueden ser:
    - Aprovechar interfaces específicas. Diferentes envolturas pueden utilizar diferentes mecanismos de acceso ofreciendo puntos especiales. Por ejemplo, una fuente de datos puede proveer algunos tipos de consulta por medio de una interfaz CORBA[9] y otras consultas a través de una forma basada en Web.
    - Incrementar el mantenimiento a través de una separación de intereses. Diferentes envolturas para diferentes partes de la fuente son ventajosos si la fuente exporta muchas estructuras de datos complejas.

---

[8] Es el acrónimo en ingles para Sistema Administrador de Bases de Datos Relacionales.

[9] En inglés *Common Object Request Broker Architecture*, arquitectura común de distribución de peticiones de objetos. Es un estándar que establece una plataforma de desarrollo de sistemas distribuidos facilitando la innovación de métodos remotos bajo un paradigma orientado a objetos. Dirigido y controlado por el *Object Management Group* (OMG).

IV.2.2. Mediadores

Los mediadores sólo tienen un esquema y sólo tratan con fuentes de datos estructurados. Las consultas de los usuarios son ejecutadas contra los esquemas del mediador, pero los datos en sí no existen físicamente en lugar central, sino que el mediador recolecta la información apropiada en tiempo de consulta de las fuentes de datos.

La principal tarea de los mediadores es la traducción de cadenas de consulta expresadas en el esquema del mediador, a conjuntos de cadenas ejecutables por la envoltura (ver sección IV.1).

IV.3. Lenguaje de Especificación de Correspondencia

Los mediadores dependen de las correspondencias entre elementos de diferentes esquemas para poder ejecutar las consultas que se les ha encargado; tales correspondencias deben ser expresadas en algún lenguaje, a este lenguaje se le llama lenguaje de especificación de correspondencia (CSL, por sus siglas en inglés). La elección de un CSL es muy importante para un MBIS, dado que determina los tipos de conflictos de esquema que puede traer consigo. De este modo, la ejecución de la cadena debe encontrar planes correctos mediante la explotación del conocimiento científico que es expresado en reglas de un cierto CSL.

Diferentes CSL, que cuentan con diferencias de expresión, cuentan con diferentes algoritmos para ejecutar la cadena. Hull (1997) distingue dos tipos de CSL: Global-as-View (GaV) y Local-as-View (LaV). En esencia, LaV relaciona una cadena de mediador a una relación de envoltura, mientras que GaV relaciona una cadena de envoltura a una relación de mediador.

IV.3.1. Global-as-View

Permite la especificación de correspondencias entre relaciones simples del esquema de mediador y vistas en esquemas de la envoltura. Si más de una vista corresponde a una relación de mediador entonces se deben resolver los posibles conflictos indicando que la extensión de esta relación es esparcida sobre muchas fuentes (Figura 7). Existen dos soluciones posibles:

- Se especifican tantas correspondencias de vistas como sean necesarias, y el mediador usa un método genérico para combinar los resultados calculados por diferentes vistas. Esto puede ser basado en reglas de equivalencia de objetos definidas por el usuario.
- Solo una correspondencia es especificada, direccionando todas las vistas relevantes de una vez. En este caso, cada correspondencia puede definir su propia forma de como los resultados de diferentes envolturas son combinados.

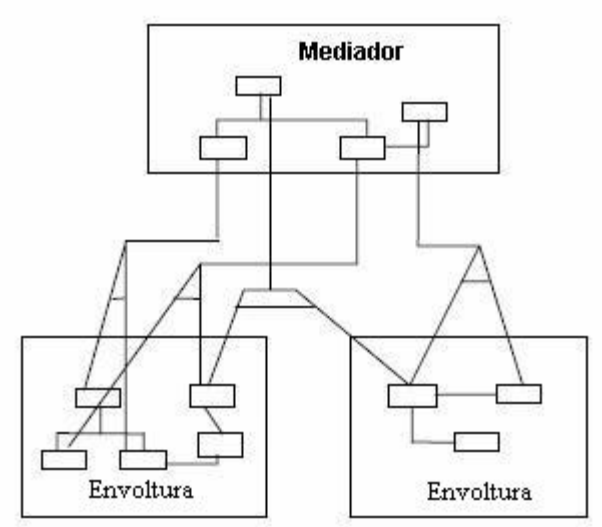

Figura 7. Global-as-View. Los ángulos significan las definiciones de las vistas. Los esquemas de mediador son definidos como vistas en el esquema de envoltura.

Por ejemplo, Papakonstantinou *et al.* (1996) utiliza dos métodos para llevar a cabo este tipo de correspondencia; el primero de ellos es llamado “fusión de objetos”, el cual asigna claves semánticas a cada objeto en cada vista. El mediador combina los datos que están

siendo asignados con la misma clave en tiempo de ejecución, independientemente de la envoltura de la cual resulta el dato. Papakonstantinou *et. al.* (1996) llama al segundo método: "fusión por uniones exteriores" (en inglés: *fusion by outer-joins*).

GaV es una extensión directa de un motor de bases de datos central para el caso distribuido. Es utilizado en muchos proyectos, tales como: TSIMMIS (García-Molina *et al.*, 1997), DISCO (Tomasic *et al.*, 1996) o IRO-DB (Fankhauser *et al.*, 1998).

IV.3.2. Local-As-View

Este planteamiento permite correspondencias entre una relación simple de un esquema de envoltura con una vista en el esquema de mediador. Esto es exactamente contrario al enfoque GaV. En LaV, cada relación de mediador puede aparecer en muchas vistas correspondientes a diferentes relaciones de envoltura. Cada correspondencia contribuye a la extensión total de la relación de mediador (Figura 8).

LaV, de acuerdo con la literatura, fue primero descrita en Tsatalos *et al.* (1996), y es, por ejemplo, utilizada en "Information Manifold" (Levy *et al.*, 1996; Levy *et al.*,1996b) y en Infomaster (Genesereth *et al.*, 1997; Duschka y Genesereth, 1997). Este enfoque tiene fuerza en ambientes con evolución frecuente de fuentes, como la Web; sin embargo, la ejecución de cadenas con reglas LaV es considerablemente mas compleja que con reglas GaV.

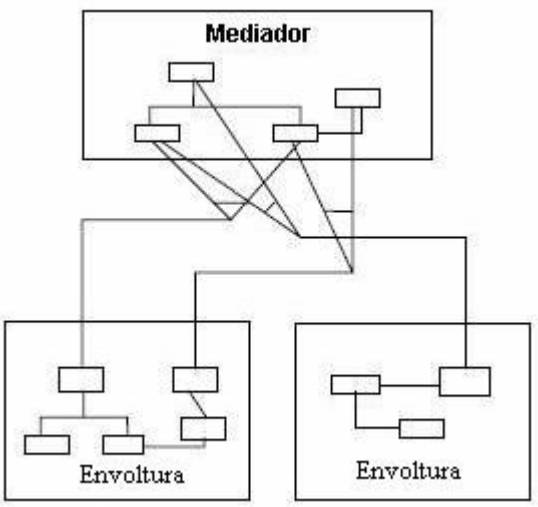

Figura 8. Local-as-View. Los ángulos significan las definiciones de las vistas. Los esquemas de envoltura son definidos como vistas en el esquema de mediador.

IV.3.3. Comparación

Las diferencias entre GaV y LaV son mejor descritas mediante el análisis de sus respectivas percepciones sobre integración.

En LaV, el esquema de mediador supone ser una estructura estable y las fuentes son adheridas y borradas basadas en qué tan útiles sean para el mediador. Esta estrategia es bien manejada por el desarrollo top-down de MBIS. Construir un sistema con el enfoque LaV, necesariamente inicia con definir el esquema de mediador. En un segundo paso, las fuentes de datos apropiadas son descubiertas y son definidas sus correspondencias con el esquema de mediador. Los planteamientos LaV nacieron en proyectos enfocados a las fuentes Web, donde los servidores y las fuentes de datos son informales y aparecen o desaparecen con alta frecuencia.

En contraste, el enfoque GaV empieza de la necesidad de integración de un conjunto de fuentes, tales como diferentes bases de datos de los departamentos de una empresa. Estas fuentes son percibidas como la parte más estable del MBIS. Esto guía a una estrategia de

desarrollo bottom-up; por lo tanto, todos los métodos de integración de esquema resultan en reglas GaV.

IV.4. Resumen

En el presente capitulo se trataron temas relacionados a los Sistemas de Información Basados en Mediadores (MBIS), para lo cual se tomó como punto de partida que los MBIS son un tipo especial de sistema federado (tratado en el capitulo III) el cual tiene como componente principal al mediador (de ahí su nombre).

El mediador es un componente de software que permite la interacción del usuario con la fuente de datos o información; el mediador debe ser ligero, flexible y reusable. Existen varios autores que mencionan la definición de mediador, sin embargo todos concuerdan en que Wilderhold (1992) fue quien introdujo el término por primera vez. De manera general un mediador traduce las consultas de los usuarios en consultas ejecutables por la envoltura.

La envoltura, que es otro término que hace referencia a un componente de un MBIS, es un componente que "encapsula" la fuente de información a que hace referencia, permitiendo que las consultas hechas por el usuario sean ejecutadas en la fuente de información. La envoltura presenta ciertas características que lo diferencian haciéndolo único en el MBIS, las cuales son:

- Envoltura única, es decir, que por cada fuente de información existe una envoltura que permite la comunicación con la fuente asociada.
- Su ubicación física no esta definida, es decir, que la envoltura puede ser implementada en el lugar físico donde se ubique la fuente de información, el mediador o en algún lugar ajeno a los anteriores.
- Posible reutilización, es decir, que a pesar de que existe una envoltura por cada fuente de información, si una fuente 'A' es administrada por un DBMS en

especifico y si existe una fuente ‘B’ con el mismo DBMS, la envoltura puede ser utilizada para las dos fuentes antes mencionadas.

Los MBIS son sistemas federados diseñados con la estrategia top-down y una característica que presentan es que solo permiten el acceso de solo lectura a los datos ubicados en las fuentes de información.

La manera por la que se comunican los mediadores con las envolturas, es mediante un lenguaje de especificación de correspondencias (CSL). Hull distingue dos tipos de CSL: Local-as-View (LaV) y Global-as-View (GaV).

- Global-as-View. Bajo este tipo de lenguaje, se relacionan las envolturas con una relación de mediador, en otras palabras, se definen ciertas cadenas predeterminadas en la envoltura para una consulta en específico hecha en el mediador.
- Local-as-View. En este caso el funcionamiento es inverso a GaV, es decir, que varias cadenas hechas en el mediador corresponden a una en especifico en la envoltura.

La elección de cuál tipo de CSL utilizar, dependerá de las necesidades a satisfacer con el lenguaje de correspondencias, trayendo consigo las propiedades del lenguaje elegido.

El funcionamiento de un sistema de información basado en mediadores utiliza estándares para poder integrar de una manera apropiada, las diferentes fuentes de información. Estos estándares son definidos con el propósito de permitir la comunicación entre las capas del mismo MBIS.

La capa de fuentes de información, como ya se ha mencionado, pueden ser bases de datos, sistemas de archivos o cualquier otro tipo de componente capaz de albergar entidades utilizadas para crear información, sin embargo para este trabajo dichos componentes son definidos como repositorios de objetos de aprendizaje.

Capítulo V

Objetos de Aprendizaje

En capítulos anteriores se ha hecho mención sobre características y conceptos que involucran a una federación. De manera especifica, se trataron temas referentes a los lugares donde reside la información que puede satisfacer la necesidad de la misma que tiene el usuario.

Estos lugares fueron tratados como sistemas de bases de datos, los cuales tienen características propias de una base de datos; sin embargo, en el marco del presente trabajo, se habla sobre búsquedas de objetos de aprendizaje.

En este capítulo se tratan temas referentes a los objetos de aprendizaje, contemplando su definición, características y los lugares en lo que se almacenan, así como los descriptores correspondientes, llamados metadatos, los cuales son utilizados para facilitar la ubicación y manejo de dichos objetos.

V.1. Definición

Los objetos de aprendizaje (OA) son elementos para la instrucción, aprendizaje o enseñanza basada en computadora. López (2005) menciona que los OA no son realmente una tecnología, más propiamente dicho son una filosofía fundamentada en la corriente de las ciencias de la computación conocida como orientación a objetos.

En el proyecto JORUM+ Project (2004) dice que “un OA es cualquier recurso que puede ser utilizado para facilitar la enseñanza y el aprendizaje y que ha sido descrito utilizando metadatos” (ver sección V.5).

El Comité de Estandarización de Tecnología Educativa (IEEE, 2001), considera a un objeto de aprendizaje como “una entidad, digital o no digital, que puede ser utilizada, reutilizada y referenciada durante el aprendizaje apoyado con tecnología”; de acuerdo con Wiley (2002) los objetos de aprendizaje son “cualquier recurso digital que puede ser reutilizado para apoyar el aprendizaje”.

Se dan como ejemplos de OA a los contenidos multimedia, el contenido instruccional, los objetivos de aprendizaje, software instruccional, personas, organizaciones o eventos referenciados durante el aprendizaje basado en tecnología (IEEE, 2001). Otros autores son menos específicos en cuanto a recursos del campo educativo, como González (2005) que considera como OA a archivos de texto, ilustraciones, vídeos, fotografías, animaciones y otros tipos de recursos digitales. Por su parte, el JORUM+ Project (2004) dice que como ejemplos pueden ser incluidos una imagen, un mapa, una pieza de texto, una pieza de audio, una evaluación o más de uno de estos recursos; dice además que hay que resaltar que se consideran extractos o sólo parte de los recursos y es posible no tomar en cuenta el recurso completo como tal, así también hace hincapié en que un OA puede ser el conjunto de dos o más recursos.

Sin embargo para fines de este trabajo, se tomará la definición propuesta por López (2005) quien cita que “cualquier recurso con una intención formativa, compuesto de uno o varios elementos digitales, descrito con metadatos, que pueda ser utilizado y reutilizado dentro de un entorno de ambiente de aprendizaje puede considerarse un OA”.

Menciona, también, que existen ciertos beneficios que un objeto de aprendizaje puede tener en un contexto educativo:

- Flexibilidad: el mismo recurso puede utilizarse en distintos contextos.
- Administración del contenido: se facilita debido a que los recursos están descritos con metadatos que permiten su control.
- Adaptabilidad: facilita al diseñador poder seleccionar y componer recursos según la aplicación.
- Código abierto: que elimina los problemas de incompatibilidad entre plataformas.

Un OA podrá utilizarse y/o reutilizarse dependiendo de la claridad de sus objetivos y su facilidad de integración en diversas aplicaciones, tanto por su contenido como por la descripción del mismo, es decir, su granularidad (ver sección V.3) y sus metadatos (ver sección V.5).

V.2. Atributos de los objetos de aprendizaje.

Los objetos de aprendizaje no pueden ser creados como un recurso de información aislado, sino que deben ser concebidos para interacción en un contexto de aprendizaje, fáciles de localizar, utilizar, almacenar y compartir. Por lo que los recursos deben ser: (Rehak y Mason, 2003):

- Reutilizables. El recurso debe ser modular para servir como base o componente de otro recurso, así como también contar con una tecnología, una estructura y los componentes necesarios para ser incluido en diversas aplicaciones.
- Accesibles. Pueden ser indexados para una localización y recuperación más eficiente, utilizando esquemas estándares de metadatos.
- Interoperables. Pueden operar entre diferentes plataformas de hardware y software.
- Portables. Pueden moverse y albergarse en diferentes plataformas de manera transparente, sin cambio alguno en estructura o contenido.
- Durables. Deben permanecer intactos a las actualizaciones de software y hardware.

Estos atributos dan sentido a las ideas plasmadas sobre la utilización y funcionamiento de los objetos de aprendizaje. López (2005) cita que la modularidad debe caracterizar a los OA, aumentando su versatilidad y funcionalidad. Así mismo menciona que la creación de los OA no es una tarea sencilla, pero los esfuerzos y costos para la producción de los OA, se equilibran con las veces que el OA puede ser reutilizado.

V.3. Granularidad

Aunque se menciona que un OA es "una pieza pequeña" o un recurso "modular", una dimensión precisa no puede ser especificada. López (2005) menciona que el tamaño de un OA es variable, y toma este concepto como la definición de granularidad.

Duncan (2003) maneja el termino "granularización" definiéndolo como el tamaño de los objetos de aprendizaje: "La granularización es una condición necesaria de los objetos de aprendizaje para que los mismos puedan ser compartidos y reutilizables". La forma en la que los recursos se agregan o unen entre sí puede ayudar a definir su granularidad, también lo puede ser su tamaño en relación al número de páginas, de duración o tamaño del archivo. Sin embargo, considera que el mejor criterio para definir la granularidad de un objeto es por sus propósitos u objetivos.

Ambos autores concuerdan en que la granularidad se refiere al tamaño del objeto de aprendizaje, sin embargo, el tamaño no puede ser delimitado, es decir, no es posible determinar la cantidad de información o elementos que un OA deba contener. El autor del objeto debe ser el encargado de poder abstraer la información necesaria para considerar al OA, que se pretende crear, como tal.

V.4. Reutilización de los Objetos de Aprendizaje

La característica más notable en las definiciones sobre objetos de aprendizaje, es la reutilización. Este concepto esta vinculado a la definición de reutilizar componentes de software, donde se trata de utilizar elementos de software previamente desarrollados para generar un nuevo producto.

Para lograr la reutilización se requiere tener un diseño, un desarrollo y una documentación que aseguren un alto nivel de calidad en el producto desarrollado. Estos mismos puntos son necesarios para la reutilización en el contexto de objetos de aprendizaje.

Debido a la modularidad de los OA y a su independencia de otros recursos, la utilización de los mismos en diferentes aplicaciones es relativamente sencilla. El gran potencial de la reutilización de los OA es poder aprovechar los contenidos que han desarrollado otros para formar nuevos recursos (López, 2005).

Para lograr la reutilización, así como también para lograr los atributos descritos (sección V.2), el objeto de aprendizaje debe contar con los metadatos que le permitan ser identificado, organizado y recuperado, entre otros aspectos; pero lo más importante es que esos metadatos estén basados en un estándar, con el fin de asegurar su compatibilidad e interoperabilidad con los sistemas que puedan reutilizar a dichos objetos (López *et al.*, 2005).

Para que un objeto de aprendizaje sea reutilizable debe estar siempre asociado al recurso, los metadatos que lo describen, y quien lo utiliza debe encontrarle los propósitos u objetivos en un contexto particular de aprendizaje (Rehak y Mason, 2003).

Sin embargo en la reutilización se tienen algunos problemas por la posible combinación de OA y detalles como diferencias en aspectos gráficos o sistemas de notación, no pueden dar los resultados esperados en la experiencia de aprendizaje. Es por esto que es recomendable

contar con políticas de consistencia y homogeneidad de los contenidos educativos (López, 2005).

Los metadatos son una parte intrínseca de los OA, por lo que la reutilización de los OA en gran medida dependerá de los metadatos, dado que es en base a éstos que se pueden determinar algunas de las características de los OA (tanto funcionales como tecnológicas y de contenido) y de esta manera decidir si es posible su integración con otros sistemas que pretendan utilizarlos así como con otros OA. Los metadatos son descritos con mayor detalle a continuación.

V.5. Metadatos

Los metadatos son un conjunto de atributos o elementos necesarios para descubrir un recurso. Por medio de los metadatos se tiene un primer acercamiento con el objeto, obteniendo conocimiento de sus principales características. Los metadatos además son útiles en los recursos que no son textuales.

Hillman (2005) apunta que los metadatos han estado presentes desde que los primeros bibliotecarios hicieron las listas de los recursos de información y anota que el término "meta" proviene del griego que significa "al lado de, siguiente, después, con" pero más recientemente los usos latinos y sajones lo utilizan para denotar algo trascendental o fuera de lo normal.

Una similitud a los metadatos, en el ámbito educativo, puede ser una ficha bibliográfica en la que se tiene toda la información que describe al recurso y se puede decidir si se consulta o no sin haber tenido contacto directo con el libro (u otro recurso documental), esto hace más fácil y ágil ubicar el recurso que se desea consultar.

López (2005) menciona que los metadatos no solo son descriptivos, sino que también pueden ser administrativos y de estructura:

- Metadatos descriptivos: tienen propósito de descubrimiento (cómo se encuentra un recurso), identificación (cómo distinguir un recurso de otro), y selección (cómo determinar que un recurso cubre una necesidad particular). Los metadatos descriptivos sirven también para formar colecciones de recursos similares.
- Metadatos administrativos: es información que facilita la administración de los recursos. Incluyen información sobre cuándo y cómo fue creado el recurso, quién es el responsable del acceso o de la actualización del contenido así como información técnica, como la versión de software o el hardware necesario para la utilización de dicho recurso.
- Metadatos estructurales: sirven para identificar cada una de las partes que componen al recurso y definen la estructura que le da forma. Por ejemplo, un libro, que contiene capítulos y páginas, se puede etiquetar con metadatos que identifican cada parte y la relación que guardan entre ellas.

Se han gestado iniciativas para la formalización del uso de metadatos a través del desarrollo de esquemas. Los esquemas de metadatos consisten en un conjunto de reglas semánticas, sintácticas y de contenido que deben seguirse para conformar el conjunto de metadatos de un recurso (López, 2005).

En el ámbito de aprendizaje electrónico se ha desarrollado un estándar para la descripción de los objetos de aprendizaje, IEEE LOM (Learning Object Metadata) (IEEE, 2001; Berlanga y García, 2004); su objetivo es servir de guía en el marcado de recursos educativos para con ello potenciar su búsqueda, evaluación, obtención y utilización. En LOM se especifica la sintaxis y la semántica de los atributos necesarios para describir los objetos de aprendizaje. Este estándar está compuesto de nueve categorías de metadatos, que agrupan elementos con los que se ha pretendido una descripción completa de los recursos educativos (ver Apéndice B).

Con el uso de esquemas estándares de metadatos se busca la reutilización de recursos y la interoperabilidad entre los sistemas involucrados, para que esto sea posible es necesario que los metadatos sean creados en base a estándares que permitan su interpretación y utilización en diferentes sistemas y plataformas, siguiendo las especificaciones o modelos necesarios para permitir su interoperabilidad.

V.6. Normalización de datos

López (2005) dice que dar valores a un metadato puede parecer trivial, sin embargo, esto es una tarea de expertos, principalmente especialistas en el manejo de información, ya que para la recuperación de dichos recursos la búsqueda se hará sobre los datos capturados.

Los lineamientos para el llenado de metadatos están especificados en los estándares, además, con el uso de interfaces de captura se elimina en gran medida el uso del criterio, permitiendo introducir valores predefinidos para cada campo y marcando errores cuando existe alguna inconsistencia con el estándar.

La normalización como complemento a la aplicación de especificaciones y estándares reporta importantes beneficios para la reutilización y la interoperabilidad de recursos y sistemas, pero lo más importante es que deja los datos preparados para futuras aplicaciones

V.7. Repositorio de Objetos de Aprendizaje

Con la aparición de los ambientes de aprendizaje se han tenido necesidades particulares para gestionar los contenidos educativos. Los repositorios de objetos de aprendizaje (ROA) comienzan a posicionarse como importantes herramientas que tienen como funciones resguardar los recursos y hacerlos disponibles, tanto para diversos usos como para compartirlos con otras aplicaciones. La utilización de los ROA facilita tanto el flujo de contenidos como la expansión de servicios.

Los repositorios de objetos de aprendizaje tienen su origen en la necesidad de buscar una solución particular que facilite la recopilación, el acceso y el compartir recursos educativos, en la que, apegándose a las necesidades específicas del sector, se tenga un sistema de almacenamiento de contenidos que se integre y comunique fácilmente con los otros sistemas que operan en los ambientes de aprendizaje en línea (McLean y Lynch, 2003).

En López (2005) se cita que un repositorio es un concepto tan amplio que va desde sencillos sistemas de almacenamiento hasta complejos entornos que incorporan, además de los sistemas de almacenamiento, conjuntos de herramientas que ayudan al proceso de reutilización.

Daniel (2004), a partir de los términos "repositorio digital", "objeto de aprendizaje" y "metadato" dice que "los repositorios de objetos de aprendizaje son bases de datos con búsquedas que alojan recursos digitales y/o metadatos que pueden ser utilizados para el aprendizaje mediado". El JORUM+ Project (2004) adopta la siguiente definición: "Un ROA es una colección de OA que tienen información (metadatos) detallada que es accesible vía Internet. Además de alojar los OA los ROA pueden almacenar las ubicaciones de aquellos objetos almacenados en otros sitios, tanto en línea como en ubicaciones locales".

Las definiciones, en un sentido general no difieren mucho entre sí y dejan ver claramente que estos repositorios, sean bases de datos o catálogos, están creados para ser utilizados en un proceso de enseñanza, lo cual lleva a que los ROA se vean como facilitadores claves para incrementar el valor de los recursos de aprendizaje dando la oportunidad a reutilizar, reorientar y hacer reingeniería para cubrir las necesidades del usuario final (Porter *et al.*, 2002).

V.7.1. Tipos de Repositorios de Objetos de Aprendizaje

Rehak y Mason (2003) consideran que existen dos tipos de repositorios, los cuales se fundamentan en la manera en la que sus recursos son concentrados, estos tipos son:

- Los que contienen los objetos de aprendizaje y sus metadatos. En éstos, los objetos y sus descriptores se encuentran en un mismo sistema e incluso alojados en un mismo servidor.
- Los que contienen sólo los metadatos. El repositorio contiene sólo los descriptores y se accede al objeto por medio de una referencia a su ubicación física que se encuentra en otro sistema o repositorio de objetos.

Downes (2004) concuerda con la clasificación de Rehak y Mason (2003), asignando los nombres centralizado y distribuido. Los más comunes son los centralizados, en los que los metadatos están contenidos en un mismo servidor, mientras que los distribuidos utilizan varios servidores que se comunican entre ellos para intercambiar los metadatos contenidos en los mismos.

El grupo de desarrollo Edutools (Leslie *et al.*, 2004) realizó un estudio a productos de software para ROA, considerando ciertos criterios que, después de ser analizados por expertos en la materia, se propusieron como características deseables, aunque de manera general, estas últimas (dadas a continuación) pueden tomarse en cuenta para tener en mente todo lo que un ROA pueda involucrar:

- Herramientas de búsqueda. Toma en consideración la búsqueda a través de palabras clave o algunos otros metadatos; la posibilidad de que el usuario pueda realizar exploraciones en listados predefinidos en algún tipo de categoría o clasificación, así como la capacidad del sistema para notificar a los usuarios sobre determinados eventos en el repositorio.

- Herramientas de recopilación. Creación de marcadores de recursos o colecciones personales y la posibilidad de creación de paquetes con varios recursos.
- Colectividad y evaluación. Posibilidad de que los usuarios puedan evaluar formal o informalmente un OA, mecanismos para registrar los contextos en los que el OA ha sido utilizado y listas de OA que el usuario desearía que se incluyeran o se modificaran.
- Meta-etiquetado. Herramienta de etiquetado, soporte de estándares y/o varios esquemas. Mecanismo de identificación única de los recursos especialmente importantes en colecciones federadas.
- Administración de contenidos. Seguimiento del flujo de creación y publicación de un OA, control de versiones y funciones de almacenamiento así como ciertas herramientas de autoría.
- Administración y cumplimiento de derechos digitales de autor. Registro, transmisión, interpretación y el hacer cumplir los derechos de autor, así como un sistema de pago, en caso de ser necesario.
- Presentación y salidas de consorcio. Accesibilidad, salidas en múltiples formatos para diferentes dispositivos, cambios de apariencia de la interfaz, soporte de caracteres de diferentes idiomas, habilidad para servir como puerta de entrada para varias colecciones y transformación de formatos.
- Integración e interoperabilidad. Federación y búsqueda en otros repositorios, soporte de servicios Web y de aplicaciones API[10] que puedan extraer información de actividades en el repositorio.
- Consideraciones técnicas. Autentificación, autorización y personalización, informe de uso, soporte para diferentes sistemas operativos, especificaciones de: la base de datos requerida por el repositorio, escalabilidad, arquitectura del modelo de software, requisitos técnicos y humanos para su puesta en marcha.

---

[10] Acrónimo de Application Program Interface (Interfaz de programación de aplicaciones). Representa una interfaz entre componentes de software. Conjunto de funciones y procedimientos que ofrece una biblioteca para ser utilizado por otro software como una capa de abstracción.

- Costo/Licenciamiento/Otros. Información de la compañía u organización que provee el software, número de instalaciones, modelo de costo o licenciamiento.

Looms y Christensen (2002) propusieron un conjunto básico de funciones que proveen acceso a objetos de aprendizaje en un ambiente seguro. Las funciones son:

- Buscar/Encontrar. Se refiere a la habilidad para ubicar un objeto de aprendizaje apropiado. Se puede incluir, también, la habilidad de navegar[11] por el repositorio.
- Pedir. Un objeto de aprendizaje que ha sido encontrado.
- Recuperar. Recibir un objeto de aprendizaje que ha sido pedido.
- Enviar. Entregar un objeto de aprendizaje a un repositorio para su almacenamiento en el mismo.
- Almacenar. Poner un objeto de aprendizaje enviado en un registro de datos bajo un identificador único que permite localizarlo.
- Colectar. Obtener metadatos de los objetos en otros repositorios mediante búsquedas federadas.
- Publicar. Proveer los metadatos a otros repositorios.

En adición a las funciones presentadas, un repositorio debe manejar puntos relacionados con DRM[12], obteniendo un identificador único global para cada objeto de aprendizaje así como autentificación para un acceso seguro a objetos de aprendizaje existentes.

V.7.2. Iniciativas de Repositorios de Objetos de Aprendizaje

López (2005) considera que la creación de repositorios de objetos de aprendizaje es relativamente reciente, y que a tomado fuerza a inicios de esta década. Leslie *et al.* (2004)

---

[11] El termino navegar se refiere a poder explorar los recursos contenidos en el repositorio.

[12] Acrónimo de Digital Rights Management (Gestión de derechos digitales). Es un término usado para tecnologías que controlan cómo es usado el contenido digital.

afirman que el mercado de software para aplicaciones de ROA, esta aun inmaduro, pero que a pesar de ello, el crecimiento ha sido rápido. La Tabla II muestra una lista de algunos de los principales proveedores e instituciones académicas involucradas en el desarrollo de repositorios (Looms y Christensen, 2002).

Tabla II. Proveedores e instituciones mayormente reconocidas en el desarrollo de ROA.

Proveedores	**Instituciones**
Artesia	Cornell University
IBM	National Science Foundation
Sun Mycrosystems	Old Dominion University
EMC	Simon Frasier University
Learning Object Network	University of Alberta
Microsoft Corporation	University of Calgary
Digital Concept, Inc.	University of Wisconsin

Existen repositorios que han formado sus colecciones mediante asociaciones entre grupos o mediante aportaciones individuales con el simple propósito de compartir el o los recursos creados. Existen, también, iniciativas que trabajan en propuestas para la interoperabilidad entre repositorios, con el propósito de formar redes de sistemas distribuidos que permitan búsquedas federadas (Hatala *et al.*, 2004).

A continuación se describen algunas iniciativas sobre repositorios, así como de propuestas de redes interoperables:

- MERLOT (Multimedia Educational Resource for Learning and Online Teaching). Es un repositorio centralizado que contiene sólo los metadatos y apunta a los objetos ubicados en sitios remotos. Es independiente y funciona como un portal de objetos de aprendizaje. Provee búsquedas y otros servicios (personalización, importación, exportación de objetos). Cualquier usuario puede tener acceso a los objetos contenidos en MERLOT pero sólo los miembros contribuyen agregando objetos. Para ser miembro no se requiere más que inscribirse y no se adquiere ninguna responsabilidad. La revisión por pares es una actividad que MERLOT

utiliza para evaluar la calidad de los objetos agregados. Disponible en http://www.merlot.org/merlot/index.htm.

- CAREO (Campus Alberta Repository of Educational Objects). Repositorio centralizado de objetos de aprendizaje multidisciplinarios de profesores de Alberta en Canadá. Es un repositorio independiente que da acceso a objetos remotos y locales mediante los metadatos contenidos en su colección. Cualquier usuario puede tener acceso a los objetos, pero los miembros tienen servicios adicionales. No existe un costo por pertenecer al grupo. Disponible en http://www.careo.org/.
- SMETE (Science, Mathematics, Engineering and Technology Education). Repositorio distribuido que integra de forma federada las colecciones de varias bibliotecas de recursos educativos. El acceso es libre para la consulta. Disponible en http://www.smete.org/smete/.
- GEM (Gateway to Educational Materials). Es un proyecto del Departamento de Educación de los Estados Unidos, originalmente conocido como National Library in Education Advisory Task Force. Está orientado a la interoperabilidad entre múltiples bases de datos por medio del uso de módulos que extraen los metadatos de los objetos en formato propio GEM. Disponible en http://www.thegateway.org/.
- CeLeBraTe (Context eLearning with Broadband Technologies). Proyecto desarrollado para los ambientes de aprendizaje virtuales de la European Learning Network, con el propósito de intercambiar los recursos digitales educativos de sus miembros. La idea que manejan es la de un repositorio centralizado pero cada miembro puede conservar, total o parcialmente, la administración local de los metadatos de su colección. Las búsquedas son realizadas en el repositorio central así como en los repositorios locales. Disponible en http://celebrate.eun.org/eun.org2/eun/en/index_celebrate.cfm.
- ELENA/Edutella. Proyecto europeo que propone mediadores de servicios educativos llamándolos: Smart Spaces, los cuales permiten la integración de servicios heterogéneos de aprendizaje (repositorios, sistemas de tutoría, entre otros). Disponible en http://www.elena-project.org/.

- eduSourceCanada. Es una propuesta para crear una red de repositorios en Canadá, uniendo los principales, creados en ese país, con una infraestructura abierta e interoperable. Soporta una amplia variedad de servicios y promete sistemas fáciles de usar y comunicar. Disponible en http://www.edusource.ca/.

La Tabla III presenta un resumen de las características de los ROA y redes interoperables presentadas anteriormente.

Tabla III. Características de ROA y proyectos asociados

Repositorio	**Nivel Escolar**	**Organización/ Pais**	**Acceso**	**Metadatos**	**OA**
MERLOT	Superior	Internacional	Abierto	Locales	Distribuidos
CAREO	Superior	Universidad de Calgary/ Canadá	Abierto	Locales	Distribuidos
SMETE	k-12[13]	SMETE Open Federation/ EEUU	Abierto	Distribuidos	Distribuidos
GEM	Todos	GEM Consortium/ EEUU	Abierto	Distribuidos	Distribuidos
CeLeBraTe	Todos	European Learning Network/ Europa	En desarrollo	Distribuidos	Distribuidos
ELENA/ Edutella	Todos	Edutella/ Europa	Interoperabilidad	Distribuidos	Distribuidos
eduSourceCanada	Todos	EduSource/ Canadá	Interoperabilidad	Distribuidos	Distribuidos

Existen otras iniciativas que aportan opciones y bases para la interoperabilidad de los repositorios de objetos de aprendizaje, entre las cuales están:

- OAI (Open Archives Initiative). Promueve estándares para la interoperbilidad de contenidos por medio de la recuperación automática de metadatos para crear colecciones. Disponible en http://www.oai.org/.
- NSDL (National Science Digital Library). Es un proyecto de la National Science Foundation el cual propone estándares de metadatos, protocolos, esquemas de

[13] Acrónimo que referencia a los niños menores de 12 años (Escuela primaria en los Estados Unidos). http://www.camaraalcoy.net/servicios_web/glosario/Glosario/K.htm

autentificación y modelos para la construcción de bibliotecas digitales. Disponible en http://nsdl.org/.

- OKI (Open Knowledge Initiative). Ofrece una arquitectura abierta y expandible que especifica cómo se comunican los componentes de un ambiente de software educativo entre ellos y con otros sistemas de la organización (ver Apéndice B). Disponible en http://www.okiproject.org/.

V.8. Resumen

En el presente capítulo se habló sobre los objetos de aprendizaje. Se menciona que existen varias definiciones que cada autor ha hecho basándose en ideas previas o en necesidades propias a satisfacer, sin embargo la definición que mas se acerca a los objetivos de este trabajo es la hecha por López (2005) quien cita que "cualquier recurso con una intención formativa, compuesto de uno o varios elementos digitales, descrito con metadatos, que pueda ser utilizado y reutilizado dentro de un entorno de ambiente de aprendizaje puede considerarse un OA (Objeto de Aprendizaje)".

Un objeto de aprendizaje debe contar con ciertas características que le permitan ser considerado como tal, dichas características son mencionadas por Rehak y Mason (2003) quienes citan que un OA debe ser reutilizable para que pueda servir como base o componente de otro recurso; accesible, permitiéndole su localización y recuperación; interoperable, que le ofrece la capacidad de funcionar en diversas plataformas de forma normal; y durable, permitiendo que permanezca intacto a las actualizaciones de hardware y software.

Las definiciones para objeto de aprendizaje coinciden, de cierta manera, en manejar un descriptor para el OA, el cual es llamado metadato. De manera general, los metadatos son un conjunto de atributos o elementos necesarios para descubrir un recurso. A manera de ejemplo, tomemos el caso de una ficha bibliográfica, en ésta se tiene toda la información que describe al libro o algún otro recurso documental, mediante el análisis de la ficha

bibliográfica se decide si se consulta o no el recurso al que pertenece, así también resulta fácil y ágil ubicar dicho recurso.

Los objetos de aprendizaje, así como sus metadatos, deben estar almacenados en algún lugar para poder tener acceso a los mismos, el lugar destinado para su almacenamiento es llamado repositorio de objetos de aprendizaje (ROA). Un ROA es un término muy general que va desde bases de datos hasta catálogos de sistemas, sin embargo puede ser entendido como un almacén de OA que permite organizarlos y manejarlos para otorgar acceso a los mismos.

Estos repositorios existen en dos tipos los cuales son (Rehak, 2003): los que contienen los OA y sus metadatos, y los que sólo contienen los metadatos. En los primeros, los OA y los metadatos se encuentran en un mismo sistema e incluso en un mismo servidor mientras que en el segundo, sólo se encuentran los descriptores de los recursos los cuales proveen una referencia para acceder al recurso ubicado en otro lugar físico.

Los ROA cuentan con ciertas características que otorgan un mejor manejo de los recursos que contiene, entre estas características están las herramientas de búsqueda y recopilación en el repositorio, el meta-etiquetado, administración de contenidos, entre otros.

También se presentaron algunas iniciativas en la creación y desarrollo de ROA, así como propuestas de redes interoperables de repositorios de objetos de aprendizaje.

Capítulo VI

Arquitectura Adoptada de Federación

En capítulos anteriores se han tratado temas diversos referentes al concepto de búsqueda federada así como los componentes necesarios para llevar a cabo dicha actividad; se planteó el concepto de federación, de cual parte el proceso de búsqueda federada; se ha mencionado el término integración que se refiere a la capacidad de comunicar entidades heterogéneas entre si, así como también se mencionaron las herramientas que pueden ayudar al desarrollo de dicha integración, otorgando una estandarización que a su vez permite integrar los componentes mencionados en otros sistemas heterogéneos de igual o mayor tamaño.

En este aspecto, también se hizo mención sobre las fuentes de información, las cuales son las encargadas de proveer los datos que los usuarios de la federación necesitan para efectuar actividades propias, es decir, son las bases de datos a las cuales se envían consultas con la finalidad de recuperar información que sea de utilidad para quien la ha solicitado.

En este capítulo se menciona la arquitectura adoptada de modelo de federación utilizado, que permite la integración de colecciones heterogéneas de metadatos.

VI.1. Escenario de trabajo

Actualmente los repositorios que se desea integrar son los ubicados en el Departamento de Ciencias de la Computación de la División de Física Aplicada del Centro de Investigación Científica y de Educación Superior de Ensenada (CICESE), ubicado en la ciudad de Ensenada, Baja California; y el repositorio localizado en el Instituto de Ingeniería de la Universidad Autónoma de Baja California (UABC) campus Mexicali, ubicado en la ciudad de Mexicali, Baja California.

Estos repositorios cuentan con objetos de aprendizaje y sus respectivos metadatos, referentes a temas relacionados con la enseñanza de las matemáticas a nivel secundaria, sin embargo, para fines experimentales y con el propósito de evaluar de una manera exhaustiva la arquitectura adoptada, se crearon 20 colecciones de metadatos distribuidas físicamente en diversos repositorios donde cada una de las mismas contiene alrededor de 1200 metadatos.

El tipo de repositorios creados es distribuido que, como ya se mencionó, son aquellos en los cuales los metadatos son almacenados en un lugar diferente al que son depositados los OA asociados. La búsqueda federada, en el contexto del presente trabajo, está enfocada a la búsqueda y recuperación de los metadatos, por lo que no es necesaria la creación de los OA para efecto de la medición de rendimiento de la arquitectura adoptada.

VI.2. Colecciones y documentos

En la sección anterior se mencionaron los términos colección y documento, los cuales son utilizados en el proceso de búsqueda y recuperación de metadatos, pero para evitar ambigüedad en las ideas, hay que clarificar los conceptos y definirlos de manera adecuada con el contexto manejado en el presente trabajo.

Powell (2001) define a un documento como un elemento que contiene datos, el cual puede ser recuperado en respuesta a una petición realizada por un usuario, y considera a una

colección como un conjunto o agrupamiento de documentos, los cuales pueden o no tener algún tipo de relación entre ellos.

En términos generales, la definición de colección es adecuada para ser utilizada como un conjunto o agrupamiento de elementos, sin embargo la definición de elemento es la que debe ser modificada.

Puesto que la arquitectura propuesta basa su funcionamiento en la búsqueda de objetos de aprendizaje mediante la ubicación de sus respectivos metadatos, nuestro interés se centra en el manejo de los mismos, por consiguiente, en este trabajo un documento hace referencia a un metadato mientras que cuando hablemos acerca de una colección, estaremos refiriéndonos a una colección de metadatos o un conjunto de los mismos.

Los metadatos fueron creados utilizando el estándar IEEE LOM, el contenido de los mismos está basado en ciertas asignaturas impartidas en las escuelas de nivel secundaria (Matemáticas, Español, Geografía, Física y Química).

En el estándar IEEE LOM existe una sección destinada específicamente para el titulo del objeto de aprendizaje al cual pertenece (*title*) y otra sección utilizada para una descripción del objeto de aprendizaje (*description*); la información contenida en estas secciones está relacionada con las materias mencionadas anteriormente, en otras palabras, el titulo del metadato puede ser un tema relacionado con la materia de matemáticas por ejemplo, mientras que la descripción del mismo será una selección de subtemas referentes al tema del cual pertenecen.

El motivo por el cual se decidió tomar este tipo de materias, fue obedeciendo a la pauta que ya se había planteado en los repositorios originales, es decir, los de CICESE y UABC-Mexicali, puesto que, como ya se mencionó, estos repositorios cuentan con metadatos y objetos de aprendizaje referentes a temas de matemáticas de nivel secundaria.

VI.3. Arquitectura adoptada

Una vez que se han descrito brevemente los componentes que forman las fuentes de información para efectuar el proceso de búsqueda, se procede con la definición formal de la arquitectura adoptada, capaz de integrar las colecciones y los metadatos que conforman las fuentes de información a ser utilizadas.

Las necesidades encontradas que deben ser satisfechas con el propósito de proveer la creación de la federación y de esta manera la integración de los diversos componentes son las mencionadas a continuación: los repositorios se encuentran distribuidos en lugares físicamente dispersos; el acceso a estos repositorios es de sólo lectura; los sistemas de información se encuentran funcionando bajo plataformas diferentes; el tipo de repositorios utilizados son distribuidos (metadatos almacenados en un sitio y OA en otro), además de que los sistemas son autónomos.

La arquitectura de federación que mejor se adapta y satisface las necesidades de integración en el ambiente de trabajo mencionado, es la de una arquitectura de sistemas de información basado en mediadores (MBIS). Las consideraciones bajo las cuales fue adoptada esta arquitectura, se presentan a continuación:

- En dicha arquitectura existe un componente encargado de determinar el funcionamiento general de la federación (mediador), por lo que en caso de existir problemas relacionados al funcionamiento de la federación es posible delimitar el alcance de los mismos, es decir, es posible conocer la causa de la falla y saber exactamente el lugar donde ocurre la misma.
- En caso de que existir otro repositorio que quisiera formar parte de la federación, simplemente es necesaria la creación de una nueva envoltura que implemente los mecanismos de acceso a dicho repositorio; a diferencia de crear todo un nuevo componente que implemente todos los mecanismos de búsqueda, acceso y recuperación necesarios en la federación.

- La autonomía de los repositorios se mantiene dado que ellos son los que deciden la información que dan a conocer y su permanencia o integración con la federación.
- En caso de que el repositorio tenga características similares a otro existente en la federación, una envoltura ya existente puede ser utilizada
- La interoperabilidad con otras federaciones puede ser soportada mediante una comunicación con otros mediadores, solo es necesaria la creación de un mecanismo de comunicación entre los respectivos mediadores así como de una envoltura que permita el intercambio de peticiones entre dichos mediadores.
- La ubicación física de los repositorios presenta un único inconveniente sólo en caso de que no existe algún medio de comunicación con dicho repositorio, en caso contrario la comunicación es soportada con el uso de su respectiva envoltura.

Como se mencionó en el Capítulo IV, los MBIS son sistemas que integran fuentes de información heterogéneas mediante las envolturas (wrappers), las cuales permiten el acceso físico a los datos almacenados en las fuentes de información. La arquitectura basada en mediadores propuesta consta de cuatro capas, llamadas:

- Capa de usuario
- Capa de mediador
- Capa de envoltura
- Capa de repositorio

Cada una de las capas mencionadas cuenta con componentes específicos que ejecutan cierto procesamiento que en conjunto llevan al correcto funcionamiento de la federación (Figura 9).

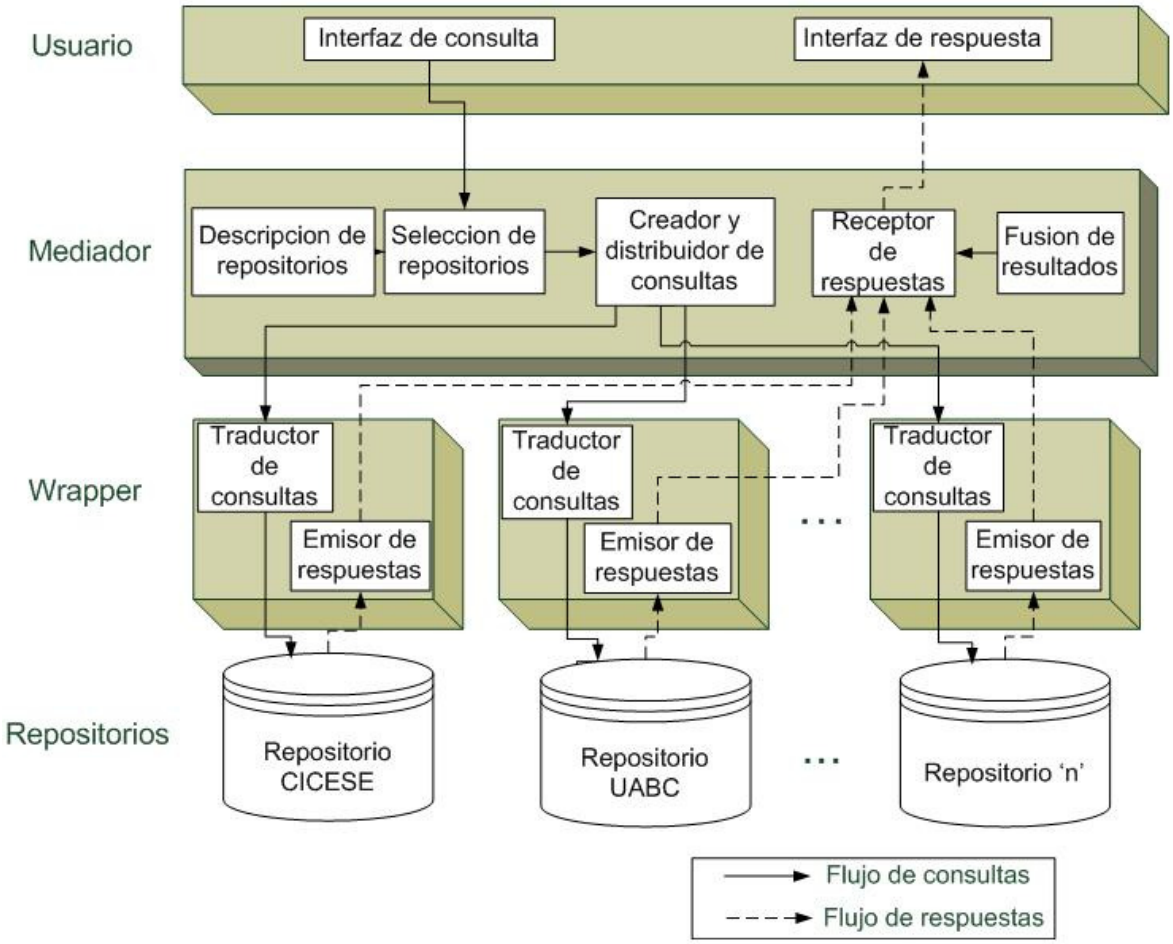

Figura 9. Diagrama a bloques de la arquitectura adoptada.

VI.3.1. Funcionamiento de las capas de la arquitectura

Cada una de las capas que conforman la arquitectura adoptada es especializada en realizar una función cuyo resultado es utilizado por otra capa para realizar su funcionamiento. Las capas de la arquitectura son descritas a continuación:

La capa de usuario es utilizada para poder comunicarse con la federación; esta capa cuenta con dos componentes que fungen como medio de comunicación con la federación. Estos componentes son la interfaz de consulta y la interfaz de respuesta.

El usuario introduce una cadena de búsqueda en la interfaz de consulta con el fin de encontrar resultados que satisfagan la necesidad de información del mismo; los resultados que, de acuerdo al criterio utilizado para seleccionar los metadatos (mencionado más adelante), satisfacen dicha necesidad son mostrados mediante la interfaz de respuesta, es decir, la interfaz de respuesta es el componente encargado de mostrar los resultados obtenidos de los repositorios de información.

En la capa de mediador, de la cual se basa el nombre, es donde se lleva a cabo el mayor procesamiento de cálculos para poder realizar las consultas y recuperación de resultados. Esta capa está compuesta por varios componentes que realizan funciones específicas para el correcto funcionamiento del mediador; dichos componentes son llamados: descripción de repositorios, selección de repositorios, creador y distribuidor de consultas, receptor de respuestas y fusión de resultados, los cuales son descritos en las siguientes líneas.

El componente llamado descripción de los repositorios, es donde se encuentra alojada información que cada repositorio da a conocer y que está dado de alta en el sistema; mediante este componente se puede saber con cuántos repositorios se cuentan para poder discriminar de entre los mismos y así determinar las características sobre la información que cada uno de ellos contiene.

Otro componente de la capa de mediador es el de selección de repositorios, el cual se encarga de determinar hacia cuáles repositorios sería conveniente enviar la consulta del usuario para poder recuperar mejores y relevantes resultados de la misma, utiliza información del componente que contiene la descripción de repositorios, con el fin de determinar los repositorios a cuales enviar la consulta, , en otras palabras, este componente es el encargado de realizar el proceso de discriminación de repositorios. El proceso de discriminación de repositorios es llevado a cabo mediante un algoritmo de selección de colecciones, del cual se hablará en una sección posterior.

En lo que respecta al componente llamado creador y distribuidor de consultas, éste es el encargado de establecer comunicación con las correspondientes envolturas para poder enviar la consulta del usuario. La consulta debe estar expresada en un lenguaje común de la federación, el cual entiendan todas las envolturas; en otras palabras, este componente es el encargado de transformar la consulta del usuario, expresada en lenguaje natural, a un lenguaje común de la federación.

En cuanto al componente receptor de respuestas, éste es el encargado de recibir todos los mensajes de respuesta de las envolturas, expresadas en el lenguaje común de la federación; dicho componente se auxilia del componente fusión de resultados para poder completar el trabajo que tiene asignado.

El componente fusión de resultados se encarga de combinar los mismos y crear una lista ordenada de los resultados arrojados por la consulta realizada, para posteriormente enviar esa consulta a la interfaz de respuesta con el propósito de presentarla al usuario.

En la capa de envoltura existen tantas envolturas como repositorios haya dados de alta en la federación, es decir, existirá una envoltura por cada repositorio. Aunque este hecho es el que se presenta de manera común en los MBIS cabe la posibilidad de que existan menos envolturas que repositorios, dado que una envoltura puede ser utilizada por un conjunto de repositorios con características similares, como la estructura de datos o lenguaje de consulta, permitiendo la reutilización de envolturas.

La envoltura es la encargada de permitir la comunicación entre el repositorio y la capa de mediador con el propósito de ejecutar las consultas realizadas por el usuario y obtener resultados que satisfagan la petición realizada. Cada envoltura está compuesta por un componente traductor de consultas y un componente emisor de respuestas, los cuales son descritos a continuación.

El componente traductor de consultas es el encargado de recibir la consulta enviada por el emisor y distribuidor de consultas localizado en la capa de mediador, dicha consulta viene expresada en un lenguaje común de la federación, es decir, el traductor de consultas se encarga de convertir esa consulta en una consulta expresada en el lenguaje local del repositorio.

Posteriormente la consulta es ejecutada en el repositorio y los resultados obtenidos son manipulados por el componente emisor de respuestas, quien es el encargado de traducir las respuestas expresadas en lenguaje local del repositorio, al lenguaje común de la federación; después de que las respuestas son expresadas en el lenguaje común de la federación, éstas son enviadas al componente receptor de respuestas para que pueda realizar las operaciones necesarias.

Finalmente, en cuanto a la capa de repositorios, ésta simplemente consta de todos los repositorios que forman parte de la federación, y de los cuales es tomada la información que reside en cada uno de ellos.

VI.3.2. Solución a los problemas de la búsqueda federada

Cada una de las capas que conforman la arquitectura propuesta, presenta detalles en su desarrollo que son de especial interés el hacer mención de ellos, puesto que gracias a dichos detalles es posible el funcionamiento de las capas de la arquitectura.

Como ya se mencionó, el núcleo o corazón del sistema, por llamarlo de algún modo, se ubica en la capa de mediador así como en la capa de envoltura, ya que gracias a las envolturas es posible el acceso a los metadatos contenidos en los repositorios.

VI.3.2.1. Problema de selección de colecciones

En el Capítulo III, dimos pauta a la definición de un sistema federado, sin embargo, el proceso de la búsqueda en este tipo de sistemas puede ser comprendido como búsqueda federada, la cual involucra una solución a tres problemas los cuales son: la selección de las colecciones, la recuperación de los resultados y la combinación de los mismos; estos problemas son soportados en la capa de mediador (selección de colecciones y combinación de resultados) y en la capa de envoltorio (recuperación de resultados).

Respecto a la selección de colecciones, este problema es atacado al utilizar un algoritmo de selección de colecciones, donde este algoritmo puede ser implementado en base al modelo de recuperación de información que se desee.

Modelos de recuperación

Existen ciertos modelos de recuperación de información, los cuales, de acuerdo con Fernández (2001), pueden ser definidos de manera informal como una especificación de la representación de los documentos y de las consultas, más la forma en que se compararán para recuperar los documentos relevantes.

Estos modelos de recuperación son conocidos como: el modelo booleano, modelo de espacio vectorial y modelo probabilístico; a continuación daremos una muy breve descripción de los mismos, profundizando un poco más en el modelo probabilístico puesto que este modelo ha sido utilizado para dar solución al problema de selección de colecciones.

El modelo booleano está basado en la teoría de conjuntos y en el álgebra booleana. Este modelo está compuesto por los documentos representados como conjuntos, las consultas, como expresiones booleanas (términos conectados por los conectivos booleanos AND, OR y NOT), y las operaciones existentes para tratar conjuntos: unión, intersección y complemento.

Los pesos de los términos en los documentos son binarios (0 y 1). Así, en este modelo, dada una consulta al sistema, se va evaluando la expresión booleana mediante la realización de las operaciones anteriores con los conjuntos formados por los documentos donde aparece cada término de la consulta. El conjunto de documentos resultante está compuesto por todos aquellos que hacen verdad la consulta booleana. La principal ventaja de este modelo es la simplicidad del mismo.

En el modelo de espacio vectorial se representa a cada documento de texto como un conjunto de términos, éstos términos son palabras extraídas de los propios documentos (Greengrass, 2000). Posteriormente se le asigna un peso a cada término, con el propósito de poder identificar a cada documento en base a los pesos asignados a sus términos. Cada documento es así identificado en todo el espacio de la colección, este documento puede ser conocido como un vector, y es mediante este último que se puede acceder a los documentos.

Por último, el modelo probabilístico está compuesto por conjuntos de variables, operaciones con probabilidades y el teorema de Bayes. Según Fernández (2001), todos los modelos probabilísticos están basados en el principio de la ordenación por probabilidad (en inglés: *the probability ranking principle*). Dicho principio fue formulado por Robertson (1997), quien asegura que el rendimiento óptimo de la recuperación se consigue ordenando los documentos según sus probabilidades de ser juzgados relevantes con respecto a una consulta, siendo estas probabilidades calculadas de la forma mas precisa posible a partir de la información disponible. Así, partiendo de este principio, el objetivo de cualquier modelo probabilístico es calcular la probabilidad de relevancia dados una consulta y un documento.

Dentro de los modelos probabilísticos se encuentran los algoritmos de selección de colección (ver Apéndice A), de los cuales, la investigación realizada (Craswell, 2000; Powell, 2001; Si, 2006; Frenchm *et al.*, 1999) concuerda en que el algoritmo con mejor desempeño (comparado con otros modelos de recuperación probabilísticos) en la recuperación de información relevante, es el algoritmo CORI .

El algoritmo CORI (*Collection Retrieval Inference Network*) (Si y Callan, 2003; Callan, 2000; Callan *et al.*, 1995) representa a las colecciones como elementos virtuales, de manera que pueda ser obtenida información sobre los mismos, por ejemplo, los términos que aparecen y la frecuencia de aparición de los mismos. CORI crea un índice de estos elementos virtuales representados por los términos y sus frecuencias de aparición en los documentos de las colecciones originales.

En otras palabras, dado un conjunto de colecciones disponibles para su búsqueda, CORI crea un índice de las colecciones, donde cada una de éstas es representada por sus términos y las frecuencias de los mismos; el índice creado se ordena en base a una consulta, por lo que cada consulta generará un índice ordenado en base a la anterior. La probabilidad de que un término r_k exista en una colección C_i, es decir $p(r_k|C_i)$, es calculada en base a las siguientes fórmulas:

$$T = \frac{df}{df + 50 + 150 \cdot cw / \overline{cw}} \tag{1}$$

$$I = \frac{\log\left(\frac{N + 0.5}{cf}\right)}{\log(N + 1.0)} \tag{2}$$

$$p(r_k|C_i) = 0.4 + 0.6 \cdot T \cdot I \tag{3}$$

Donde:

df es el número de documentos en la colección C_i que contienen el término r_k

cf es el número de colecciones que contienen a r_k

N es el número de colecciones totales a ordenar

cw es el número de palabras en C_i

$\overline{cw}$ es la media del número de palabras en las colecciones

Las constantes encontradas en la definición de T, son explicadas por un estudio paramétrico reportado en Callan et al., (1995). En dicho estudio, T fue originalmente definido como:

$$T = d_t + (1 - d_t) \cdot \frac{df}{df + K} \tag{4}$$

Donde $K = k \cdot ((1-b) + b \cdot cw/\overline{cw})$, k y b con constantes y d_t se refiere a la frecuencia de un término en un documento. El estudio realizado por Callan *et al.* (1995) encontró que para un ambiente de pruebas de 17 colecciones, la mejor combinación de valores fue k = 200, b = 0.75. Al utilizar estas constantes, la definición original de T resulta en las constantes encontradas en la ecuación (1)., así como también, dicho estudio fundamenta la decisión de tomar un valor de 0.4 como *default*, originando las constantes de la ecuación (3), la cual es definida de manera diferente en la investigación de Callan *et al.* (1995), y que, simplificándola, genera la ecuación final.

El tratar a las colecciones como documentos virtuales, es simplemente una variación la forma general de recuperación de documentos $tf \cdot idf$ (Ramos, 1995; Powell, 2001; Callan *et al.,* 1995). Esta representación es formada a partir de dos estadísticas, la frecuencia del término (tf) y la frecuencia inversa de documento (idf). La frecuencia del término es una cuenta del número de ocurrencias de un término t_j en un documento d_i. La frecuencia inversa de documento está definida como:

$$idf = \log\left(\frac{N}{df}\right) \tag{5}$$

Donde N es el número de documentos en la colección, y df es el número de esos documentos que contienen el término t_j. El objetivo de combinar estas dos estadísticas es asignar valores altos a los términos que ocurren frecuentemente en un documento pero infrecuentemente en la colección como un todo. Tales términos son considerados como los mejores representantes del contenido de un documento.

Esta técnica de filtrado (*idf*) está basada en la ley de Zipf (Robertson, 2004), que establece que las palabras con mayor frecuencia absoluta son las palabras vacías, mientras que las más infrecuentes son aquéllas que reflejan el estilo y riqueza del vocabulario del autor.

La ecuación (2) se basa en la definición inicial de *idf* , sin embargo, utiliza otros componentes con el propósito de normalizar los resultados haciendo que oscilen en el rango de [0,1] (Callan *et al.*, 1995; Callan, 2000; Craswell, 2000).

En la representación del algoritmo CORI, *df* es utilizado como referencia para el término *tf*. En las fórmulas del algoritmo CORI, la literal *T* es usada para representar el componente *df* y la literal *I* representa al componente *idf*, por lo que la formula final de CORI tiene la forma $T \cdot I$.

Una vez que se ha creado el índice de las colecciones ordenadas en base a una consulta dada, es posible discriminar de entre todo el conjunto de colecciones con el propósito de elegir a aquellas que satisfagan de mejor manera a la consulta inicial, permitiendo con esto, un menor tiempo de procesamiento así como mejores resultados.

El problema de selección de colecciones está implementado en el componente selección de repositorios, es decir, el componente consta de la implementación del algoritmo CORI, pero, como ya se mencionó, CORI necesita información adicional sobre las colecciones que se pretenden indexar, dicha información es provista por el componente descripción de repositorios.

El funcionamiento de los dos componentes produce el índice final de los repositorios en base a una consulta dada, solucionando así el primer problema de la búsqueda federada: la selección de colección.

VI.3.2.2. Problema de recuperación de resultados

Una vez que se han seleccionado los repositorios para realizar la consulta, ésta origina que cada repositorio responda con los resultados que la satisfacen; dichos resultados son los que deben ser presentados al usuario final, sin embargo, el proceso de recuperación de resultados no es tan sencillo como aparenta.

El presente trabajo esta enfocado a la búsqueda federada de los metadatos de objetos de aprendizaje y por consiguiente los resultados de las consultas en los repositorios, serán metadatos. En otras palabras, el problema de recuperación puede ser conocido como recuperación de metadatos.

Para tal efecto existen varios protocolos encargados de la recuperación de metadatos, entre los cuales están: Z39.50 (ANSI/NISO-Z39.50-2003, 2003) y el OAI-PMH (OAI, 2004) (ver Apéndice A); sin embargo el protocolo que mejor se adapta a la arquitectura propuesta, es el OAI-PMH.

El Protocolo para Recolección de Metadatos de la Iniciativa de Archivos Abiertos (OAI-PMH, por sus siglas en inglés), proporciona interoperabilidad entre las fuentes de documentos, en este caso, entre los repositorios de metadatos (véase Apéndice B). La implementación del protocolo está realizada en la capa de envoltura, puesto que la función de la misma, como ya se mencionó, es permitir la comunicación con la fuente física de datos logrando así la recuperación de los datos que residen en ella.

De manera específica, la implementación del protocolo se encuentra en los componentes traductor de consultas y emisor de respuestas, donde se lleva a cabo el proceso de petición de resultados y entrega de los mismos con la ayuda de dicho protocolo. El OAI-PMH utiliza el lenguaje XML para poder comunicarse entre el cliente (emisor de la consulta) y el proveedor de servicios (repositorio de metadatos), además de basar su funcionamiento en los métodos GET y POST de HTTP.

Otro punto importante es el formato de los metadatos; los metadatos deben estar creados bajo el estándar de metadatos Dublin Core (DCMI, 2008), dado que es el estándar soportado por el protocolo, además de que permite la interoperabilidad de los mismos y resulta relativamente más fácil de aplicar y comprender, en comparación con otros estándares (IEEE LOM [IEEE/LTSC, 2002], por ejemplo), debido a la cantidad de información necesaria para su llenado. Cabe destacar que el protocolo sólo permite la recuperación de los metadatos, no se encarga de la búsqueda de los mismos.

VI.3.2.3. Problema de combinación de resultados

Después de que se han recolectado los resultados de los distintos repositorios seleccionados, dichos resultados deben ser presentados al usuario como una sola lista ordenada, es decir, la selección de las colecciones y la recuperación de los resultados debe ser transparente para el usuario, dando la idea de que se esta comunicando con una sola colección en lugar de varias.

Para poder presentar esta lista ordenada, los resultados deben ser normalizados con el fin de contar con un criterio común para su ordenamiento dado que los mismos provienen de varias colecciones heterogéneas.

Powell (2001) realiza la combinación de resultados mediante la normalización, primero, de las colecciones y posteriormente la normalización de los documentos; dicho proceso es el utilizado por el sistema de recuperación *Inquery* (Callan *et al.*, 1992). En este sistema, la normalización de los puntajes es calculada mediante una combinación entre el puntaje de la colección y el puntaje del documento, es decir, el puntaje normalizado de la colección es calculado mediante la siguiente fórmula:

$$C' = (C - C_{\min}) / (C_{\max} - C_{\min}) \tag{6}$$

Donde:

C es el puntaje original de la colección

$C_{\max}$ es la colección con mayor puntaje respecto a una consulta en particular de todo el conjunto de colecciones

$C_{\min}$ es la colección con menor puntaje respecto a una consulta en particular de todo el conjunto de colecciones

El puntaje C de la colección, puede ser calculado en base a un análisis de los datos generales de dicha colección, como la cantidad de documentos que contiene y el enfoque de dichos documentos, es decir, el tema del que tratan y palabras claves de los mismos. La normalización de los puntajes de las colecciones es necesario para el cálculo del proceso similar respecto a los documentos, por lo que, el puntaje normalizado D'' de un documento con un puntaje inicial D, será calculado conforme a las siguientes fórmulas:

$$D' = (D - D_{\min}) / (D_{\max} - D_{\min}) \quad (7)$$

$$D'' = (1.0 \cdot D' + 0.4 \cdot C' \cdot D') / 1.4 \quad (8)$$

Donde:

$D_{\min}$ es el documento con menor puntaje respecto a una consulta en particular en su respectiva colección

$D_{\max}$ es el documento con mayor puntaje respecto a una consulta en particular en su respectiva colección

D' es el puntaje normalizado del documento, en el marco de la colección a la que pertenece.

De manera similar al cálculo de los puntajes de las colecciones, el puntaje de cada documento contenido en cada colección es calculado en base a la cantidad de palabras clave que lo identifiquen y de información sobre la colección en la que se encuentren almacenados.

Las constantes presentadas en la ecuación (8), fueron definidas en el estudio realizado por Callan *et al.* (1995) y, al igual que en las constantes de las ecuaciones (1) a (3), fueron definidas con el propósito de normalizar los resultados al utilizar la formula general de recuperación de información $tf \cdot idf$.

Con la información obtenida de la aplicación del algoritmo CORI y de las fórmulas de normalización, es posible la creación de una sola lista ordenada que contenga los resultados ordenados de todas las colecciones que satisfagan de mejor manera a la consulta realizada, resolviendo de esta manera el problema de la combinación de resultados.

VI.4. Resumen

En el presente capitulo se explicó de manera específica la funcionalidad de la arquitectura basada en mediadores propuesta, se presentó información detallada referente a cada una de las capas de la misma. A su vez, se recordó que el proceso de búsqueda federada involucra tres problemas: la selección de las colecciones, la recuperación de los resultados y la combinación de los mismos en una sola lista ordenada.

Para el problema de selección de colecciones, la solución que mejor se adapta y que, de acuerdo a la investigación previa, funciona de una manera mejor en comparación con soluciones similares, es el algoritmo CORI; este algoritmo calcula la probabilidad de que una consulta obtenga mejores resultados que la satisfagan y en base a esta probabilidad se define el orden de búsqueda y se discrimina de entre las colecciones disponibles. El algoritmo es implementado en la capa de mediador.

El problema de la recuperación de los resultados, es atacado con la implementación del protocolo para la recuperación de metadatos de la OAI, es decir, el OAI-PMH. Este protocolo basa su funcionamiento en los métodos GET y POST de HTTP; dicho protocolo es implementado en la capa de envoltura, es decir, en cada envoltura de cada repositorio.

En cuanto al problema de combinación de resultados, la solución al mismo es la utilización de un puntaje normalizado, con el fin de contar con un criterio de comparación y presentar al usuario la lista ordenada de los resultados obtenidos de todos los repositorios seleccionados. Esta lista ordenada es creada a partir de normalizar los puntajes de las colecciones y posteriormente, con estos puntajes normalizados, realizar el mismo procedimiento a los puntajes de los documentos.

Con las soluciones presentadas, se garantiza que el proceso de búsqueda puede ser llevado a cabo recuperando resultados que mejor se adapten a la consulta realizada por el usuario final.

Capítulo VII

Evaluación de la Arquitectura Adoptada

La arquitectura de federación basada en mediadores propuesta debe ser sometida a pruebas para corroborar que las decisiones tomadas sobre la implementación de los mecanismos que permiten la solución a los problemas que implica la búsqueda federada sean las correctas.

En este capítulo se presenta información referente a la evaluación de la arquitectura adoptada; se hace mención de las métricas utilizadas para la misma, así como del proceso de evaluación utilizado en el cual se define el criterio de relevancia elegido para tal fin.

Una vez definido el criterio para determinar si un documento es relevante o no; este criterio es utilizado con el propósito de poder, de alguna manera, contar con un mecanismo que permite medir la satisfacción del usuario a las consultas realizadas.

VII.1. Medidas de evaluación

El funcionamiento de la arquitectura debe ser evaluado conforme a ciertas medidas que permitan el análisis claro y preciso del mismo. Por lo que la selección del tipo de medidas a utilizar dependerá de lo que se desee evaluar.

La literatura no considera alguna métrica de evaluación para el concepto de búsqueda federada como tal (Frenchm *et al.*, 1999; Powell, 2001; Manning *et al.*, 2008; Greengrass,

2000), en vez de ello se consideran dos términos utilizados para medir el proceso de recuperación de documentos relevantes, los cuales son: Exhaustividad y Precisión (en inglés: *Recall* y *Precision*, respectivamente). Estas medidas son definidas de la siguiente manera:

$$Exhaustividad = \frac{NDRR}{NDR} \tag{9}$$

$$\Pr ecisión = \frac{NDRR}{ND\mathrm{Re}} \tag{10}$$

Donde:

$NDRR$ es el número de documentos relevantes recuperados,

NDR es el número de documentos relevantes,

$ND\mathrm{Re}$ es el número de documentos recuperados.

La precisión es obtener la proporción de material recuperado realmente relevante, del total de los documentos recuperados; mientras que la exhaustividad es la proporción de material relevante recuperado, del total de los documentos que son relevantes en la colección, independientemente de que éstos sean recuperados o no.

De acuerdo con Fernández (2001), el problema fundamental que poseen estas dos medidas es que la precisión se calcula de manera exacta mientras que la exhaustividad no, ya que no se tiene un conocimiento claro de cuántos documentos relevantes existen en un colección para una consulta dada; sin embargo, para el presente trabajo sí se cuenta con esta información por lo que es posible un mejor cálculo de las medidas mencionadas.

VII.1.1. Criterio de relevancia y consultas

En la sección anterior se mencionaron las métricas utilizadas para medir el proceso de recuperación de documentos relevantes, sin embargo existe un problema que hay que resolver antes de poder aplicar dichas fórmulas, el cual es la definición de relevancia.

De acuerdo con Gómez Díaz (2003), el concepto de relevancia se ha estudiado de diversos enfoques, los cuales se pueden resumir en dos tendencias: la relevancia objetiva y la relevancia subjetiva. La primera hace hincapié en cómo es que la materia de la información recuperada coincide con la de la pregunta o consulta realizada. La subjetiva, es la que tiene en cuenta al usuario; es decir, la relevancia mirada desde el punto de vista del usuario.

Para Schamber *et al.* (1990), la relevancia se refiere a la utilidad o al uso potencial de los materiales recuperados con relación a la satisfacción de los objetivos, el interés, el trabajo o los problemas intrínsecos del usuario.

En el contexto de este trabajo y para fines prácticos, hemos definido el criterio de relevancia utilizado para catalogar a los documentos de las diferentes colecciones utilizadas. Como se mencionó en el Capítulo VI, la descripción de los documentos esta formada por subtemas referentes a temas de ciertas materias de nivel secundaria, estos subtemas son repetidos cierta cantidad de veces por lo que tomamos como documento relevante a aquel en el que la consulta aparezca una mayor cantidad de veces, es decir, mientras mas veces aparezca la cadena de búsqueda (consulta) en un documento, más relevante será el mismo respecto a dicha consulta.

En los experimentos realizados, la estrategia de generación de consultas produjo un conjunto de aproximadamente 300 cadenas de consulta divididas en 2 conjuntos: cadenas largas y cadenas cortas.

Las cadenas cortas fueron construidas en base a palabras clave que caracterizasen a un subtema en especifico donde su longitud no exceda las 2 palabras, por ejemplo: supongamos que un documento es llamado “Características de los números naturales”, las palabras que caracterizan de mejor manera a este documento son: “números naturales”, por lo que esta cadena es agrupada en el conjunto de cadenas cortas.

Por otro lado, la consulta: “características de números naturales” entraría en el conjunto de cadenas largas, en donde las cadenas que pertenezcan a este conjunto tienen como característica que su longitud está entre las 3 y 6 palabras clave que mejor describan al documento en cuestión.

La determinación de qué cadena entra en cuál conjunto se fundamenta en el conocimiento básico sobre las materias y, de cierta manera, también por sentido común; ya que, retomando el ejemplo, la cadena corta “características”, sería demasiado ambigua como para considerarla en alguno de los conjuntos de consultas.

VII.2. Evaluación de la arquitectura

Como ya se ha mencionado, el proceso de evaluación lo enfocamos a la recuperación de documentos relevantes, por lo que a continuación se presenta un análisis de las consultas ejecutadas en el proceso de recuperación y su interpretación de acuerdo a la cantidad de documentos recuperados relevantes, documentos relevantes y documentos totales en cada colección.

La metodología de evaluación consistió en la ejecución de diversas consultas, cada una de éstas construida a partir de los temas utilizados en la construcción de los metadatos; el conjunto de consultas fue dividido en consultas largas y cortas, donde cada una recuperaba resultados diversos.

VII.2.1. Tamaño de la muestra

Al ejecutar cada consulta, la cantidad de metadatos recuperados consiste de 1024, sin embargo dependiendo de la consulta realizada, el número de metadatos relevantes variará, por lo que los valores de precisión y exhaustividad serán diferentes en cada consulta realizada.

La cantidad de metadatos a recuperar fue tomada en base a conceptos sobre teoría de muestreo, la cual menciona que para calcular el tamaño de una muestra se debe tomar en cuenta tres factores (Larios-Osorio, 1999):

1. El porcentaje de confianza
2. El porcentaje de error
3. El nivel de variabilidad

La confianza o el porcentaje de confianza es el porcentaje de seguridad que existe para generalizar los resultados obtenidos. Esto quiere decir que un porcentaje del 100% equivale a decir que no existe ninguna duda para generalizar tales resultados, pero también implica estudiar a la totalidad de los casos de la población. Comúnmente en las investigaciones se busca un 95% de confianza, por lo que en este trabajo se utilizará dicho porcentaje. El nivel de confianza se obtiene a partir de la distribución normal estándar, pues la proporción correspondiente al porcentaje de confianza es el área simétrica bajo la curva normal que se toma como la confianza, y la intención es buscar el valor Z de la variable aleatoria que corresponda a tal área.

El porcentaje de error equivale a elegir una probabilidad de aceptar una hipótesis que sea falsa como si fuera verdadera, o la inversa: rechazar la hipótesis verdadera por considerarla falsa. Al igual que en el caso de la confianza, si se quiere eliminar el riesgo del error y considerarlo como 0%, entonces la muestra es del mismo tamaño que la población, por lo que conviene correr un cierto riesgo de equivocarse. Comúnmente se aceptan entre el 4% y el 6% como error.

La variabilidad es la probabilidad (o porcentaje) con el que se aceptó y se rechazó la hipótesis que se quiere analizar en alguna investigación anterior o en un ensayo previo a la investigación actual. El porcentaje con que se aceptó tal hipótesis se denomina variabilidad positiva y se denota por p, y el porcentaje con el que se rechazó la hipótesis es la variabilidad negativa, denotada por q.

Se considera que p y q son complementarios, es decir, que su suma es igual a la unidad: $p+q=1$. Además, cuando se habla de la máxima variabilidad, en el caso de no existir antecedentes sobre la investigación (no hay otras o no se pudo aplicar una prueba previa), entonces los valores de variabilidad son $p=q=0.5$.

La ecuación que permite obtener el tamaño ideal de la muestra, en base al tamaño de la población total, es la siguiente:

$$n = \frac{Z^2 pqN}{NE^2 + Z^2 pq} \tag{11}$$

Donde

n es el tamaño de la muestra

Z es el nivel de confianza

p es la variabilidad positiva

q es la variabilidad negativa

N es el tamaño de la población

E es la precisión o el error

Para el trabajo presentado se toma una confianza de 95%, entonces hay que considerar la proporción correspondiente, que es 0.95. Lo que se busca en seguida es el valor Z para la variable aleatoria z tal que el área simétrica bajo la curva normal desde $-Z$ hasta Z sea igual a 0.95, es decir, $P(-Z<z<Z)=0.95$. Entonces la distribución normal estándar de Z es 1.96. Sustituyendo los respectivos valores en la ecuación 11, el tamaño de la muestra es de 1024; por lo que el tamaño de la muestra utilizada para este trabajo es de 1024.

VII.2.2. Metodología de evaluación

Para la evaluación de la arquitectura adoptada es necesario someterla a la realización de consultas de metadatos, con el propósito de que cada consulta regrese metadatos que sean

relevantes a la consulta realizada; esta afirmación puede ser tomada como nuestra hipótesis nula.

La información que debe estar contenida en los metadatos con el fin de ser descriptiva respecto a algún tema, fue tomada a partir de temas referentes a materias impartidas en el nivel secundaria; dichos temas forman un conjunto de 43 elementos referentes a las materias de Matemáticas, Física, Química, Español y Geografía. A su vez, cada uno de dichos temas está subdivido en un conjunto de subtemas. La Tabla IV presenta la cantidad de elementos de los conjuntos de temas y subtemas, así como la materia a la cual están asociados.

Del conjunto que engloba los diferentes subtemas mencionados anteriormente, se creó un nuevo conjunto formado por subcadenas provenientes de cada uno de los subtemas, es decir, del conjunto de subtemas se creó el conjunto de consultas las cuales fueron ejecutadas en la implementación de la arquitectura adoptada con la finalidad de recuperar documentos relevantes a cada una de ellas.

Dicho conjunto de consultas está compuesto por dos tipos de consultas, dependiendo de la longitud de la cadena creada. Si una cadena de consulta es creada con una longitud mayor o igual a cuatro palabras, entonces ésta es considerada en el subconjunto de consultas largas; en caso de que la longitud de la misma sea menor o igual a tres palabras, entonces es considerada como una consulta corta. El conjunto de consultas totales a realizar es de 383, divididas en 213 consultas cortas y 170 consultas largas. La cantidad de consultas largas y cortas para cada tema es mostrada en la Tabla V.

Tabla IV. Cantidad de los elementos que componen los conjuntos de temas y subtemas para la creación de los metadatos.

Materia	Tema	Subtemas
Química	0	3
Química	1	4
Química	2	3
Química	3	4
Química	4	3
Química	5	6
Química	6	4
Química	7	4
Química	8	5
Química	9	4
Matemáticas	10	10
Matemáticas	11	10
Matemáticas	12	9
Matemáticas	13	7
Matemáticas	14	5
Matemáticas	15	5
Matemáticas	16	7
Matemáticas	17	7
Matemáticas	18	7
Matemáticas	19	6
Geografía	20	4
Geografía	21	3
Geografía	22	3
Geografía	23	3
Geografía	24	3
Geografía	25	3
Geografía	26	3
Geografía	27	10
Geografía	28	4
Geografía	29	7
Física	30	5
Física	31	4
Física	32	5
Física	33	6
Física	34	4
Física	35	5
Física	36	7
Física	37	4
Física	38	7
Física	39	7
Español	40	3
Español	41	3
Español	42	6

Tabla V. Cantidad de consultas largas y cortas para cada uno de los temas utilizados en la creación de los metadatos, así como el número que identifica a cada consulta asociado a cada uno de los temas.

Tema	Consultas totales	Consultas cortas	Consultas largas	Identificador de consulta
0	10	7	3	1, 2, 3, 4, 5, 6, 7 ,8 9, 10
1	4	4	0	11, 12, 13, 14
2	9	6	3	15, 16, 17, 18, 19, 20, 21, 22, 23
3	7	4	3	24, 25, 26, 27, 28, 29, 30
4	6	4	2	31, 32, 33, 34, 35, 36
5	6	6	0	37, 38, 39, 40, 41, 42
6	10	8	2	43, 44, 45, 46, 47, 48, 49, 50, 51, 52
7	9	6	3	53, 54, 55, 56, 57, 58, 59, 60, 61
8	13	7	6	62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74
9	11	6	5	75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85
10	15	6	9	86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99, 100
11	15	4	11	101,102,103,104,105,106,107,108,109,110,111,112,113,114,115
12	14	6	8	116,117,118,119,120,121,122,123,124,125,126,127,128,129
13	13	5	8	130,131,132,133,134,135,136, 137,138,139,140,141,142
14	15	8	7	143,144,145,146,147,148,149,150,151,152,153,154,155,156,157
15	5	0	5	158, 159, 160, 161, 162
16	11	6	5	163, 164, 165, 166, 167, 168, 169, 170, 171, 172, 173
17	14	6	8	174,175,176,177,178,179,180,181,182,183,184,185,186,187
18	9	1	8	188, 189, 190, 191, 192, 193, 194, 195, 196
19	12	5	7	197, 198, 199, 200, 201,202,203, 204, 205, 206, 207,208
20	4	4	0	209, 210, 211, 212
21	5	4	1	213, 214, 215, 216, 217
22	6	4	2	218, 219, 220, 221, 222, 223
23	4	1	3	224, 225, 226, 227
24	7	4	3	228, 229, 230, 231, 232, 233, 234
25	6	4	2	235, 236, 237, 238, 239, 240
26	2	1	1	241, 242
27	10	10	0	243, 244, 245, 246, 247, 248, 249, 250, 251, 252
28	9	5	4	253, 254, 255, 256, 257, 258, 259, 260, 261
29	14	9	5	262,263, 264, 265, 266, 267, 268, 269,270,271,272,273,274, 275
30	6	6	0	276, 277, 278, 279, 280, 281
31	7	3	4	282, 283, 284, 285, 286, 287, 288
32	11	7	4	289, 290, 291, 292, 293, 294, 295, 296, 297, 298, 299
33	11	6	5	300, 301, 302, 303, 304, 305, 306, 307, 308, 309, 310
34	6	2	4	311, 312, 313, 314, 315, 316
35	5	4	1	317, 318, 319, 320, 321
36	12	6	6	322, 323, 324, 325, 326, 327, 328, 329, 330, 331, 332, 333
37	6	4	2	334, 335, 336, 337, 338, 339
38	8	4	4	340, 341, 342, 343, 344, 345, 346, 347
39	12	6	6	348, 349, 350, 351, 352, 353, 354, 355, 356, 357, 358, 359
40	6	4	2	360, 361, 362, 363, 364, 365
41	5	3	2	366, 367, 368, 369, 370
42	13	7	6	371, 372, 373, 374, 375, 376, 377, 378, 379, 380, 381, 382, 383

VII.3. Implementación de la arquitectura

Una vez que se ha definido la manera en la cual fueron construidos los metadatos y el conjunto de consultas a realizar, es necesaria la mención de los repositorios donde serán almacenados los metadatos, así como la cantidad de los mismos que contendrá cada uno de dichos repositorios.

Diversas investigaciones realizadas utilizan una cantidad de repositorios y de documentos (metadatos, en nuestro caso) diferentes entre ellas, por ejemplo los experimentos realizados por Chernov (2005) involucran una cantidad de 50 colecciones almacenando aproximadamente 500 documentos en cada una de ellas; así mismo en la arquitectura propuesta por Saavedra (2003) se integran tres bases de datos que contienen información referente a libros y documentos de literatura española pero no menciona la cantidad exacta de éstos que es almacenada en cada base de datos, lo que hace suponer que dicha cantidad puede estar entre los cientos y miles de documentos; el trabajo realizado por Si y Callan (2003) maneja 20 bases de datos con alrededor de 300 documentos almacenados en cada una de ellas; a su vez Powell (2001) toma en consideración 9 colecciones con alrededor de 3 mil documentos en cada colección.

En base a las investigaciones mencionadas en el párrafo anterior, la cantidad de documentos empleados en la totalidad de las colecciones utilizadas oscila entre los 6 mil y 27 mil aproximadamente distribuidos en los diferentes repositorios cuya cantidad también es variable (entre 3 y 50 repositorios). Esto nos hace pensar que la cantidad tanto de metadatos como de repositorios no esta definida por algún patrón en especifico pero, considerando los datos mencionados, la cantidad de colecciones utilizadas en la implementación de la arquitectura adoptada en el presente trabajo es de 20 colecciones conteniendo entre mil y mil 500 metadatos cada una, lo que significa que la cantidad total de metadatos creados es de 25 mil aproximadamente.

Cabe resaltar que en las investigaciones mencionadas, no consideran el número de repositorios empleados sino solamente la cantidad de colecciones utilizadas; por lo que debido a restricciones en la cantidad de repositorios requeridos, las colecciones creadas serán distribuidas en 4 repositorios de prueba. Como punto importante, dichas investigaciones consideran que todos sus documentos son diferentes, por lo que en nuestra investigación utilizamos el mismo criterio.

Respecto a la ubicación de los metadatos creados, éstos son repartidos de forma aleatoria entre las colecciones creadas de manera que cada colección contiene metadatos referentes a los 42 temas utilizados para su creación, con la característica de que ningún metadato es duplicado. Esta propiedad es definida en base a un identificador que lo describe de manera única en todo el conjunto de metadatos. La Tabla IX ubicada en Apéndice C muestra la cantidad de metadatos contenidos en cada colección, así como el tema utilizado para la creación de cada metadato.

VII.4. Resultados obtenidos

Como ya se mencionó, la evaluación de la arquitectura adoptada consiste en la ejecución de 383 consultas que incluyen tanto consultas largas como consultas cortas. Para efectos de comparación de resultados y dado que el análisis del conjunto completo de consultas sería muy extenso y redundante, tomaremos la consulta número 10 la cual pertenece al conjunto de consultas largas.

La ejecución de esta consulta resultó en la recuperación de 1024 metadatos de los cuales 350 de ellos son relevantes al tema asociado a dicha consulta, es decir al tema 0. De la totalidad de los metadatos creados distribuidos en los 4 repositorios, 578 son relevantes al tema 0 en mayor o menor grado (de acuerdo al criterio de relevancia ya definido).

Como ya se mencionó, la consulta seleccionada para su análisis es la número 10 perteneciente al tema 0 llamado “Importancia de la química para el ser humano y el ambiente”; dicha consulta fue tomada de manera aleatoria por lo que, en caso de seleccionar alguna otra, el análisis de los resultados obtenidos por dicha consulta pueden ser discutidos de manera similar a la número 10.

La Tabla VI presenta los temas existentes y la cantidad de documentos relevantes que pertenecen a cada uno de dichos temas. El conjunto resultado relevante (el conjunto de 350 metadatos relevantes), está integrado en el conjunto resultado total (los 1024 metadatos recuperados); este último conjunto resultado, es presentado en la Tabla VII, y en la cual se pueden apreciar las medidas de precisión y exhaustividad para cada metadato.

Sin embargo, dado que el conjunto resultado es demasiado grande para ser mostrado en su totalidad, solo se presenta un fragmento del mismo y en el cual aparecen algunos de los documentos relevantes, así como su respectiva posición en el proceso de ordenamiento.

Así mismo, le representación gráfica de los datos presentados en la tabla mencionada anteriormente, es mostrada en la Figura 10, en donde se puede apreciar cómo varía la precisión y también la exhaustividad a medida que se avanza en la lista ordenada de metadatos recuperados. El eje ‘x’ representa el número de metadatos evaluados y las curvas muestran el comportamiento de las medidas.

Tabla VI. Nombres de temas y cantidad de documentos relevantes pertenecientes a su respectivo tema.

Tema	Nombre	Documentos relevantes
0	Importancia de la quimica para el ser humano y el ambiente	578
1	Fenomenos quimicos cotidianos	635
2	Mediciones de materia	624
3	Coloides y suspensiones	627
4	Disoluciones acuosas y su concentracion	587
5	Metodos de separacion de mezclas	602
6	Los atomos y las moleculas	566
7	Pesos atomicos de los elementos	596
8	La tabla periodica	610
9	Acidez y basicidad	599
10	Los numeros naturales y sus operaciones	554
11	Los decimales y sus operaciones	598
12	Fracciones	585
13	Prealgebra	577
14	Dibujo y trazos geometricos	644
15	Plano cartesiano y funciones	634
16	Operaciones con expresiones algebraicas	572
17	Ecuaciones y sistemas de ecuaciones lineales	588
18	Circulo	564
19	Simetria	578
20	Grandes regiones fisiograficas del pais	586
21	Las aguas oceanicas	585
22	Las aguas continentales	556
23	Los climas de Mexico	607
24	Las regiones naturales de Mexico	582
25	La poblacion de Mexico	593
26	Poblacion rural y urbana	554
27	Las actividades economicas en Mexico	583
28	El planeta Tierra en el Sistema Solar	610
29	La Tierra, nuestro planeta	577
30	Magnitudes fundamentales de la fisica	574
31	La medida	575
32	Sistema Internacional de Unidades	589
33	Instrumentos de medida y medicion	531
34	Leyes de Newton	569
35	Las maquinas simples	583
36	Medicion de la temperatura	572
37	Los materiales y su conductividad electrica	612
38	Corriente electrica	611
39	El sonido y su propagacion	578
40	Recopilacion y redaccion de textos	588
41	El predicado en la oracion simple	588
42	Los elementos del predicado	594

Tabla VII. Detalles de la consulta número 10 que pertenece al tema 0.

Posición	¿Relevante?	Orden	Exhaustividad	Precisión
1	Si	1	0.002857143	1
2	Si	2	0.005714286	1
3	Si	3	0.008571429	1
4	Si	4	0.011428571	1
5	Si	5	0.014285714	1
6	Si	6	0.017142857	1
7	Si	7	0.02	1
8	Si	8	0.022857143	1
...	…	…	…	…
35	Si	35	0.1	1
36... 39	No	35	0.1	0.972222222
40	Si	36	0.102857143	0.9
41	Si	37	0.105714286	0.902439024
…	…	…	…	…
49	Si	45	0.128571429	0.918367347
50, 51	No	45	0.128571429	0.9
…	…	…	…	…
69	Si	51	0.145714286	0.739130435
70	Si	52	0.148571429	0.742857143
…	…	…	…	…
400	Si	290	0.828571429	0.725
…	…	…	…	…
462	No	350	1	0.757575758
…	…	…	…	…
1024	No	350	1	0.341796875

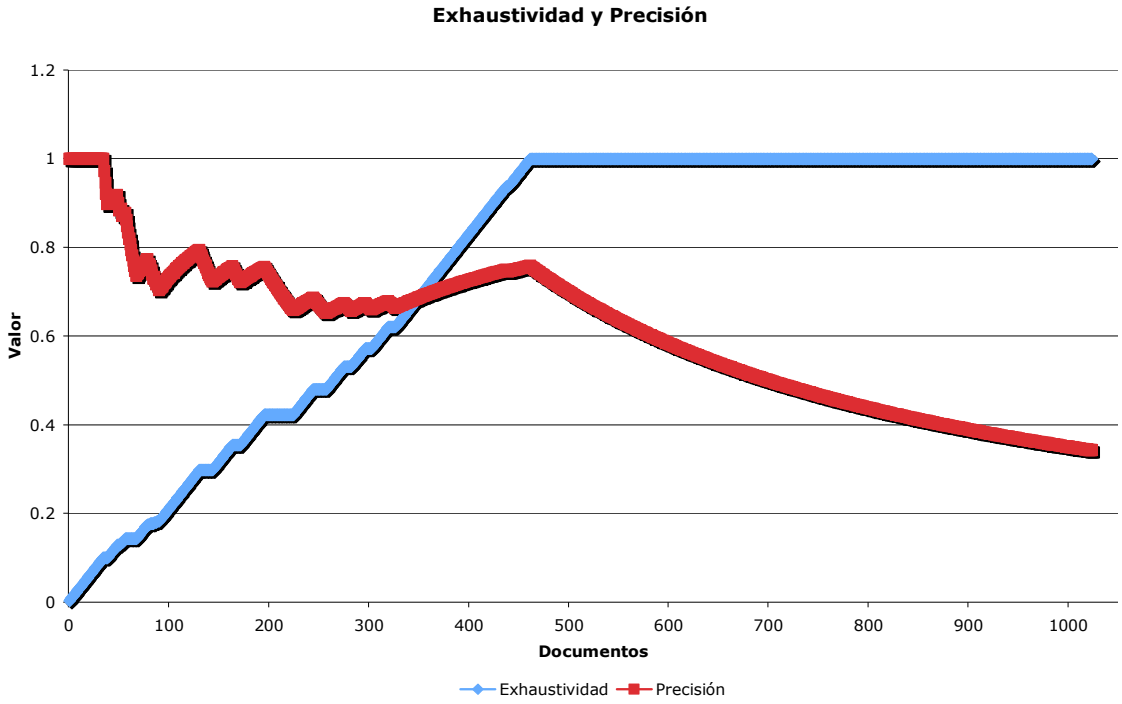

Figura 10. Evolución de Exhaustividad y Precisión para la consulta número 10.

La curva decreciente equivale al valor que obtiene la precisión conforme la cantidad de documentos recuperados aumenta, en otras palabras, la salida obtenida en la recuperación es ordenada en función de la relevancia, por lo que los documentos más relevantes están al comienzo de la salida, de esta manera a medida que avanzamos en el número de documentos recuperados, la precisión decae.

Al igual que la precisión, el valor de la exhaustividad oscila en el rango [0,1], lo que significa que si dicho valor es igual a 1, tendremos la exhaustividad máxima y por consiguiente, habremos encontrado todo lo relevante que había en la colección, dicho de otra forma, la recuperación es perfecta. En la Figura 10, la exhaustividad es representada por la curva creciente, en la que se puede apreciar que a medida que aumenta el número de documentos recuperados, la exhaustividad va en aumento.

Existen dos objeciones que se basan en la precisión y en la exhaustividad (Martínez-Méndez y Rodríguez-Muñoz, 2004; Gómez-Díaz, 2003). La primera es que mientras la precisión se puede determinar, la exhaustividad no, ya que para calcularla necesitamos previamente el número de documentos relevantes. El segundo de los puntos es que la exhaustividad y la precisión son igualmente significativas para los usuarios; mientras que unos prefieren una precisión mayor, otros prefieren una exhaustividad más alta, y en lo que los autores concuerdan es que esto es imposible de lograr.

Martínez-Méndez y Rodríguez-Muñoz (2004) y Gómez-Díaz (2003), mencionan que una gran mayoría de usuarios consideran mucho más importante la precisión, relegando generalmente a la exhaustividad a un cometido secundario; mientras la búsqueda proporcione información relevante, el usuario no suele detenerse a pensar en la cantidad de documentos relevantes que no recupera.

Las dos medidas así como el análisis de cada una de ellas, se pueden utilizar para cualquier consulta realizada, obviamente, los resultados obtenidos variarían dependiendo de la consulta pero los resultados son parecidos y por consiguiente, el análisis de los mismos puede ser el ya utilizado independientemente de la consulta realizada.

El análisis anterior fue hecho en base a la consulta número 10 en la cual los documentos recuperados corresponden al tema 0 cuyos valores se encuentran en la Tabla VIII. La representación gráfica de las medidas de precisión y exhaustividad correspondientes a cada una de las consultas del tema en cuestión es presentada en la Figura 11, en la cual se puede apreciar que las ya mencionadas medidas, presentan un comportamiento similar, a diferente escala.

El comportamiento mencionado es debido a que el sistema propuesto está más enfocado a la precisión que a la exhaustividad, lo que concuerda con las ideas mencionadas por Martínez-Méndez y Rodríguez-Muñoz (2004) y Gómez-Díaz (2003), por lo que la recuperación de metadatos que concuerden con la cadena de búsqueda estará enfocada a la

mayor recuperación de metadatos que se adapten de mejor manera a la consulta realizada; dicho de otra manera, la arquitectura de federación adoptada, está enfocada a la recuperación de una mayor cantidad de documentos relevantes, basándose en el criterio de relevancia definido anteriormente.

Tabla VIII. Valores referentes a la Precisión y Exhaustividad así como la cantidad de metadatos recuperados para las consultas referentes al tema 0.

Consulta	Metadatos recuperados	Metadatos relevantes	Precisión	Exhaustividad
1	1024	578	1	0.564453125
2	1024	103	0.178200692	0.100585938
3	1024	23	0.039792388	0.022460938
4	1024	24	0.041522491	0.0234375
5	1024	68	0.117647059	0.06640625
6	1024	16	0.027681661	0.015625
7	1024	124	0.214532872	0.12109375
8	1024	308	0.532871972	0.30078125
9	1024	291	0.503460208	0.284179688
10	1024	350	0.605536332	0.341796875

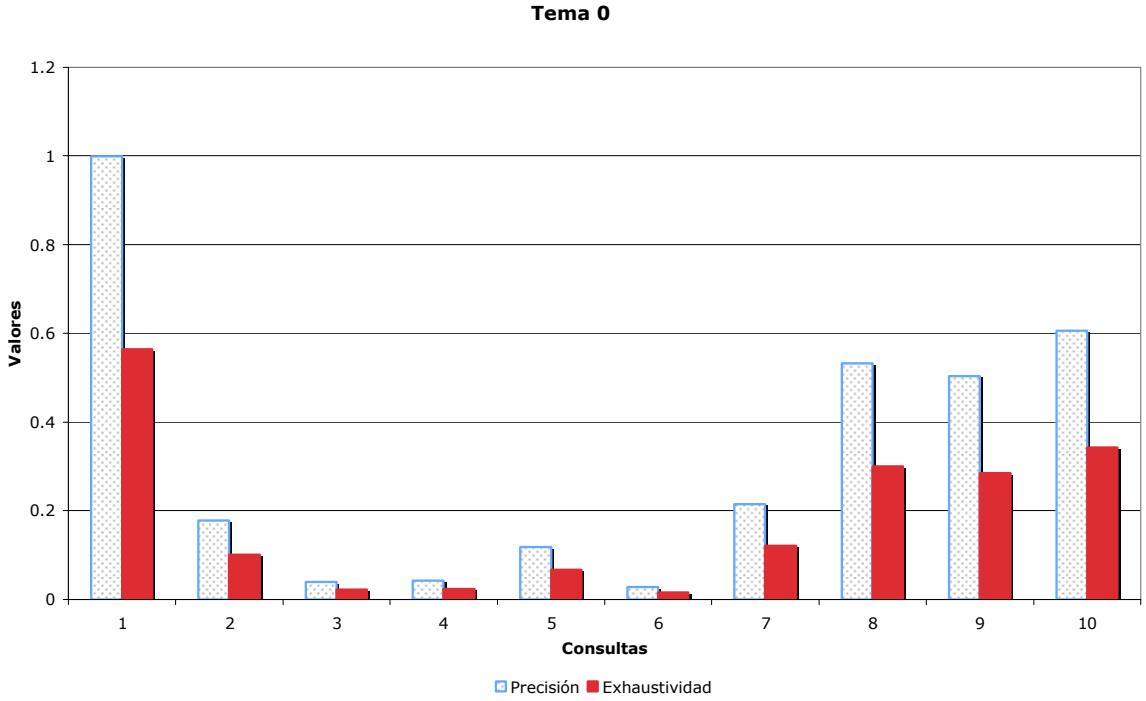

Figura 11. Medidas de Precisión y Exhaustividad para cada una de las 10 consultas referentes al tema 0.

En la Figura 11 se puede apreciar que las barras correspondientes a los valores de precisión corresponden a la recuperación de los metadatos que concuerdan con la consulta realizada. En contraste con las demás consultas realizadas, los resultados obtenidos para cada una de ellas serían similares, por ejemplo, una consulta, ya sea corta o larga, que corresponda al tema número 18 recuperará cierta cantidad de resultados, los cuales serán interpretados de manera similar a la consulta mostrada en la Figura 11, dichos resultados pueden satisfacer la necesidad de información que tiene el usuario.

En algunas consultas, como por ejemplo, la número 6, presenta valores cercanos a 0, esto es debido a que la cadena de búsqueda fue creada de una manera que no satisficiera los criterios de relevancia adoptados. Una explicación mayor es presentada más adelante en esta sección.

Las medidas de Precisión y Exhaustividad correspondientes a las consultas ejecutadas en la arquitectura adoptada son mostrados en la Tabla X ubicada en el Apéndice C. Dichos resultados muestran que para cada consulta realizada, el valor de Precisión es mayor que el de la Exhaustividad corroborando con esto que los métodos y procedimientos empleados en la arquitectura adoptada están enfocados a la precisión de la recuperación de documentos relevantes a la cadena de búsqueda.

En otras palabras, todas las consultas realizadas resultan en una cierta cantidad o en la totalidad de metadatos relevantes; cada conjunto resultado es diferente dependiendo de la consulta a la cual satisface y dado que la cantidad de consultas es extensa, solo un fragmento de las mismas es presentado en el Apéndice C.

En cada una de las gráficas presentadas en dicho Apéndice, se pueden apreciar los correspondientes valores de Precisión y Exhaustividad, resaltando que los valores pertenecientes a la precisión son aquellos en los que en todas las gráficas aparecen con valores cercanos a 1, en comparación con los valores pertenecientes a la exhaustividad; esto comprueba lo que ya se ha mencionado sobre el enfoque de los métodos y procedimientos

empleados en la arquitectura adoptada, el cual es la precisión de la recuperación de metadatos relevantes. Todas las gráficas creadas, así como sus respectivos valores pueden ser consultados en el CD anexo a este trabajo de tesis.

Hemos estado haciendo mención de las consultas creadas, sin hacer notar que pertenezcan al conjunto de consultas largas o al conjunto de cadenas cortas. Powell (2001) menciona que el tamaño de la cadena de búsqueda puede influir en la recuperación de resultados que satisfagan a la misma, es decir que mientras más grande sea dicha consulta se supone que puede ser más especializada y por lo tanto recuperar resultados más relevantes a dicha consulta.

Los resultados obtenidos por Powell (2001) demuestran que el tamaño de la cadena de búsqueda sí influye en la recuperación de resultados más relevantes; sin embargo, el criterio de relevancia utilizado en dicha investigación involucra otros criterios para considerar que un documento es relevante, como por ejemplo: el tema del cual tratan los documentos (dado que no todas las colecciones empleadas contienen información similar) y el año de creación y el autor de los mismos.

En nuestro caso, como el criterio de relevancia está relacionado con la cantidad de veces que aparece la cadena de búsqueda en la descripción del metadato, la mayor cantidad de metadatos relevantes es recuperada mediante la utilización de consultas cortas. Lo anterior es debido a que mientras más grande sea la longitud de la consulta realizada implica que debe existir esa cadena en la descripción de los metadatos, por lo que en caso de que la cadena no exista en dicha descripción entonces ése metadato no es recuperado como resultado relevante.

Otro punto importante a considerar en la creación de las consultas, tanto largas como cortas, es el mecanismo o criterio utilizado para dicha creación. Callan *et al.* (1992), Chernov (2005), Craswell (2000) y Frenchm *et al.* (1999) mencionan que la creación de las consultas realizadas en sus respectivas investigaciones, fueron hechas con la ayuda de

expertos en cada respectiva área; es decir, las consultas creadas fueron redactadas en base a criterios tomados de personas consideradas profesionales en los temas incluidos en los documentos de las colecciones empleadas.

Para este trabajo, la redacción de las consultas creadas fue hecha en base a conocimiento previo sobre el tema y, también, por sentido común. Debido a esto la creación de las consultas no fue realizada por expertos en la respectiva materia por lo que dicha creación pudo tener ciertas fallas, originando que las consultas creadas no son las que recuperen la mayor cantidad de metadatos relevantes. Esta idea aplica tanto para las consultas cortas como para las consultas largas. Este análisis explica que algunas consultas creadas regresaran valores cercanos a 0.

Aunque la creación de las consultas juega un rol importante en la recuperación de documentos relevantes, en este trabajo todas las consultas realizadas se enfocaron en la precisión de la recuperación, independientemente de que fueran consultas cortas o largas; sin embargo, la creación de las consultas influyó en la cantidad de metadatos recuperados, dado que las consultas largas recuperaron menos metadatos relevantes que las consultas cortas. Estos valores pueden ser consultados en la Tabla X ubicada en el Apéndice C.

En la Figura 12 se pueden apreciar las medidas de Precisión y Exhaustividad para las cadenas cortas referentes al tema 0, mientras que en la Figura 13 se muestran las mismas medidas pero orientadas a las cadenas largas para el mismo tema. Las totales creadas para el tema 0 fueron 10, de las cuales 7 son cortas (Figura 12) y 3 son consultas largas (Figura 13). Como se puede apreciar, tanto en las consultas cortas como en las largas, se recuperan metadatos relevantes; la diferencia radicaría en la cantidad de éstos que es recuperada (ver Tabla X).

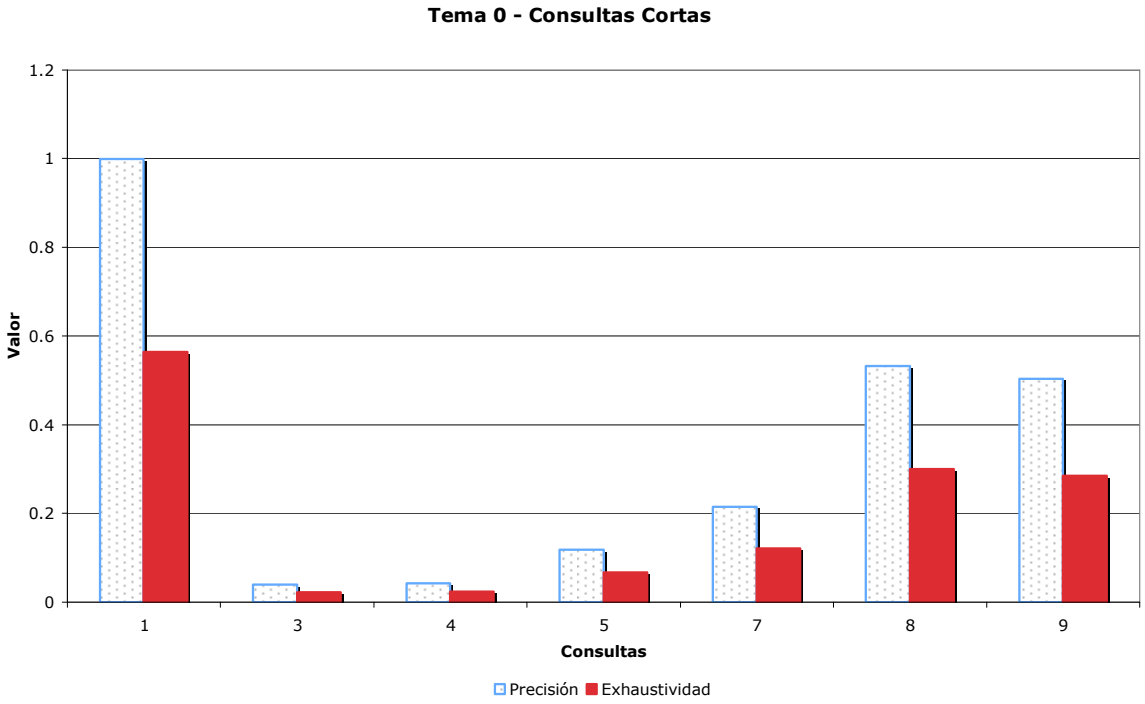

Figura 12. Consultas cortas realizadas para el tema 0.

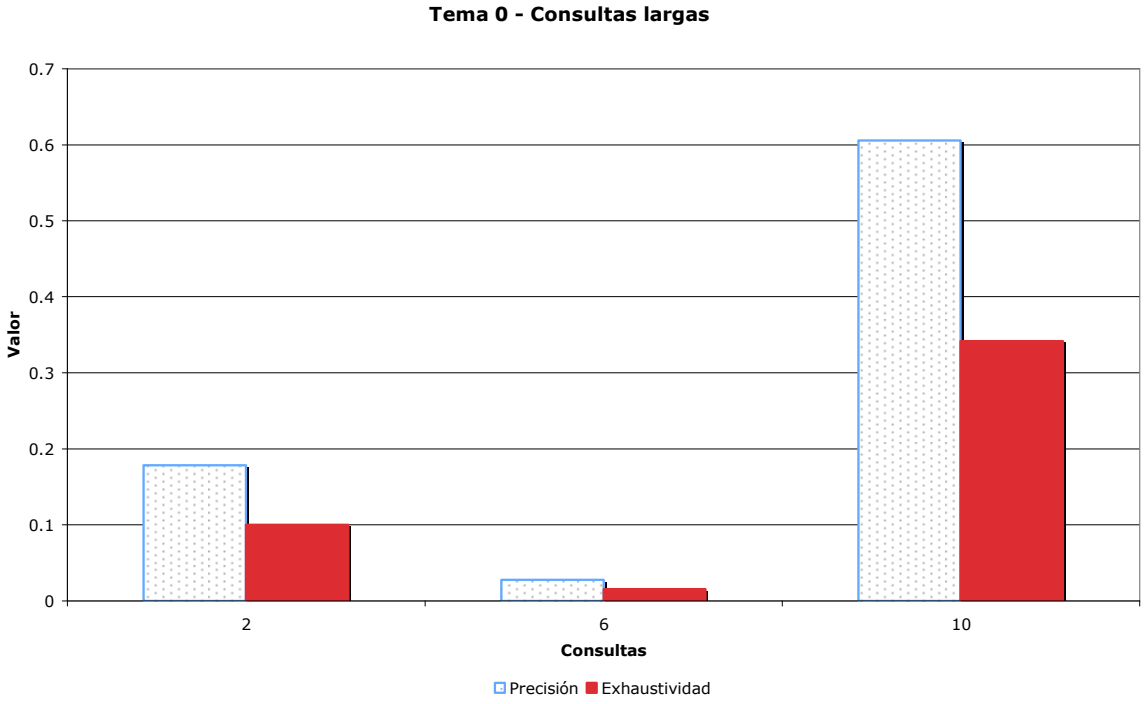

Figura 13. Consultas largas realizadas para el tema 0.

La cantidad de consultas largas y cortas varia dependiendo del tema que se trate, es decir, para un tema puede existir una mayor cantidad de consultas largas debido a que una cadena corta puede ser muy "genérica" para recuperar metadatos que satisfagan a la respectiva consulta. La cantidad de metadatos recuperados, varía en cada consulta realizada, obviamente por el tipo de información que se está solicitando y por la manera en la que se han creado las consultas.

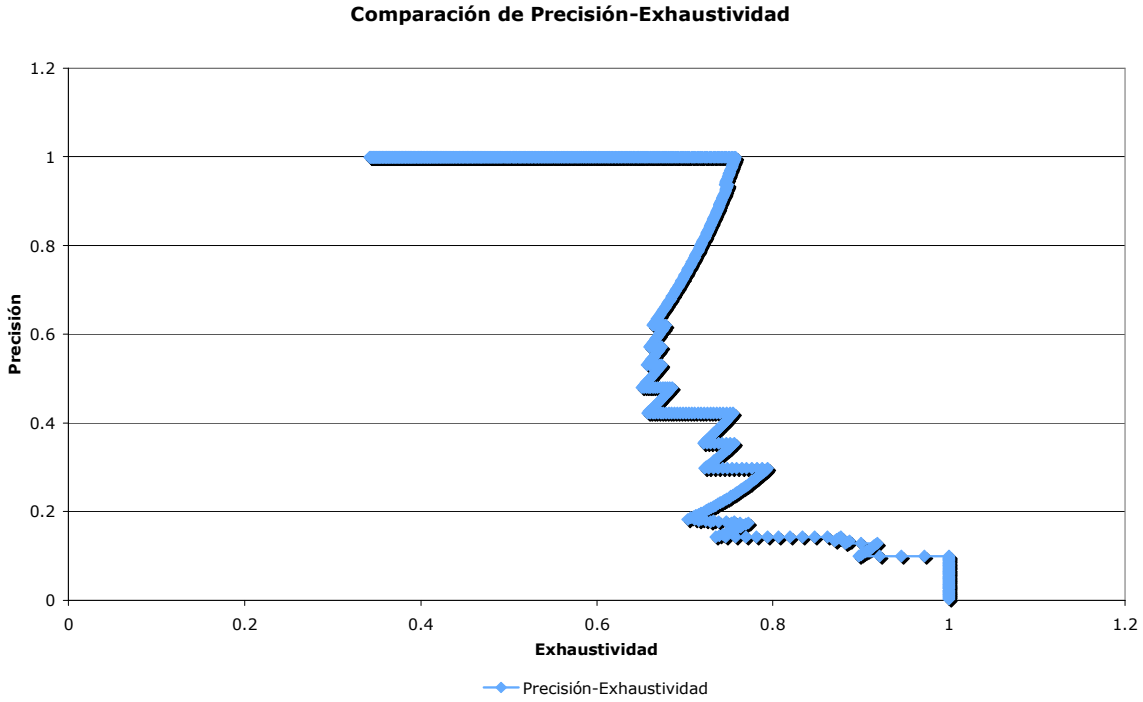

Figura 14. Comparación de las medidas de Precisión y Exhaustividad para el tema 0.

La manera en la cual se relacionan las medidas de Precisión y Exhaustividad para el tema 0 puede ser apreciada en la Figura 14, en donde en el eje 'x' es puesto el rango de valores que puede alcanzar la Exhaustividad (el cual es [0,1]), mientras que en el eje 'y' aparecen los valores que puede tomar la Precisión. Dicha figura es una representación gráfica de los datos presentados en la Tabla VII, pero, como ya se mencionó, la cantidad de datos es muy extensa para presentarlos en su totalidad, por lo que pueden ser consultados en el CD anexo.

De la Figura 14 se puede afirmar que los métodos para recuperación de metadatos implementados en la arquitectura propuesta, siempre alcanzan una mayor precisión al recuperar los metadatos entre los primeros lugares de la lista de respuesta; esto se debe a que la exhaustividad máxima es alcanzada antes de tener que revisar toda la respuesta. Cabe señalar que esta ideología puede ser aplicada para cada una de las consultas realizadas, y por lo tanto los resultados serán similares para cada una de ellas.

Debido a que el conjunto de consultas ejecutadas es de cierta manera grande, un análisis de cada una de ellas seria demasiado extenso y redundante respecto a las conclusiones obtenidas sobre los mismos, esto debido a que, como ya se mencionó, los resultados en cada consulta son similares. Por lo anterior es que sólo algunas consultas son seleccionadas y sus respectivas gráficas son creadas y ubicadas en el Apéndice C, las cuales permiten corroborar lo ya mencionado referente al enfoque de Precisión.

Las medidas de Precisión y Exhaustividad son ampliamente usadas en el rubro de recuperación de información, sin embargo algunos autores (Martínez-Méndez y Rodríguez-Muñoz, 2004; Frakes y Baeza-Yates, 1992) concuerdan en ciertos aspectos:

Para poder determinar la Exhaustividad máxima para una consulta se requiere conocer completamente la colección, a tal grado de poder discernir los documentos relevantes de los que no lo son. Por otro lado, la Precisión se puede calcular de manera exacta mientras que la Exhaustividad no siempre.

Estas medidas capturan aspectos diferentes del conjunto de respuesta y, en algunos casos, resulta más útil una medida única. Se puede decir que la Exhaustividad y la Precisión se encuentran relacionadas de tal manera que si se les analiza por separado muestran una vista incompleta de la efectividad del sistema evaluado. Estas medidas requieren del procesamiento por lotes de un conjunto de consultas, lo que no resultaría útil en sistemas que sean de cierta manera interactivos, es decir, aquellos cuyo contenido sea cambiante constantemente.

Ahora bien, las medidas utilizadas nos permiten comprobar que tanto el algoritmo de selección de colección (CORI) y los algoritmos de normalización de resultados, funcionan de manera que recuperan resultados realmente relevantes para la consulta realizada, en base al criterio de relevancia adoptado, posicionándolos en los primeros lugares de la lista ordenada de resultados.

Por otro lado, la arquitectura de federación adoptada basada en mediadores presenta un funcionamiento óptimo en base a que tanto los algoritmos empleados, las cadenas de consulta creadas y los repositorios de prueba implementados, resultaron en la recuperación de una gran cantidad de documentos relevantes almacenados en sus respectivos repositorios, considerando el criterio de relevancia adoptado.

Respecto a la integración de las fuentes de información, la misma pudo ser soportada con la utilización del protocolo de recuperación de metadatos OAI-PMH, debido a la sencillez tanto de los mecanismos de comunicación como de la implementación del mismo, tomando en cuenta las especificaciones necesitadas por dicho protocolo, como por ejemplo el estándar de metadatos necesario (DC).

VII.5. Resumen

En el presente capítulo se trataron temas referentes a la evaluación de los mecanismos de búsqueda y recuperación de metadatos de la arquitectura propuesta, los cuales se enfocan en las medidas denominadas Exhaustividad y Precisión (Recall y Precision, en inglés respectivamente).

Estas medidas se fundamentan en un concepto denominado relevancia, el cual es de carácter subjetivo ya que, dependiendo de la necesidad de información que se tenga, los resultados obtenidos pueden saciar a la misma de manera distinta; dicho de otra forma, quien es el encargado de determinar qué tan relevante o no es un documento recuperado, es

el usuario final, debido a que éste es quien tuvo la necesidad de información inicial y quien esta solicitando se satisfaga la misma.

En base a lo anterior, y debido a que lo que se desea obtener es la evaluación de los mecanismos implementados desde el punto de vista tecnológico, el concepto de relevancia es reducido a la cantidad de ocurrencias de las palabras que se consideren clave, es decir, mientras más veces aparezca un término en el documento, mas relevante será respecto a dicho término.

La recuperación de documentos fue llevada a cabo en base a la ejecución de las consultas largas y cortas, donde cada tipo de consultas recupera resultados diferentes. La definición de cómo formar estas consultas fue hecha en base a conocimiento general sobre los temas tratados así como por sentido común.

Una vez que se han ejecutado el conjunto de consultas, los resultados que éstas arrojaron fueron analizados, permitiendo determinar el tipo de recuperación que los métodos implementados en la arquitectura realizan. Con los resultados obtenidos pudimos constatar que dichos métodos están enfocados en la precisión, es decir, al ejecutar una consulta, el conjunto de respuestas contendrá una gran cantidad de documentos relevantes (ver Tabla X).

Como la cantidad de temas es grande, sólo es presentado el análisis de uno de ellos (el tema 0), sin embrago, el análisis sobre el mismo puede ser aplicado a cualquiera de los temas restantes debido a que los resultados que arrojaron las demás consultas, presentan características similares.

Capítulo VIII

Conclusiones

La búsqueda federada tiene como finalidad una recuperación de resultados contenidos en múltiples fuentes de información heterogéneas, autónomas y distribuidas de manera que los resultados sean presentados al usuario final como si hubieran sido localizados en un solo sistema; dicho de otra manera, incluye una selección de las fuentes de información, realizar la consulta en cada una de las fuentes seleccionadas y combinar los resultados obtenidos para presentarlos al usuario en una sola lista ordenada.

Existen arquitecturas de federación que pueden ser utilizadas para dar soporte al proceso de búsqueda federada; sin embargo las características que la arquitectura basada en mediadores presenta, permiten satisfacer las necesidades de integración expuestas en este trabajo, por lo que dicha arquitectura es la adoptada en el mismo. Dichas características involucran un acceso de solo lectura a los sistemas de información, una localización distribuida de los sistemas de información así como autonomía y heterogeneidad de los mismos. La arquitectura adoptada consta de cuatro capas las cuales son: capa de usuario, capa de mediador, capa de envoltura y capa de repositorio. El conjunto de estas capas da solución a los tres problemas expuestos de la búsqueda federada.

La capa de usuario funge como el intermediario entre el funcionamiento general de la arquitectura y el usuario final; es utilizada para poder comunicarse con la federación. En la capa de mediador se da solución a dos de los problemas de la búsqueda federada; dichos

problemas son la búsqueda de recursos y la combinación de resultados. El primero de ellos es resuelto mediante la implementación de mecanismos de selección de fuentes de información, en específico un algoritmo de selección de colecciones. El problema de combinación de resultados es resuelto mediante la implementación de mecanismos de normalización de puntajes de los documentos recuperados (metadatos). La capa de envoltura permite la comunicación con la fuente física de información, en nuestro caso, la comunicación con los repositorios por lo que es en esta capa donde se da solución al problema de recuperación de recursos.

Los mecanismos implementados en las capas de la arquitectura adoptada están basados en estándares definidos por diferentes organismos y organizaciones, esto con el propósito de permitir la interoperabilidad entre diversas fuentes de información y posiblemente otras federaciones, permitiendo una mayor integración de componentes y que los elementos que conforman la arquitectura puedan ser implementados en diversos sistemas.

En las pruebas realizadas, la ejecución de cada una de las consultas creadas en base a los temas definidos, produjo diferentes resultados, en cuanto a cantidad se refiere; sin embargo, la calidad de los resultados recuperados fue significativa en base a que en todas las consultas existieron resultados relevantes, estos valores pueden ser consultados en la Tabla X del Apéndice C. El funcionamiento de la arquitectura se enfoca en recuperar la mayor cantidad de documentos relevantes ordenados en base a cierto criterio de relevancia, donde los recursos recuperados más relevantes sean posicionados en los primeros lugares de la lista presentada al usuario final.

El proceso de recuperación fue realizado con un enfoque de simulación, lo que permitió en primera instancia conocer cómo se comportarían los mecanismos implementados al utilizar metadatos y colecciones reales; de esta manera podemos conocer una aproximación a la calidad de documentos relevantes que pueden ser recuperados por los mecanismos ya mencionados.

El análisis de los resultados obtenidos por la evaluación de los elementos que en su conjunto conforman la arquitectura propuesta, arrojaron certidumbre para aseverar que los procesos, algoritmos y tecnologías empleadas (algoritmos CORI y de normalización, OAI-PMH, IEEE-LOM) puedan ser implementados en sistemas de búsqueda federada.

El mecanismo utilizado para una implementación de federación es la utilización de una arquitectura basada en mediadores; los metadatos deben estar basados en el estándar de metadatos IEEE-LOM para la descripción de los mismos alojados en los repositorios, este estándar debe ser convertido al estándar Dublin Core ya que este último es utilizado por el protocolo OAI-PMH. Dicho protocolo permite la consulta y recuperación de los metadatos asociados a los respectivos objetos de aprendizaje, toda vez que los repositorios soporten dicho protocolo.

La selección de los metadatos relevantes debe estar basada en la utilización del algoritmo de selección de colección CORI y de normalización de resultados. Además de que las investigaciones realizadas por otros autores fundamentan la utilización de CORI, los resultados obtenidos (ver Tabla X) mediante la utilización de este algoritmo combinado con los algoritmos de normalización de puntajes, demuestran que la selección de las colecciones fue óptima en base a que todos las consultas ubicaron a los metadatos realmente relevantes para cada consulta en los primeros lugares de la lista ordenada de resultados.

La cantidad de metadatos recuperados se pudo ver afectada por los criterios tomados tanto en la relevancia de documentos como en la creación de las consultas. Si el criterio utilizado para determinar la relevancia de un documento se apoyara de otras variables (por ejemplo: año de creación, autor, número de páginas) y la creación de las consultas es hecha por expertos en la materia, la cantidad de resultados relevantes recuperados sería mayor a comparación de los obtenidos. Sin embargo, al considerar otros criterios para la relevancia de documentos y creación de consultas, la arquitectura adoptada puede ser orientada en realizar una búsqueda enfocada en la medición de exhaustividad, es decir, se perdería

precisión en los resultados pero los mismos serían significativamente más relevantes para la cadena de búsqueda.

Los componentes presentados en conjunto funcionan como un mecanismo capaz de proveer la funcionalidad de federación con todas las características que involucran a la misma y que además, funcionan como solución a la problemática presentada en este trabajo.

VIII.1. Aportaciones

Dentro de las principales aportaciones de este trabajo, se consideran:

- El análisis, estudio y definición de los mecanismos utilizados para dar solución a los problemas que involucran la búsqueda federada (selección y recuperación de recursos, y combinación de resultados) los cuales pueden ser implementados en diferentes sistemas debido a que fueron creados en base a estándares de integración e interoperabilidad.
- La arquitectura propuesta sirve como base para una implementación que involucre repositorios, metadatos y objetos de aprendizaje reales, siempre y cuando cumplan con los estándares utilizados en la misma.
- Debido a que la arquitectura propuesta es una basada en mediadores, es posible una integración de repositorios que no utilicen el protocolo implementado en la misma; bastaría con crear su respectiva envoltura que permitiera la comunicación entre el mediador y la fuente física de información, implementando los mecanismos necesarios para la recuperación de la misma. La creación de esta envoltura seria delegada al administrador de la fuente que se desee integrar.
- En términos generales, toda la información presentada en esta tesis, puede ser tomada como referencia para investigaciones posteriores, dado que la mayoría de la información referente a búsquedas federadas, se encuentran en un idioma diferente al español.

VIII.2. Trabajo futuro

A continuación se presentan ciertos puntos que pueden dar pauta a investigación futura y que servirían para fortalecer la investigación presentada en esta tesis:

- Implementación de algún tipo de componente sobre reconocimiento semántico; puesto que el mecanismo empleado actualmente, sólo busca ocurrencias exactas de palabras.
- Ampliación de las cadenas de búsqueda; es decir, implementar algún tipo de mecanismo que permita la ejecución de múltiples palabras. Esto debido a que, el mecanismo actual, busca las ocurrencias de las palabras o frases exactas.
- Implementación de la arquitectura propuesta utilizando repositorios, objetos de aprendizaje y metadatos reales, con la finalidad de corroborar que la misma funcione de manera correcta en ambientes de trabajo reales.
- Diseño de las consultas apoyándose por expertos en las materias correspondientes así como por parte de los usuarios finales; esto con la finalidad de no dejar cabida a cadenas formadas de manera incorrecta por falta de conocimiento pero sin dejar de lado el lenguaje coloquial o no especializado por parte de algunos usuarios finales.
- Implementación de un componente que permita la integración con perfiles de usuario, permitiendo que las búsquedas federadas sean efectuadas con la finalidad de recuperar recursos que se adapten de mejor manera a cada usuario en particular.
- Una ampliación de las búsquedas federadas, es decir, actualmente las búsquedas solo son efectuadas en los repositorios dados de alta en la federación, sin embargo dicho proceso puede ser extendido con el propósito de que, en caso de no encontrar resultados que concuerden con la cadena de búsqueda, se puedan regresar cierto tipo de los mismos en lugar de regresar valores nulos.
- Nueva evaluación de los mecanismos de recuperación de documentos o metadatos, pero comparándolo con otras implementaciones o motores de búsqueda, con el fin de determinar el desempeño general de la arquitectura propuesta.

Referencias

(JISC), J. I. S. C. (2002). "Subject Portals Project Phase II." Consultado el 28 de Febrero de 2008. Disponible en http://www.portal.ac.uk/spp/.

(JISC), J. I. S. C. (2004). "Middleware for Distributed Cognition." Consultado el 28 de febrero de 2008. Disponible en http://www.jisc.ac.uk/whatwedo/programmes/elearning_framework/mdc.aspx.

(JISC), J. I. S. C. (2005). "Resource List Toolkit." Consultado el 28 de febrero de 2008. Disponible en http://tweed.lib.ed.ac.uk/RLI/.

(JISC), J. I. S. C. (2007). "Accesing and Storing Knowledge (ASK)." Consultado el 28 de febrero de 2008. Disponible en http://projects.oucs.ox.ac.uk/ask/.

ADL (2006). Advanced Distributed Learning (ADL) Sharable Content Object Reference Model (SCORM®) 2004 3rd EditionOverview, ADL Advanced Distributed Learning.

ADL. (2007). "ADL Advanced Distributed Learning." Consultado el el 26 de marzo de 2008. Disponible en http://www.adlnet.gov/index.aspx.

AICC. (2008). "Aviation Industry CBT (Computer-Based Training) Committee (AICC)." Consultado el el 18 de marzo de 2008. Disponible en http://www.aicc.org/.

ANSI/NISO-Z39.50-2003 (2003). Information Retrieval (Z39.50): Application Service Definition and Protocol Specification. ANSI/NISO Z39.50-2003**:** 276 pp.

Arencibia, R. (2006). "Las iniciativas para el acceso abierto a la información científica en el contexto de la web semántica." E-LIS Comité Editorial, Biblios **7**(25): 14.

Berlanga, A. J. y F. J. García (2004). "Introducción a los estándares y especificaciones para ambientes e-learning." En F. J. García Peñalvo y Mª N. Moreno García (Eds.) Tendencias en el Desarrollo de Aplicaciones Web: 25-37.

Borgman, C. L. (1999). "What are digital libraries, who is building them, and why?" Digital libraries: Interdisciplinary concepts, challenges and opportunities: 29.

Breg, F. y C. D. Polychronopoulos (2005). "Computational network federations: a middleware architecture for network-based computing." IEEE Journal onSelected Areas in Communications **23**(10): 2041-2048.

Buneman, P. (1997). "Semistructured Data." Proceedings of the sixteenth ACM SIGACT-SIGMOD-SIGART symposium on Principles of database systems: 117-121.

Busse, S., R.-D. Kutsche, U. Leser y H. Weber (1999). Federated Information Systems: Concepts, Terminology and Architectures, Technische Universität Berlin.

Busse, S. y C. Pons (2001). Schema Evolution in Federated Information Systems. Datenbanksysteme in Büro, Technik und Wissenschaft (BTW), 9. GI-Fachtagung, Londres, Reino Unido, Springer-Verlag.

Calegari, D., M. Viera y R. Motz (2005). "Design of a Service-Oriented Architecture for Federated Systems." JCS\&T **5**(4): 6.

Callan, J. (2000). Distributed information retrieval. En: W.B. Croft (ed.) Advances in information retrieval. , Springler, Berlin 127-150 p.

Callan, J. P., W. B. Croft y S. M. Harding (1992). The INQUERY Retrieval System. Proceedings of DEXA-92, 3rd International Conference on Database and Expert Systems Applications, Valencia, España.

Callan, J. P., Z. Lu y W. B. Croft (1995). Searching Distributed Collections With Inference Networks. Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Seattle, Washington, United States, ACM Press.

Canabal-Barreiro, J. M. y A. Sarasa-Cabezuelo (2007). Agrega- Plataforma de Objetos Digitales Educativos. IV Smposio Pluridisciplinar sobre Diseño, Evaluación y Desarrollo de Contenidos Educativos Reutilizables (SPDECE07). Bilbao, España. **20 de septiembre de 2007**.

CETIS. (2003). "Why are Learnign Standars so Important?" Consultado el el 17 de marzo de 2007. Disponible en http://zope.cetis.ac.uk/static/why.html.

Consortium, I. G. L. (2001). "IMS Learning Resource Meta-Data Best Practice and Implementation Guide." Consultado el 28 de noviembre de 2007. Disponible en http://www.imsglobal.org/metadata/imsmdv1p2p1/imsmd_bestv1p2p1.html.

Consortium, I. G. L. (2003). "IMS Digital Repositories Interoperability - Core Functions Information Model." Consultado el 24 de noviembre de 2007. Disponible en http://www.imsglobal.org/digitalrepositories/driv1p0/imsdri_infov1p0.html.

Consortium, I. G. L. (2004). "IMS Content Packaging Information Model." Consultado el 28 de noviembre de 2007. Disponible en http://www.imsglobal.org/content/packaging/cpv1p1p4/imscp_infov1p1p4.html.

Craswell, N. E. (2000). Methods for Distributed Information Retrieval, The Australian National University. **Tesis Doctoral**.

Chernov, S. (2005). Result Merging in a Peer-to-Peer Web Search Engine, UNIVERSITÄT DES SAARLANDES. **Tesis de Maestría**.

Daniel, G. (2004). "Learning Object Repositories." Consultado el 20 de Enero de 2008. Disponible en http://magazines.fasfind.com/wwwtools/m/1030.cfm.

David, J. R. y C. Lagoze (2000). "NCSTRL: Design and deployment of a globally distributed digital library." Journal of the American Society of Information Science **51**(3): 273-280.

DCMI. (2008). "Dublin Core Metadata Element Set, Version 1.1." Consultado el 28 de Marzo de 2008. Disponible en http://dublincore.org/documents/dces/.

DCMI. (2008). "Dublin Core Metadata Initiative (DCMI)." Consultado el 28 de marzo de 2008. Disponible en http://dublincore.org/.

Downes, S. (2004). The Learning Marketplace. Meaning, Metadata and Content Syndication in the Learning Object Economy, Moncton, New Brunswick.

Duncan, C. (2003). Granularization. Reusing Online Resources: A Sustainable Approach to E-learning. A. Littlejohn, Kogan Page**:** 12-19.

Duschka, O. M. y M. R. Genesereth (1997). Answering recursive queries using views. 16th ACM Symposium on Principles of Database Systems, Tuscon, Arizona.

Duval, E., E. Forte, K. Cardinaels, B. Verhoeven, R. V. Durm, K. Hendrikx, M. W. Forte, N. Ebel, M. Macowicz, K. Warkentyne y F. Haenni (2001). "The Ariadne knowledge pool system." Communications of the ACM **44**(5): 72-78.

Eder, J. y H. Frank (1994). "Schema Integration for Object Oriented Database Systems." Tanik M., Rossak W., Cooke D. (eds.): Software Systems in Engineering, ASME **59**: 275-284.

EDUMAT-TI. (2000). "Grupo de Educación de las Matematicas con Tecnologias de la Información (EDUMAT-TI)." Consultado el 20 de febrero de 2007. Disponible en http://azul.iing.mxl.uabc.mx/.

Fankhauser, P., G. Gardarin, M. Lopez, J. Munoz y A. Tomasic (1998). "Experiences in Federated Databases: From IRO-DB to MIRO-Web." The VLDB Journal: 655-658.

Fernández-Luna, J. M. (2001). Modelos de Sistemas de recuperación de información basados en redes de creencia. Departamento de Ciencias de la Computacion e Inteligencia Artificial. Granada, España, Universidad de Granada, Escuela Técnica Superior de Ingeniería Informática. **Tesis Doctoral**.

Frakes, W. B. y R. Baeza-Yates (1992). Information Retrieval. Data Structures & Algorithms, Prentice-Hall. 300 pp.

Frenchm, J. C., A. L. Powell, J. Callan, C. L. V. T. Emmitt, K. J. Prey y Y. Mou (1999). Comparing the Performance of Database Selection Algorithms, Research and Development in Information Retrieval**:** 238-245.

Garcia-Molina, H., Y. Papakonstantinou, D. Quass, A. R. Y. Sagiv, J. D. Ullman, V. Vassalos y J. Widom (1997). "The TSIMMIS Approach to Mediation: Data Models and Languages." Journal of Intelligent Information Systems **8**(2): 117-132.

Genesereth, M. R., A. M. Keller y O. M. Duschka (1997). Infomaster: An Information Integration System. ACM SIGMOD Int. Conference on Management of Data 1997, Tuscon, Arizona.

Gómez-Díaz, R. (2003). "La evaluación en recuperación de la información." Consultado el 25 de Julio de 2008. Disponible en http://www.hipertext.net/web/pag238.htm

González, M. (2005). Cómo desarrollar contenidos para la formación online basados en objetos de aprendizaje. RED Revista de Educación a Distancia. **Número monográfico III**.

Gravano, L. y H. García-Molina (1995). Generalizing GIOSS to Vector-Space Database and Broker Hierarchies. Proceedings of the 21th Interntional Conference on Very Large Data Bases, Morgan Kaufmanm Publishers Inc. San Francisco, CA.

Gravano, L., H. García-Molina y A. Tomasic (1994). "The effectiveness of GIOSS for the text database discovery problem." ACM SIGMOD Record **23**(2): 126-137.

Gravano, L., H. García-Molina y A. Tomasic (1999). "GIOSS: text-source discovery over the Internet." ACM Transactions on Database Systems (TODS) **24**(2): 229-264.

Greengrass, E. (2000). Information Retrieval: A Survey**:** Disponible en http://clgiles.ist.psu.edu/IST441/materials/texts/IR.report.120600.book.pdf.

Guri-Rosenblit, S. (2002). "A Top Down Strategy to Enhance Information Technologies into Israeli Higher Eductation." International Reviw of Research in Open and Distance Learning **2**(2).

Hatala, M., G. Richards, T. Eap y J. Willms (2004). "The Interoperability of Learning Object Repositories and Services: Standards, Implementations and Lessons Learned." ACM Press: 19-27.

Hillmann, D. (2005). "Using Dublin Core." Dublin Core Metadata Initiative (DCMI) Consultado el 15 de Enero de 2008. Disponible en http://dublincore.org/documents/usageguide/.

Hull, R. (1997). Managing semantic heterogeneity in databases: a theoretical prospective. ACM Symposium on Principles of Databases (Invited Tutorial).

IEEE. (2001). "Learning Object Metadata Working Group." Consultado el 19 de enero de 2008. Disponible en http://ltsc.ieee.org/wg12/index.html.

IEEE. (2007). "IEEE Institute of Electrical and Electronics Engineers/Learning Technology Standards Committee." Consultado el 30 de marzo de 2007. Disponible en http://www.ieee.org/portal/site.

IEEE. (2007). "IEEE Learning Technology Standards Committe." Consultado el 30 de marzo de 2007. Disponible en http://ieeeltsc.org/.

IEEE/LTSC (2002). "Draft Standard for Learning Object Metadata." Learning Technology Standards Committee of the IEEE.

IMS. (2004). "Instructional Management Systems (IMS) Global Learning Consortium." Consultado el 25 de noviembre de 2007. Disponible en http://www.imsglobal.org/.

Ince, L. (2000). The role of expert systems in federated database systems, Naval Postgraduate School. **Tesis de Maestría**.

JORUM+Project y MIMAS (2004). The JISC Online Repository for [learning and teaching] Materials.

Klischewski, R. (2003). "Top down or botton up? How to establish a common ground for semantic interoperability within e-government communities." E-government: modelling norms and conceps as key issues. Proceedings of 1st international workshop on e-government at ICAIL 2003: 17-26.

Larios-Osorio, V. (1999, 21 de Septiembre de 1999). "Teoría de Muestreo." Consultado el 15 de Agosto de 2008. Disponible en http://www.uaq.mx/matematicas/estadisticas/xu5.html.

Leiner, B. M. (1998). "The Scope of the Digital Library." DLib Working Group on Digital Library Metrics Consultado el 29 de octubre de 2007. Disponible en http://www.dlib.org/metrics/public/papers/dig-lib-scope.html.

Leser, U. (2000). Query Planning in Mediator Based Information Systems, Informatik der Technischen Universität Berlin zur Erlangung des akademischen Grades. **Tesis Doctoral**.

Leslie, S., B. Landon, B. Lamb y R. Poulin (2004). Learning Object Repository Software. Recuperado el 17 de febrero de 2008. Disponible en http://www.edutools.info/static.jsp?pj=4\&page=LOR, WCET's EduTools.

Levy, A. Y., A. Rajaraman y J. J. Ordille (1996). Querying Heterogeneous Information Sources Using Source Descriptions. Twenty-second International Conference on Very Large Databases, Bombay, India, VLDB Endowment, Saratoga, Calif.

Levy, A. Y., A. Rajaraman y J. J. Ordille (1996b). Query-Answering Algorithms for Information Agents. Thirteenth National Conference on Artificial Intelligence and the Eighth Innovative Applications of Artificial Intelligence Conference, Menlo Park, AAAI Press / MIT Press.

Litwin, W., L. Mark y N. Roussopoulos (1990). "Interoperability of multiple autonomous databases." ACM Computing Surveys (CSUR) **22**: 267 - 293.

Looms, T. y C. Christensen (2002). Advanced Distributed Learning Emerging and Enabling Technologies for the Design of Learning Object Repositories Report. 1901 N. Beauregard Street Alexandria, VA 22311, ADL. Advanced Distributed Learning Initiative.

López, C. (2005). Los repositorios de objetos de aprendizaje como soporte a un entorno de e-learning. Dpto. de Informática y Automática. Salamanca, España, Universidad de Salamanca. **Tesis Doctoral**.

López, C., F. J. García y P. Pernías (2005). "Desarrollo de repositorios de objetos de aprendizaje a través de la reutilización de los metadatos de una colección digital: de Dublin Core a IMS." RED Revista de Educacion a Distancia **Monográfico II**(2).

Manning, C. D., P. Raghavan y H. Schütze (2008). An Introduction to Information Retrirval, Cambridge University Press.

Martínez-Méndez, F. J. y J. V. Rodríguez-Muñoz (2004). "Reflexiones sobre la Evaluación de los Sistemas de Recuperación de Información: Necesidad, Utilidad y Viabilidad." Anales de Documentación(7): 153-170.

McLean, N. y C. Lynch (2003). "Interoperability between Information and Learning Environments - Bridging the Gaps." A Joint White Paper on behalf of the IMS Global Learning Consortium and the Coalition for Networked Information.

OAI (2004) "Open Archives Initiative Protocol for Metadata Harvesting." **Volume**, DOI:

Papakonstantinou, Y., S. Abiteboul y H. Garcia-Molina (1996). "Object Fusion in Mediator System." 22nd Conf. on Very LArge Databases.

Pesch, O. (2006). "Re-inventing Federated Searching." Serials Review **32**(3): 183-185.

Porter, D., J. Curry, B. Muirhead y N. Galan (2002). A report On Learning Object Repositories, CANARIE \& Industry Canada.

Powell, A. L. (2001). Database Selection in Distributed Information Retrieval: A Study of Multi-Collection Information Retrieval. Faculty of the School of Engineering and Applied Science University of Virginia. Virginia. **Tesis Doctoral**.

Ramos, J. (1995). Using TF-IDF to Determine Word Relevance in Document Queries. Piscataway, NJ, Department of Computer Science, Rutgers University**:** 4.

Reese, T. (2005). Building Lite-Weight EAD Repositories. International Conference on Digital Libraries, Denver, CO, USA, ACM.

Rehak, D. R. y R. Mason (2003). Keeping the Learning in Learning Objects. Reusing Online Resources: A Sustainable Approach to E-learning. A. Littlejohn, Kogan Page**:** 20-34.

Roantree, M., J. B. Kennedy y P. J. Barclay (2001). Interoperable Services for Federations of Database Systems, Oasis Technical Report, Dublin City University.

Robertson, S. (1997). "The probability ranking principle in IR." Journal of Documentation **33**(4): 294-304.

Robertson, S. (2004). "Understanding Inverse Document Frequency: On theoretical arguments for IDF " Journal of Documentation **60**.

Saavedra, M. A. (2003). Arquitectura para Federación de Bases de Datos Documentales basada en Ontologías. Departamento de Computación. Coruña, Universidade da Coruña. **Tesis Doctoral**.

Schamber, L., M. B. Eisenberg y M. S. Nilan (1990). "A re-examination of relevance: toward a dynamic, situational definition." Information Processing and Management: an International Journal **26**(6): 755-776.

Sheth, A. P. y J. A. Larson (1990). "Federated Database Systems for Managing Distributed, Heterogeneous, and Autonomous Databases." ACM Computing Surveys **22**: 183 - 236.

Si, L. (2006). Federated Search of Text Search Engines in Uncooperative Enviroments. Language Technology Institute, School of Computer Science, Carnegie Mellon University. **Tesis Doctoral**.

Si, L. y J. Callan (2003). The effect of database size on resource selection algorithms. In Proc. SIGIR 2003 Workshop on Distributed Information Retrieval, Springer.

Tomasic, A., L. Raschid y P. Valduriez (1996). Scaling Heterogeneous Databases and the Design of Disco. International Conference on Distributed Computing Systems.

Tsatalos, O. G., M. H. Solomon y Y. E. Ioannidis (1996). "The GMAP: A Versatile Tool for Physical Data Independence." VLDB Journal: Very Large DataBases **5**(2): 101-118.

Tzitzikas, Y., P. Constantopoulos y N. Spyratos (2001). "Mediators over Ontology-Based Information Sources." WISE: 31-40.

Tzitzikas, Y., N. Spyratos y P. Constantopoulos (2005). "Mediators over taxonomy-based information sources." VLDB Journal(14): 112-136.

W3C. (2007). "World Wide Web Consortium (W3C)." Consultado el 1 de abril de 2007. Disponible en http://www.w3.org/.

WebFeat. (2007). "WebFeat - The original federated search engine." Consultado el 15 de Noviembre de 2007. Disponible en http://www.webfeat.org/index.htm.

Wiederhold, G. (1992). "Mediators in the Architecture of Future Information Systems." The IEEE Computer Magazine **3**(25): 38 - 49.

Wiederhold, G. (1993). "Intelligent Integration of Information." ACM-SIGMOD 93: 434 - 437.

Wiley, D. (2002). Connecting learning objects to instructional design theory: a definition, a metaphor, and a taxonomy. D. Wile. Bloomington, IN., The Agency for Instructional Technology**:** 3-23.

Apéndice A

Algoritmos de Selección de Colección

Como se mencionó en el Capítulo VI, existen ciertos modelos de recuperación de información (Fernández, 2001), de los cuales se hace énfasis en el modelo de recuperación probabilístico; esto debido a que, de acuerdo con las investigaciones consultadas (Fernández, 2001; Powell, 2001; Si y Callan, 2003; Si, 2006; Roantre, 2001), estos modelos son los que se presentan un mejor rendimiento en la recuperación, en comparación con los otros modelos mencionados en el Capítulo VI.

El modelo probabilístico se compone por conjuntos de variables, operaciones con probabilidades y el teorema de Bayes. Robertson (1997) asegura que el rendimiento óptimo de la recuperación se consigue ordenando los documentos de acuerdo a sus probabilidades que tienen de ser juzgados como relevantes con respecto a una consulta, siendo estas probabilidades calculadas de la forma más precisa posible a partir de la información disponible. A partir de este principio, el objetivo de cualquier modelo probabilístico es calcular la probabilidad de relevancia dados una consulta y un documento.

En cuanto a los algoritmos de selección de colección, como su nombre lo indica, son los encargados de encontrar un subconjunto de colecciones, las cuales son las más relevantes con respecto a una consulta dada; es decir, estos algoritmos pueden seleccionar aquellas colecciones con una mayor probabilidad de responder a la consulta realizada.

Los algoritmos de selección de colección son, de hecho, algoritmos de ordenamiento de colecciones; estos algoritmos ordenan el conjunto completo de colecciones en base a una consulta y se consideran las colecciones posicionadas en los primeros lugares del ordenamiento.

Los algoritmos descritos aquí son CORI, CVV, bGlOSS y vGlOSS. El motivo por el cual consideramos los algoritmos anteriores, es que de acuerdo con la literatura consulta, éstos son los más utilizados comúnmente y los que mejores resultados aportan (Roantree, 2001; Powell, 2001;Frenchm et al., 1999; Ince, 2000; Greengrass, 2000; Frakes y Baeza-Yates 1992).

A.1 CORI

Como ya se mencionó en el Capítulo VI, el algoritmo CORI (Callan et al., 1995) representa las colecciones como documentos virtuales, de manera que se pueda obtener información sobre los mismos. El algoritmo crea un índice de estos elementos virtuales representados por sus términos y sus frecuencias de aparición en los documentos de las colecciones originales.

La probabilidad de que un término r_k exista en una colección C_i, es decir $p(r_k|C_i)$, es calculada en base a las siguientes fórmulas:

$$T = \frac{df}{df + 50 + 150 \cdot cw / \overline{cw}} \tag{11}$$

$$I = \frac{\log\left(\frac{N + 0.5}{cf}\right)}{\log(N + 1.0)} \tag{12}$$

$$p(r_k|C_i) = 0.4 + 0.6 \cdot T \cdot I \tag{13}$$

Donde:

df es el número de documentos en la colección C_i que contienen el término r_k
cf es el número de colecciones que contienen a r_k
N es el número de colecciones totales a ordenar
cw es el número de palabras en C_i
$\overline{cw}$ es la media del número de palabras en las colecciones

Una vez que se ha creado el índice de las colecciones ordenadas en base a una consulta dada, es posible discriminar de entre las mismas, dicho de otra manera, seleccionar las colecciones ubicadas en los primeros lugares del ordenamiento, con el propósito de satisfacer de manera correcta a la consulta inicial, es decir, recuperar mejores resultados.

Un análisis detallado de CORI es presentado en el Capitulo VI, puesto que es en éste algoritmo en el cual se basa la selección de repositorios respecto a la investigación presentada en esta tesis.

A.2 CVV

Yuwono y Lee (1997) propusieron una aproximación al gran problema de la búsqueda distribuida, consideraron la selección de colección, reenvío de consultas y combinación de resultados. Ellos se refirieron a la porción de selección de colección como el método de ordenamiento *Cue Validity Variance* (*CVV*). CVV se refiere al método de ordenamiento y al componente en su cálculo del puntaje estimado o mérito de la colección.

De manera descriptiva, ha sido observado que una característica F_i es útil para categorizar un concepto X_j si esa característica no está asociada con otro concepto contrastante. La definición tradicional de la validez de entrada de F_i para X_j es:

$$P(X_j \mid F_i) = \frac{P(F_i \mid X_j)}{P(F_i \mid X_j) + P(F_i \mid X_k)} \tag{14}$$

Donde X_k es algún concepto contrastante. La validez de entrada se incremente con la probabilidad que la característica F_i describe el concepto X_j pero decrementa si F_i es también representativa de algún otro concepto X_k.

El método de ordenamiento CVV emplea una combinación de la información de la frecuencia de documento (*df*) y la validez de varianza de entrada. La validez de varianza de entrada tiende a caracterizar la distribución de la densidad de los valores *df* , es decir, la variabilidad de la fracción de documentos en una colección que contienen el término dado. La información de la frecuencia de documentos es utilizada para aproximar qué tan importante es un término en su colección; el objetivo de CVV es estimar si un término es útil para diferenciar una colección de otra.

El cálculo de la estimación del puntaje o mérito de la colección se presenta resumido a continuación. Cabe resaltar que existen ciertas diferencias en cuanto a la notación se refiere, entre las fórmulas mostradas aquí y las presentadas en Yuwono y Lee (1997); lo anterior como notación para un mejor entendimiento y comprensión de dicho cálculo.

La formula para calcular el puntaje o mérito estimado de la colección es:

$$merito\_est(C_i, q) = \sum_{\{t_j \in q\}} CVV_j \cdot df_{ij} \tag{15}$$

Donde t_j es un término en la consulta q, df_{ij} es la frecuencia de documento de t_j en la colección C_i, y

$$CVV_j = \frac{\sum_{i=1}^{N} \left(CV_{ij} - \overline{CV_j} \right)^2}{N} \tag{16}$$

Donde

$$CV_{ij} = \frac{\frac{df_{ij}}{|C_i|}}{\frac{df_{ij}}{|C_i|} + \frac{\sum_{k \neq i}^{N} df_{kj}}{\sum_{k \neq i}^{N} |C_k|}} \tag{17}$$

Y, N es el número de colecciones en el sistema, y

$$\overline{CV_j} = \frac{\sum_{i=1}^{N} CV_{ij}}{N} \tag{18}$$

El método de ordenamiento CVV utiliza solo información de (o derivable de) una matriz de estadísticas de frecuencias de documentos. El objetivo de el cálculo del merito estimado es el de identificar las colecciones con alta concentración de términos de consulta.

Mayor información sobre el algoritmo puede ser encontrada en: Yuwono y Lee (1997), Powell (2001) y Craswell (2000).

A.3 bGIOSS y vGIOSS

Gravano y García-Molina propusieron los algoritmos de descubrimiento de fuentes de texto bGIOSS (Gravano *et al.* 1994) y vGIOSS (Gravano *et al.* 1999) (aunque vGIOSS fue descrito primero como gGIOSS en [Gravano y García-Molina 1995]). El método bGIOSS ordena servidores[14] los cuales presentan un sistema de recuperación booleano, usando frecuencia de documentos e información sobre el tamaño del servidor así como una

[14] En el contexto de estos algoritmos, las colecciones de documentos son conocidas como servidores, por lo que en esta sección utilizaremos esa terminología.

suposición sobre la distribución de los términos de la consulta en los documentos.

Por ejemplo, supongamos que las distribuciones del término de la consulta son independientes, es posible estimar el numero de documentos que concuerden con una cadena conjuntiva de dos términos: dado un servidor con 1500 documentos, con 200 de ellos que contienen el término *A* y 50 de los mismos que contienen el término *B*, el número estimado de documentos que satisfacen la consulta $A \wedge B$ es $\frac{200}{1500} \times \frac{50}{1500} \times 1500 = 6\frac{2}{3}$. bGIOSS entonces ordena servidores en orden descendiente del numero estimado de documentos relevantes. Gravano y García-Molina admiten que esta suposición de independencia es cuestionable, pero permite buenos resultados en los experimentos realizados.

Los métodos de ordenamiento de servidores *Max(l)* y *Sum(l)* de vGIOSS, ordenan servidores en los cuales los sistemas de recuperación están basados en el modelo de espacio de vectores. Para una mejor comprensión del método, es necesaria una breve explicación del modelo de espacio de vectores. Cada dimensión en un espacio de vector simple para recuperación de documentos corresponde a un término el cual ocurre en la colección de documentos a ser buscada. Un documento d_j puede ser representado como un vector en ese espacio, con su peso en una dimensión $wt_{j,k}$ proporcional a la frecuencia de documento del término t_k en la colección.

Los vectores de documentos en el espacio son muy escasos, conteniendo pesos diferentes de cero para solo una fracción del vocabulario total de la colección. Para llevar a cabo la recuperación, la consulta es creada como un vector en el espacio de documento y los documentos que concuerden más con la misma, son posicionados en los primeros lugares del ordenamiento. Los vectores de documento están casi siempre normalizados a recuperar documentos de cualquier tamaño, más que enfocarse a documentos grandes que pueden tener grandes pesos debido a la gran cantidad de vocabulario.

La selección de vGIOSS esta basada en la suma de los vectores normalizados de documentos de los servidores así como sus estadísticas de frecuencia de documentos. La suma de vectores de un servidor s_i y el término t_k es:

$$cwt_{i,j} = \sum_{j=0}^{j<SS_i} wt_{j,k} \tag{19}$$

Donde $wt_{j,k}$ es el peso del término t_k en el vector de documento normalizado representando el documento d_j del servidor s_i.

Max(l) y *Sum(l)* acumulan puntajes de todos los documentos en los cuales el puntaje esta por encima de *l*, para una consulta particular en el servidor. Ambos cálculos son estimados, basados en los pesos exportados y estadísticas de frecuencia de documentos, de una suma de documentos *Ideal(l)* para documentos con puntaje por arriba de *l*.

Gravano y García-Molina (1995) dijeron que lo mejor que se puede esperar de cualquier herramienta como vGIOSS (gGIOSS) es que predice las respuestas que los servidores darán cuando se presente una consulta.

A.4 Comparaciones de estos algoritmos

Powell (2001), Frenchm et al. (1999), Si (2006) y Craswell (2000) realizaron ciertas pruebas de comparación de los algoritmos mencionados en este apartado, en las cuales se pudo determinar que estos algoritmos realizan un trabajo similar al poner colecciones con bajo mérito o probabilidad al final de sus respectivos ordenamientos. Las diferencias entre estos algoritmos son pequeñas en cuanto a consultas cortas se refiere, sin embargo en promedio CORI parece ser afectado más por el efecto de las consultas cortas.

Las investigaciones realizadas arrojaron resultados similares a los expresados por Powell (2001), quien dice que la aproximación de selección de colección CORI se desempeña de

una manera más adecuada y consistente. De acuerdo con las pruebas, CORI es significativamente mejor que las otras aproximaciones.

Cuando se compara CORI y el *Ideal(l)* de vGlOSS, de una manera amplia, la principal ventaja de CORI es que requiere menos información estadística sobre las colecciones. La información *df* que CORI requiere es también más fácil de comparar entre colecciones usando diferentes sistemas de recuperación de información, que la información de pesos de términos que requiere vGlOSS. La información *df* puede ser eficientemente aproximada o calculada por técnicas de muestreo.

CVV es una aproximación intuitiva atrayente que también requiere información estadística limitada, además CVV fue el que se comportó de peor manera en los experimentos. CVV puede saber los términos que existen en cada descripción de colección.

A.5 Resumen

En el presente Apéndice, se mencionaron algunos algoritmos de selección de colecciones que utilizan el modelo de recuperación probabilístico. Estos algoritmos fueron seleccionados en base a que la mayoría de la literatura consultada, concuerda en que dichos algoritmos son los más utilizados en los procesos de recuperación de información.

Un análisis del desempeño de los algoritmos, concluye en que el algoritmo con mejor desempeño en cuanto a ordenamiento de colecciones, es el algoritmo CORI, el cual se basa en el modelo de recuperación de información general ($df \cdot icf$) utilizando información estadística de las colecciones con el fin de realizar su ordenamiento correcto basado en una consulta hecha. El análisis realizado por los investigadores mencionados, fue la principal causa del por qué hemos decidido utilizar el algoritmo de selección de colecciones CORI.

Apéndice B

Estándares y Especificaciones de Integración

Previamente se ha dado mención al término estándar o estandarización, ya que al manejar componentes heterogéneos para distintos sistemas y tecnologías, la estandarización se convierte en un tema clave para lograr una integración de los componentes y alcanzar el objetivo de compartir datos entre distintas plataformas y sistemas.

La estandarización debe ser alcanzada en distintos niveles; primero; cuando se están creando los componentes con el fin de que sean compatibles con distinto sistemas que deseen reutilizarlos; segundo, cuando dichos componentes son incluidos en un repositorio y deben ser descritos con sus metadatos para facilitar su localización y garantizar su compatibilidad; tercero, en la utilización de los componentes cuando sean integrados en otros sistemas; y cuarto, cuando los sistemas debe comunicarse con otros sistemas para cumplir sus funciones o incrementar las mismas.

Básicamente, un estándar tiene como objetivo acordar de qué forma compartir, comunicar o desarrollar modelos y sistemas con el propósito de interoperar entre los diversos componentes.

Existen diferentes grupos que trabajan en el desarrollo tanto de estándares como de especificaciones en los distintos niveles que se requieren para poder lograr entornos integrados e interoperables.

B. 1 Estándar y Especificación

Existe cierta confusión en cuanto a las definiciones de estándar y especificación, por lo que se mencionan las diferencias entre los mismos, con el propósito de aclarar las dudas que pudieran existir.

De acuerdo con la Real Academia Española[15], un estándar es algo "que sirve como tipo, modelo, norma, patrón o referencia"; mientras que una especificación corresponde a una "acción o efecto de especificar" que significa "explicar, declarar algo con individualidad, fijar o determinar de modo preciso".

Según Berlanga y García (2004) una tecnología, formato o método que ha sido ratificado por algún organismo oficial de estandarización, se trata de un estándar. Pero si una tecnología, formato o método propuesto no ha sido aprobado por algún organismo oficial de estandarización, se trata de una especificación, aunque puede considerarse como estándar de facto si su uso es extendido.

De acuerdo con López (2005) existen dos tipos de estándares:

- Estándares de jure, cuando provienen de una organización acreditada que certifica una especificación; y
- Estándares de facto, cuando la especificación ha sido adoptada por un grupo mayoritario de individuos.

Y considera que una especificación es un conjunto de declaraciones detalladas y exactas de los requisitos funcionales y particularidades de algo que quiere construirse, instalarse o manufacturarse. De lo anterior se podría inferir que un estándar siempre proviene de una especificación.

---

[15] http://www.rae.es/rae.html

Las definiciones de Berlanga y García (2004) y López (2005) son muy parecidas, además la Real Academia Española permite corroborar que la idea es muy similar, por lo que definiremos que se le llama a una tecnología o método que ha sido aprobada por un organismo de estandarización, un estándar; mientras que una especificación es definida como un conjunto de reglas exactas, así como requisitos y particularidades funcionales para construir o instalar algo.

Cuando una especificación es adoptada por algún grupo importante de desarrollo, éste último le asigna el nombre de estándar, sin embargo como no ha sido aprobada por alguna organización oficial, el estándar adoptado es conocido como estándar de facto.

B. 2 Ventajas del uso de estándares

Como se ha mencionado en capítulos anteriores, la estandarización es un punto de vital importancia si lo que se pretende es interoperar, crecer, expandir o generalizar distintos sistemas y utilizar componentes heterogéneos. El uso de estándares en el proceso de integración, tiene varias ventajas, entre ellas:

- Reutilización y consistencia de los contenidos.
- Normalización en la organización de los recursos
- Acceso a más fuentes y recursos.
- Migración sencilla de los sistemas a nuevas versiones o plataformas.
- Comunicación e intercambio de información con otros sistemas.

La utilización de estándares amplía las opciones de los usuarios finales, reduciendo las restricciones de los sistemas propietarios y de soluciones aisladas (CETIS 2003). Los administradores de sistemas de información se verán beneficiados al garantizar la interoperación de sus componentes y sistemas con otros de diferente estructura y organización.

B. 3 Grupos de desarrollo

Los repositorios de objetos de aprendizaje, así como los mismos objetos de aprendizaje, están enfocados a la enseñanza asistida por computadora, es decir, dichos componentes son englobados en un entorno llamado aprendizaje electrónico, (en inglés: e-learning), dicho de otra manera, es un tipo de educación a distancia basada en Web, el cual hace uso de servicios y facilidades de Internet para fortalecer el proceso de enseñanza-aprendizaje (López 2005).

Existen grupos encargados del desarrollo de especificaciones y estándares orientados a identificar, definir y comunicar los recursos involucrados en el entorno de aprendizaje electrónico. Estos grupos son:

- AICC (Aviation Industry Computer-Based Training Committee). Es una asociación de entrenamiento profesional basado en tecnología, especializado en el sector de la aviación pero que se ha ido desviando a otros sectores. Se reconoce como una de los precursores de la estandarización de materiales del entrenamiento profesional (AICC 2008).
- IMS Global Consortium. Cuenta con miembros de organizaciones comerciales, educativas y gubernamentales dedicadas a definir y distribuir arquitecturas abiertas para actividades de educación en línea. Habilita el crecimiento e impacto de tecnología de aprendizaje en la educación. Uno de sus resultados es el estándar IMS (IMS 2004;Consortium 2001; Consortium 2003; Consortium 2004).
- Advanced Distributed Learning (ADL). En 1997 el Departamento de Defensa de Estados Unidos y la Oficina de Ciencia y Políticas Tecnológicas de la Casa Blanca lanzan la iniciativa ADL. La iniciativa es un esfuerzo colaborativo entre el gobierno, la industria y la academia para establecer un nuevo ambiente de aprendizaje distribuido que permita la interoperabilidad de herramientas de aprendizaje y contenido educativo en una escala global (ADL 2007). Para cumplir

con estos objetivos crean el modelo SCORM (Sharable Content Object Reference Model) (ADL 2006).

- ARIADNE (Alliance of Remote Instructional Authoring and Distribution Networks for Europe). Es un proyecto de investigación y desarrollo tecnológico de telemática para la educación y el entrenamiento, patrocinado por la Unión Europea, el cual inició en 1996. El proyecto esta enfocado al desarrollo de herramientas y metodologías con el fin de producir, administrar y reutilizar elementos pedagógicos asistidos por computadora (Duval *et al.* 2001).
- IEEE/LTSC (Institute of Electrical and Electronics Engineers/Learning Technology Standards Committee). El IEEE es una asociación internacional, que promueve procesos para la creación, desarrollo, integración, compartición y aplicación del conocimiento sobre tecnologías electrónicas y de ciencias afines. IEEE funciona como catalizador para innovación tecnológica y soporta las necesidades de sus miembros a través de una amplia variedad de programas y servicios (IEEE 2007). IEEE cuenta con el Comité de Estándares para Tecnología del aprendizaje (LTSC, por sus siglas en inglés), el cual desarrolla estándares técnicos acreditados internacionalmente, practicas recomendadas así como guías para la tecnología de aprendizaje (IEEE 2007).
- W3C. (World Wide Web Consortium). El consorcio W3C se encarga del desarrollo de estándares Web, en otras palabras es una guía para obtener un máximo potencial de la Web a través del desarrollo de protocolos y pautas que aseguren el crecimiento de la misma. El W3C no esta directamente relacionado con el aprendizaje electrónico, sin embargo este último utiliza o depende de la interoperabilidad que se encuentre en la Web (W3C 2007).

Las necesidades de interoperabilidad en el contexto del presente trabajo, abarcan interoperabilidad de objetos de aprendizaje, de repositorios y de metadatos, por lo que a continuación se describen a detalle los estándares que proporcionan las herramientas necesarias para cumplir con la integración de componentes en el marco del presente trabajo.

B. 4 IEEE LOM

Este estándar especifica la sintaxis y semántica de los metadatos de un objeto de aprendizaje, definidos como atributos requeridos para describir completa y adecuadamente a un objeto de aprendizaje. El estándar se enfoca en un conjunto mínimo de atributos necesarios para permitir a los objetos de aprendizaje ser manejados, ubicados y evaluados

Fue desarrollado por la IEEE el cual especifica características para metadatos de objetos de aprendizaje. El LOM (Learning Object Metadata) especifica un esquema conceptual de datos que define la estructura de una instancia de metadatos para un objeto de aprendizaje; en otras palabras, describe las características relevantes del objeto de aprendizaje al que hace referencia (IEEE/LTSC 2002). Dichas características pueden ser agrupadas en las siguientes categorías:

- General (General). Agrupa la información general que describe un objeto de aprendizaje de manera global.
- Lifecycle (Ciclo de vida). Agrupa características relacionadas con la historia y el estado actual del objeto de aprendizaje, así como aquellas que lo han afectado en su evolución.
- Meta-Metadata (Meta-Metadatos). Agrupa la información sobre la propia instancia de Metadatos (en lugar del objeto de aprendizaje que describe la instancia de metadatos).
- Technical (Técnica). Agrupa los requerimientos y características técnicas del objeto educativo.
- Educational (Uso Educativo). Agrupa las características educativas y pedagógicas del objeto.
- Rights (Derechos). Agrupa los derechos de propiedad intelectual así como las condiciones para el uso del objeto educativo.
- Relation (Relación). Agrupas las características que definen la relación entre este y otros objetos de aprendizaje relacionados.

- Annotation (Anotación). Permite incluir comentarios sobre el uso educativo del objeto así como información sobre cuándo y por quién fueron creados dichos comentarios.
- Classification (Clasificación). Describe el objeto de aprendizaje en relación a un determinado sistema de clasificación.

LOM tiene varios propósitos entre los cuales están: permitir a los aprendices o instructores buscar, evaluar, adquirir y utilizar OA, así como compartir e intercambiar los mismos entre tecnologías que soporten sistemas de aprendizaje; motivar el desarrollo de objetos de aprendizaje en unidades que pueden ser combinadas y descompuestas de varias maneras; soportar la seguridad necesaria y autentificación para la distribución de objetos de aprendizaje, entre otros (IEEE 2001).

Como ya se ha mencionado, un objeto de aprendizaje puede ser referenciado y compartido gracias a sus metadatos. Podemos hablar de un documento Web que trate sobre números perfectos; este documento Web es el objeto de aprendizaje y sus metadatos indicaran su titulo, palabras clave, descripción, especificaciones técnicas, etc. La Figura 15 muestra un extracto de cómo el documento Web se anotaría siguiendo el estándar IEEE LOM.

```
<lom>
<general>
    <identifier>
        <catalog>metadatos/matematicas/MERLOT</catalog>
        <entry>LEA0875</entry>
    </identifier>
    <title lang="es">Numeros Perfectos</title>
    <language>es</language>
    <description lang="es">
        Aqui se explica y ejemplifica que Un numero perfecto es igual a la suma
        de todos sus divisores excepto el mismo
    </description>
    <keyword lang="es" weight="0">Calculo</keyword>
...
    <technical>
        <format value="text/html">
            <contentType type="image">0.1</contentType>
            <contentType type="audio">0.7</contentType>
            <contentType type="text">0.1</contentType>
            <contentType type="video">0.1</contentType>
        </format>
        <size height="400" width="400">6000</size>
        <location type="URL">
        http://azul.iing.mxl.uabc.mx/NumerosyFormas/Numeros/NumerosPerfectos/NumerosPerfectos.htm
        </location>
...
    <educational>
    <interactivityType type="1.0"/>
    <learningResourceType type="Tipo1"/>
    <interactivityLevel level="1.0"/>
    <semanticDensity>medium</semanticDensity>
    <intendedEndUserRole>
        <rol>teacher</rol>
    </intendedEndUserRole>
    <context>higher</context>
    <typicalAgeRange>
        <begin>18</begin>
        <end>21</end>
    </typicalAgeRange>
    <difficulty dif="0.01"/>
...
    </educational>
...
</lom>
```

Figura 15. Extracto de un ejemplo de un OA anotado según el estándar IEE LOM.

B. 5 Dublin Core

Dublin Core es un modelo de metadatos elaborado y mantenido por la DCMI (Dublin Core Metadata Initiative) (DCMI 2008), la cual es una organización dedicada a fomentar la adopción extensa de los estándares interoperables de los metadatos y a promover el desarrollo de los vocabularios especializados de metadatos para descubrimiento de recursos.

El nombre fue asignado debido a la primera reunión realizada de especialistas en metadatos y Web en el año 1995, realizada en Dublín (Ohio, E.U.) y "core" debido a que sus elementos son amplios, genéricos y útiles para describir un gran rango de recursos.

Dublin Core (DC) es un sistema de quince definiciones semánticas descriptivas, las cuales son opcionales, se pueden repetir y pueden aparecer en cualquier orden. Estas definiciones proporcionan un vocabulario "base" que provee información descriptiva sobre cualquier recurso (DCMI 2008a).

Los elementos que componen al estándar Dublin Core son:

- Contributor (Colaboradores). Se refiere a una entidad responsable de hacer contribuciones a la fuente.
- Coverage (Cobertura). es la característica de cobertura espacial y/o temporal del contenido intelectual del recurso. La cobertura espacial se refiere a una región física mientras que la cobertura temporal se refiere al contenido del recurso.
- Creator (Creador). Una entidad principalmente responsable de crear el recurso.
- Date (Fecha). Un punto o periodo de tiempo asociado con un evento en el ciclo de vida del recurso.
- Description (Descripción). Una descripción textual del recurso.
- Format (Formato). Se refiere al formato del archivo, medio físico o dimensiones del recurso.
- Identifier (Identificador). Una referencia única al recurso en un contexto dado.
- Language (Lenguaje). El lenguaje del contenido del recurso.
- Publisher (Editor). Entidad responsable de hacer que el recurso se encuentre disponible.
- Relation (Relación). Identificador de un segundo recurso y su relación con el recurso actual.
- Rights (Derechos). Información sobre derechos contenidos y/o sobre el recurso.

- Source (Fuente). Un Segundo recurso del cual se deriva el recurso descrito.
- Subject (Tema). Los tópicos del recurso.
- Title (Titulo). El nombre dado al recurso.
- Type (Tipo). La naturaleza o género del recurso.

El estándar Dublin Core presenta las ventajas que caracterizan a todo estándar, entre ellas: simplicidad, flexibilidad, interoperabilidad, entre otras. En la Figura 16 se puede apreciar un ejemplo de cómo luciría un objeto de aprendizaje descrito con el estándar Dublin Core; el ejemplo representa un objeto de aprendizaje sobre números perfectos.

```
<dc>
    <title>Numeros Perfectos</title>
    <creator>publisher,TEL:014422112600</creator>
    <subjetc>Calculo</subjetc>
    <description>
    Aqui se explica y ejemplifica que Un numero perfecto es igual a la suma de todos sus
    divisores excepto el mismo
    </description>
...
    <identifier>
    http://azul.iing.mxl.uabc.mx/NumerosyFormas/Numeros/NumerosPerfectos/NumerosPerfectos.htm
    </identifier>
    <source>
    en Dublin Core ispartof: es parte de,Descripcion de la relacion entre objetos.
    </source>
    <language>es</language>
...
</dc>
```

Figura 16. Extracto de un Objeto de Aprendizaje descrito con el estándar DC.

B. 6 Protocolo Z39.50

El protocolo Z39.50 (ANSI/NISO-Z39.50-2003), es un protocolo para búsqueda y recuperación de registros de una BD. Esta búsqueda y recuperación se hace mediante una asociación cliente-servidor, que en el ámbito del protocolo se conoce como asociación Z. Los servicios Z39.50 son llevados a cabo mediante el intercambio de mensajes entre el cliente y el servidor. El objetivo de este estándar es facilitar la interconexión de clientes y servidores para aplicaciones donde los clientes busquen y recuperen información de servidores de BD.

La búsqueda con este protocolo es realizada mediante una cadena en la que se especifican valores a ser comparados contra la BD. El subconjunto de resultados es llamado conjunto resultado. Este conjunto resultado está disponible desde el servidor para referencia del cliente. Dicho conjunto es una lista ordenada de elementos, cada uno con un apuntador a un registro de la BD.

En el contexto del protocolo, el funcionamiento del mismo es efectuado utilizando lo que ha sido llamado facilidades; es decir, las funciones conocidas para recuperar los registros de la base de datos. Dichas facilidades son descritas a continuación:

1. Inicialización: El cliente Z (origen) se conecta con el servidor y sugiere los parámetros básicos para la sesión. El servidor Z puede modificar estos parámetros, y si cliente y servidor llegan finalmente a un acuerdo comienza la asociación Z.
2. Búsqueda: La estrategia de búsqueda es una cadena de datos y parámetros asociados que definen los registros que se quieren recuperar como por ejemplo la base de datos a buscar, etc.
3. Recuperación: La recuperación comprende dos servicios: presentar y segmentar. Presentar es una petición al servidor Z para que envíe ciertos registros, y segmentar es el proceso de dividir un número extenso de registros es grupos pequeños para que la transmisión sea más fácil.
4. Borrado de conjunto resultado: El servicio permite borrar un juego de resultados en el origen, el Servidor Z dará una respuesta acorde.
5. Control de acceso: El control del acceso construido a través del Z permite al servidor no autorizar virtualmente el acceso a operaciones pedidas por el cliente. Si el control de acceso no funciona, la asociación Z puede continuar con las operaciones que estén permitidas llevar a cabo o finalizar. El control de acceso puede realizarse sobre registros individuales si se requiere, así una biblioteca puede tener registros que no proveerá sin autorización, por razones de seguridad, pagos, etc.

6. Control de recursos/cuentas: Este servicio es enviado por el servidor como parte de una operación específica o de la asociación Z global. Puede opcionalmente incluir un informe de recursos que puede explicar cómo están siendo los gastos actuales o los previstos y si puede excederse de los límites acordados.
7. Ordenar: Es llevada a cabo en el servidor, algunos clientes tienen implementada esa opción y pueden ordenar y filtrar la información después de recuperar los registros. Esta forma de trabajo es más fácil, más rápida y más flexible.
8. Visualizar (Browse): El servicio individual se denomina "scan" (consulta por índices o claves), y permite explorar u hojear una lista de autores o encabezamientos de materia. Esto contrasta con una búsqueda típica Z39.50, donde se recupera un conjunto de resultados en respuesta a una estrategia de búsqueda.
9. Servicios extendidos: Los servicios extendidos permiten al cliente Z aplicar paquetes de tareas sobre el servidor Z y controlar cómo van a operar.
10. Información sobre el servidor: El cliente Z puede consultar esta base de datos y descubrir los servicios concretos que ofrece el servidor así como sus características básicas.
11. Terminación: permite a un cliente Z o a un servidor Z interrumpir una asociación Z dando una razón para este cierre, por ejemplo, problemas en el sistema, límite de costes, violación de la seguridad, etc.

Los beneficios que presenta el protocolo son:

- Los clientes modernos Z pueden enviar peticiones a varias bibliotecas simultáneamente ya sea la misma petición o diferentes. Esta característica permite un gran ahorro de tiempo al realizar búsquedas de elementos poco comunes o de un gran número de registros.

- El formato básico de intercambio es MARC[16]. El cliente Z ofrece registros MARC para visualizar y realizar otros procesos posteriores. Todas las bibliotecas "negocian" con registros bibliográficos. EL Z39.50 abre este mercado estandarizando las funciones básicas de búsqueda y recuperación de la información.
- Los servicios extendidos para la petición de documentos, actualización de bases de datos y almacenamiento de búsquedas pueden ser definidos y controlados mediante Z39.50.

B. 7 OAI-PMH

El OAI-PMH es un protocolo que proporciona interoperabilidad no inmediata entre fuentes que pueden ser documentos electrónicos, bibliotecas digitales o cualquier servidor que quiera hacer visibles los metadatos de los documentos que tiene almacenados para un sistema que quiera recolectarlos. El protocolo para recolección de metadatos de la Iniciativa de Archivos Abiertos (OAI-PMH, por sus siglas en inglés) (OAI 2004), proporciona un entorno de interoperabilidad independiente de la aplicación basado en la recolección de metadatos (metadata harvesting).

La Open Archives Initiative (OAI) se creo con la misión de desarrollar y promover estándares de interoperabilidad para facilitar la difusión eficiente de contenidos en Internet. Surgió como un esfuerzo para mejorar el acceso a archivos de publicaciones electrónicas (eprints), en otras palabras, para incrementar la disponibilidad de las publicaciones científicas.

---

[16] MARC es el acrónimo para MAchine-Readable Cataloging. Que define un formato de datos que nació de la Biblioteca del Congreso de E.U., siendo la iniciativa que empezó cerca de los años cuarenta. Provee el mecanismo mediante el cual las computadoras intercambian, usan e interpretan información bibliográfica, y sus elementos de datos forman la mayoría de los catálogos utilizados en la actualidad.

Los orígenes de OAI radican en un creciente interés en la búsqueda de alternativas a los modelos tradicionales de comunicación científica. En Octubre de 1999 se organizó una reunión en Santa Fe (Nuevo México, E.U.), con la idea de que la interoperabilidad de los archivos eprints era clave para aumentar su impacto entre la comunidad académica, ya que con ella se podrían federar varios archivos, intercambiar registros o realizar búsquedas en disciplinas relacionadas, al mismo tiempo.

El resultado de la reunión fue un conjunto de acuerdos técnicos y organizativos conocidos como la Convención de Santa Fe. Los aspectos técnicos incluían tres puntos fundamentales: un formato para los metadatos, un protocolo basado en el antiguo Dienst (David y Lagoze 2000) y un sistema de identificación.

Los acuerdos establecidos dieron nacimiento al protocolo OAI-PMH, cuyas especificaciones fueron hechas en Enero de 2001. Los participantes en Santa Fe tomaron una decisión clave en cuanto a la arquitectura del protocolo. Adoptaron un modelo que rechazaba la búsqueda distribuida a favor de simplemente contar con servidores proporcionando metadatos.

El OAI-PMH define el intercambio de solicitudes y metadatos entre el servidor de documentos digitales y un programa recolector externo; proporciona un entorno de interoperabilidad independiente de la aplicación basado en la recolección de metadatos. En OAI existen las instituciones llamadas Proveedores de Datos (Data Providers), que ofrecen facilidades para publicación y almacenamiento de documentos y su distribución a través de un servidor conectado a Internet. El otro tipo de institución que existe son los Proveedores de Servicios (Service Provider) que recolectan metadatos de uno o más proveedores de datos y con esos metadatos ofrecen servicios de valor añadido. Ejemplos de estos servicios serían acceso unificado a catálogos de diferentes proveedores de datos (a través de un portal Web único), o elaboración de bases de datos sobre temas específicos, etc.

El intercambio de mensajes entre el Proveedor de Datos (PD) y el programa recolector (harvester) del Proveedor de Servicios (PS), para la transferencia de metadatos es unidireccional. El PS hace solicitudes al PD, el cual responde enviando metadatos. Las solicitudes del PS son realizadas a través del protocolo HTTP[17], usando mandos codificados a través de los métodos GET o POST. Dichas solicitudes constan de una lista de opciones con la forma de pares del tipo: clave=valor. Las peticiones que un cliente puede hacer a un proveedor son:

- GetRecord. Utilizado para recuperar un registro concreto. Necesita dos argumentos: identificador del registro pedido y especificación del formato bibliográfico en que se debe devolver.
- Identify. Utilizado para recuperar información sobre el servidor: nombre, versión del protocolo que utiliza, dirección del administrador, etc.
- ListIdentifiers. Recupera los encabezamientos de los registros, en lugar de los registros completos. Permite argumentos como el rango de fechas entre los que queremos recuperar los datos.
- ListRecords. Igual que el anterior pero recupera los registros completos.
- ListSets. Recupera un conjunto de registros. Estos conjuntos son creados opcionalmente por el servidor para facilitar una recuperación selectiva de los registros. Sería una clasificación de los contenidos según diferentes entradas. Un cliente puede pedir que se recuperen solo los registros pertenecientes a una determinada clase. Los conjuntos pueden ser simples listas o estructuras jerárquicas.
- ListMetadataFormats. Devuelve la lista de formatos bibliográficos que utiliza el servidor.

Las solicitudes son respondidas por el PD a través del envío de los metadatos de los documentos almacenados, codificados en XML. El OAI-PMH establece que el PD debe

[17] Acrónimo de Hypertext Transfer Protocol.

publicar sus metadatos en un formato Dublin Core no calificado. Por su parte, el PD puede ofrecer otros formatos de metadatos más complejos, como el MARC.

Las respuestas del servidor estarán formateadas según el protocolo HTTP con los encabezamientos adecuados. Serán documentos XML correctos que se podrán validar contra el esquema definido en el protocolo. Un ejemplo de petición y respuesta utilizando el protocolo OAI-PMH, es presentado en la Figura 17.

```
Petición:
http://an.oa.org/OAI-script?
 verb=GetRecord&identifier=oai:arXiv:hep-th/9901001&metadataPrefix=oai_dc

Respuesta:
<?xml version="1.0" encoding="UTF-8" ?>
<OAI-PMH xmlns="http://www.openarchives.org/OAI/2.0/"
         xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
         xsi:schemaLocation="http://www.openarchives.org/OAI/2.0/
         http://www.openarchives.org/OAI/2.0/OAI-PMH.xsd">
 <responseDate>2002-05-01T19:20:30Z</responseDate>
 <request verb="GetRecord" identifier="oai:arXiv:hep-th/9901001"
          metadataPrefix="oai_dc">http://an.oa.org/OAI-script</request>
 <GetRecord>
   <record>
      <header>
       <identifier>oai:arXiv:cs/0112017</identifier>
   <datestamp>2001-12-14</datestamp>
   <setSpec>cs</setSpec>
   <setSpec>math</setSpec>
     </header>
<metadata>
      <oai_dc:dc
         xmlns:oai_dc="http://www.openarchives.org/OAI/2.0/oai_dc/"
         xmlns:dc="http://purl.org/dc/elements/1.1/"
         xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
         xsi:schemaLocation="http://www.openarchives.org/OAI/2.0/oai_dc/
         http://www.openarchives.org/OAI/2.0/oai_dc.xsd">
        <dc:title>Using Structural Metadata to Localize Experience of Digital
                  Content</dc:title>
        <dc:creator>Dushay, Naomi</dc:creator>
        <dc:subject>Digital Libraries</dc:subject>
        <dc:description>With the increasing technical sophistication of
            both information consumers and providers, there is
            increasing demand for more meaningful experiences of digital
            information. We present a framework that separates digital
            object experience, or rendering, from digital object storage
            and manipulation, so the rendering can be tailored to
            particular communities of users.
        </dc:description>
        <dc:description>Comment: 23 pages including 2 appendices,
            8 figures</dc:description>
        <dc:date>2001-12-14</dc:date>
      </oai_dc:dc>
    </metadata>
</record>
 </GetRecord>
</OAI-PMH>
```

Figura 17. Formato de petición y respuesta utilizando el protocolo OAI-PMH

B. 8 Resumen

En el presente apéndice se trató información referente a los estándares y especificaciones de integración de componentes heterogéneos. Debido a que existe cierta dificultad para poder entender la diferencia entre estándar y especificación, se mencionan características de cada uno de ellos y se concluye que el estándar parte de una especificación, pero que una especificación puede ser considerada como estándar con ciertas características.

Se hizo una mención sobre los diferentes grupos de investigación y desarrollo que están trabajando con diferentes proyectos para lograr la interoperabilidad en distintos niveles. Se tomó especial interés en los estándares de metadatos IEEE LOM y Dublin Core, debido a que los mismos son de importancia trascendente para el desarrollo de la investigación presentada en este documento.

El estándar IEEE LOM especifica la sintaxis y semántica de los metadatos de un objeto de aprendizaje, es decir, describe las características relevantes del objeto de aprendizaje al que hace referencia las cuales pueden ser agrupadas en ciertas categorías: General, Lifecycle, Meta-Metadata, Technical, Educational, Rights, Relation, Annotation y Classification. Cada una de estas características cuenta con propiedades que permiten describir, de una mejor manera, a un objeto de aprendizaje en específico.

Otro estándar de metadatos mencionado es el Dublin Core el cual es un sistema de quince definiciones descriptivas, las cuales son opcionales, se pueden repetir y pueden aparecer en cualquier orden. Las definiciones mencionadas proveen un vocabulario que otorga información descriptiva de cualquier recurso. Dichas definiciones son: Contributor, Coverage, Creator, Date, Description, Format, Identifier, Language, Publisher, Relation, Rights, Source, Subject, Title y Type; de igual manera que el estándar LOM, las quince definiciones de Dublin Core presentan ciertas propiedades para definir de una manera mejor y correcta a un recurso en específico con el propósito de identificar y darlo a conocer a quien lo requiera.

Por otra parte, en este capitulo también se mencionan protocolos de comunicación de repositorios, estos protocolos son necesarios para el intercambio de información entre fuentes heterogéneas. Los protocolos mencionados aquí son el Z39.50, o simplemente protocolo Z; y el protocolo OAI-PMH. La razón de hacer una mención detallada de los mismos es debida a que estos protocolos se adecuan de buena manera a las necesidades que se pretender satisfacer en este trabajo.

El protocolo Z, es un protocolo para búsqueda y recuperación de registros de una base de datos; dicha búsqueda y recuperación se hace mediante una asociación cliente-servidor (llamada asociación Z). El objetivo de este estándar es facilitar la interconexión de clientes y servidores para aplicaciones donde los clientes busquen y recuperen información de servidores de bases de datos.

El OAI-PMH es un protocolo que proporciona interoperabilidad no inmediata entre fuentes que pueden ser documentos electrónicos, bibliotecas digitales o cualquier servidor que quiera hacer visibles los metadatos de los documentos que tiene almacenados para un sistema que los requiera.

El OAI-PMH define el intercambio de solicitudes y metadatos entre el servidor de documentos digitales y un programa recolector externo; proporciona un entorno de interoperabilidad independiente de la aplicación basado en la recolección de metadatos.

El protocolo funciona mediante solicitudes realizadas a través del protocolo HTTP usando mandos codificados a través de los métodos GET o POST del mismo; dichas solicitudes, en el ámbito del protocolo, son conocidas como verbos, los cuales son: GetRecord, Identify, ListIdentifiers, ListRecords, ListSets y ListMetadataFormats. Mediante estas solicitudes, se puede acceder a los metadatos alojados en un servidor, los cuales son entregados en formato Dublin Core, codificados en XML y formateados utilizando el protocolo HTTP.

Apéndice C

Tablas y Gráficas de Precisión y Exhaustividad

Un fragmento de las medidas de Precisión y Exhaustividad es presentado en este anexo, con la finalidad de mostrar que las consultas realizadas, resultaron en respuestas que pueden ser analizadas de manera similar. Esto es, los mecanismos implementados de recuperación de metadatos están enfocados en la recuperación de documentos relevantes, los cuales se encuentran ubicados en los primeros lugares de la lista resultante.

Lo anterior puede ser interpretado de manera tal que, para las consultas realizadas, siempre se recuperaron documentos relevantes en menor o mayor proporción; es decir, dependiendo de la cadena de búsqueda, se encontraban tantos metadatos en los cuales la anterior, ocurriera una mayor cantidad de veces.

Las gráficas en cuestión, fueron seleccionadas con un criterio aleatorio, es decir, no existe algún patrón de seguimiento entre las mismas, esto con la finalidad de dar a entender que los resultados obtenidos son similares así como las conclusiones sobre las cadenas de búsqueda creadas.

A continuación se presentan las gráficas mencionadas con su respectivo título, el cual es de cierta manera evidente, así como los correspondientes valores con los cuales se formaron las respectivas gráficas donde estos datos fueron obtenidos en base a las consultas realizadas permitiendo su comprensión general.

En la Tabla IX se pueden apreciar la cantidad de metadatos que aloja cada colección, así mismo se presentan los temas de los que constan los ya mencionados metadatos. Cada gráfica está acompañada con sus respectivos datos, los cuales fueron tomados para la realización de las ya mencionadas gráficas. La Tabla X contiene todas las consultas creadas, así como la cantidad de metadatos recuperados y la cantidad de metadatos relevantes.

Tabla IX. Cantidades de metadatos alojados en las colecciones categorizados por tema.

Tema	Colección																				
I	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**	**Metadatos totales**
0	23	40	30	29	24	28	29	29	32	31	30	28	22	32	33	33	19	23	34	29	**578**
1	33	48	30	36	21	29	29	26	30	38	28	31	24	32	32	38	37	32	37	24	**635**
2	28	34	34	28	28	36	26	28	16	49	33	33	33	30	29	31	40	31	28	29	**624**
3	27	43	29	33	22	34	27	25	28	37	21	39	30	33	32	48	37	27	29	26	**627**
4	26	25	27	35	21	47	20	23	24	34	29	33	30	27	31	33	29	32	39	22	**587**
5	28	34	29	31	24	28	25	25	25	39	35	29	26	32	43	33	31	34	25	26	**602**
6	28	38	28	35	25	32	28	19	21	35	25	29	24	34	36	34	33	19	26	17	**566**
7	20	34	35	25	19	35	33	32	20	36	30	45	25	40	16	21	36	30	35	29	**596**
8	20	34	37	35	24	36	25	28	26	46	27	26	25	30	27	32	39	24	35	34	**610**
9	21	33	39	27	20	42	26	24	25	39	23	25	28	29	33	35	41	27	34	28	**599**
10	25	29	21	25	26	29	26	9	22	32	31	31	30	24	36	28	43	18	34	35	**554**
11	22	29	24	27	21	32	31	26	29	44	32	30	27	35	34	31	36	24	37	27	**598**
12	18	35	41	30	27	33	21	27	28	33	31	31	29	38	38	30	31	24	24	16	**585**
13	30	30	32	29	25	23	36	21	25	25	20	37	26	27	37	26	35	22	39	32	**577**
14	28	46	32	30	33	41	35	21	26	22	32	39	26	30	28	37	40	34	28	36	**644**
15	24	43	29	42	19	42	27	29	25	39	20	27	28	34	35	42	38	28	31	32	**634**
16	27	24	34	40	25	31	23	26	25	21	37	18	25	31	26	45	35	23	30	26	**572**
17	26	31	36	25	23	33	29	27	30	31	31	22	31	21	46	27	30	29	34	26	**588**
18	19	31	23	31	27	37	27	32	25	32	22	30	16	27	30	32	34	28	38	23	**564**
19	32	27	22	32	28	36	36	19	28	40	23	28	27	36	26	31	26	33	31	17	**578**
20	28	33	40	39	30	29	37	21	22	30	25	28	20	22	27	35	25	30	42	23	**586**
21	25	28	31	34	20	36	28	15	22	30	25	45	26	45	30	29	34	26	32	24	**585**
22	22	55	15	23	27	25	26	28	22	30	21	33	20	32	30	27	30	29	34	27	**556**
23	23	30	34	32	22	43	45	17	14	36	26	38	20	37	25	29	32	29	48	27	**607**

Tabla IX. Cantidades de metadatos alojados en las colecciones categorizados por tema. (Continuación)

24	20	33	47	41	17	25	26	29	22	32	30	25	26	37	32	36	34	24	25	21	**582**
25	14	29	33	37	30	35	32	17	25	44	33	36	32	27	26	28	34	19	38	24	**593**
26	20	34	27	35	24	34	25	25	22	33	29	20	28	24	24	38	33	34	24	21	**554**
27	20	39	36	42	25	31	25	20	27	33	22	25	21	31	27	36	32	31	36	24	**583**
28	30	33	26	30	18	42	30	21	30	30	29	34	23	28	34	50	37	25	35	25	**610**
29	16	39	21	30	32	33	29	21	25	37	26	28	26	33	35	29	39	25	31	22	**577**
30	18	35	28	29	23	29	35	23	27	36	19	26	22	33	32	37	29	35	41	17	**574**
31	21	25	25	35	32	27	26	28	33	35	25	28	23	27	24	37	39	28	31	26	**575**
32	16	46	34	34	21	27	39	30	32	29	27	20	27	32	32	33	28	28	28	26	**589**
33	31	26	26	29	23	31	23	22	20	37	23	24	18	29	31	29	32	16	35	26	**531**
34	19	35	33	27	27	31	25	32	29	34	33	28	24	34	24	34	28	25	28	19	**569**
35	19	33	21	24	25	43	26	20	26	32	28	37	33	28	31	37	29	25	36	30	**583**
36	19	23	32	25	28	46	25	18	29	27	25	38	29	37	28	31	29	26	27	30	**572**
37	25	34	22	43	27	43	21	27	30	44	26	35	22	39	38	36	28	23	33	16	**612**
38	20	30	25	20	34	45	24	26	28	34	26	26	27	35	36	33	48	35	24	35	**611**
39	30	25	31	36	26	19	22	33	23	34	28	29	22	37	27	38	37	29	21	31	**578**
40	28	25	22	34	26	33	38	23	24	36	39	23	21	28	37	34	23	27	30	37	**588**
41	16	41	25	27	27	39	33	22	21	31	30	34	22	33	32	35	34	24	38	24	**588**
42	19	29	24	36	22	29	30	19	26	29	37	31	31	37	31	38	40	26	41	19	**594**
Metadatos totales	**1004**	**1448**	**1270**	**1367**	**1068**	**1459**	**1229**	**1033**	**1089**	**1476**	**1192**	**1302**	**1095**	**1367**	**1341**	**1456**	**1444**	**1161**	**1406**	**1108**	**25315**

Tabla X. Consultas creadas, cantidad de metadatos recuperados y relevantes, tipo de consulta y valores de Precisión y Exhaustividad para cada una de las consultas creadas.

Tema	Consulta	Corta = C Larga = L	Metadatos recuperados	Metadatos relevantes	Precisión	Exhaustividad
0	1	C	1024	578	1	0.564453125
0	2	L	1024	103	0.178200692	0.100585938
0	3	C	1024	23	0.039792388	0.022460938
0	4	C	1024	24	0.041522491	0.0234375
0	5	C	1024	68	0.117647059	0.06640625
0	6	L	1024	16	0.027681661	0.015625
0	7	C	1024	124	0.214532872	0.12109375
0	8	C	1024	308	0.532871972	0.30078125
0	9	C	1024	291	0.503460208	0.284179688
0	10	L	1024	350	0.605536332	0.341796875
1	11	C	1024	635	1	0.620117188
1	12	C	1024	496	0.781102362	0.484375
1	13	C	1024	635	1	0.620117188
1	14	C	1024	635	1	0.620117188
2	15	C	1024	161	0.258012821	0.157226563
2	16	C	1024	176	0.282051282	0.171875
2	17	L	1024	56	0.08974359	0.0546875
2	18	C	1024	284	0.455128205	0.27734375
2	19	C	1024	624	1	0.609375
2	20	L	1024	50	0.080128205	0.048828125
2	21	C	1024	258	0.413461538	0.251953125
2	22	C	1024	176	0.282051282	0.171875
2	23	L	1024	19	0.030448718	0.018554688
3	24	C	1024	298	0.515570934	0.291015625
3	25	C	1024	324	0.560553633	0.31640625
3	26	C	1024	324	0.560553633	0.31640625
3	27	C	1024	324	0.560553633	0.31640625
3	28	L	1024	68	0.117647059	0.06640625
3	29	L	1024	10	0.017301038	0.009765625
3	30	L	1024	20	0.034602076	0.01953125
4	31	C	1024	500	0.851788756	0.48828125
4	32	C	1024	404	0.688245315	0.39453125
4	33	L	1024	56	0.095400341	0.0546875
4	34	C	1024	260	0.442930153	0.25390625
4	35	C	1024	186	0.316865417	0.181640625
4	36	L	1024	20	0.03407155	0.01953125
5	37	C	1024	602	1	0.587890625
5	38	C	1024	602	1	0.587890625
5	39	C	1024	602	1	0.587890625
5	40	C	1024	602	1	0.587890625
5	41	C	1024	602	1	0.587890625

Tabla X. Consultas creadas, cantidad de metadatos recuperados y relevantes, tipo de consulta y valores de Precisión y Exhaustividad para cada una de las consultas creadas. (Continuación)

5	42	C	1024	602	1	0.587890625
6	43	C	1024	259	0.457597173	0.252929688
6	44	C	1024	259	0.457597173	0.252929688
6	45	C	1024	566	1	0.552734375
6	46	L	1024	29	0.051236749	0.028320313
6	47	C	1024	306	0.540636042	0.298828125
6	48	C	1024	566	1	0.552734375
6	49	C	1024	269	0.475265018	0.262695313
6	50	C	1024	566	1	0.552734375
6	51	L	1024	16	0.028268551	0.015625
6	52	C	1024	29	0.051236749	0.028320313
7	53	C	1024	236	0.395973154	0.23046875
7	54	C	1024	596	1	0.58203125
7	55	C	1024	306	0.513422819	0.298828125
7	56	C	1024	254	0.426174497	0.248046875
7	57	L	1024	406	0.681208054	0.396484375
7	58	C	1024	596	1	0.58203125
7	59	L	1024	29	0.048657718	0.028320313
7	60	C	1024	104	0.174496644	0.1015625
7	61	L	1024	56	0.093959732	0.0546875
8	62	C	1024	610	1	0.595703125
8	63	C	1024	63	0.103278689	0.061523438
8	64	L	1024	54	0.08852459	0.052734375
8	65	C	1024	610	1	0.595703125
8	66	L	1024	31	0.050819672	0.030273438
8	67	C	1024	189	0.309836066	0.184570313
8	68	C	1024	245	0.401639344	0.239257813
8	69	L	1024	95	0.155737705	0.092773438
8	70	L	1024	89	0.145901639	0.086914063
8	71	C	1024	398	0.652459016	0.388671875
8	72	C	1024	34	0.055737705	0.033203125
8	73	L	1024	45	0.073770492	0.043945313
8	74	L	1024	35	0.057377049	0.034179688
9	75	C	1024	599	1	0.584960938
9	76	C	1024	599	1	0.584960938
9	77	L	1024	102	0.170283806	0.099609375
9	78	L	1024	78	0.130217028	0.076171875
9	79	L	1024	89	0.148580968	0.086914063
9	80	C	1024	599	1	0.584960938
9	81	C	1024	216	0.360601002	0.2109375
9	82	C	1024	599	1	0.584960938
9	83	C	1024	599	1	0.584960938
9	84	L	1024	35	0.058430718	0.034179688
9	85	L	1024	38	0.063439065	0.037109375

Tabla X. Consultas creadas, cantidad de metadatos recuperados y relevantes, tipo de consulta y valores de Precisión y Exhaustividad para cada una de las consultas creadas. (Continuación)

10	86	L	1024	102	0.184115523	0.099609375
10	87	L	1024	156	0.281588448	0.15234375
10	88	L	1024	57	0.102888087	0.055664063
10	89	L	1024	35	0.063176895	0.034179688
10	90	C	1024	256	0.462093863	0.25
10	91	C	1024	198	0.357400722	0.193359375
10	92	L	1024	169	0.305054152	0.165039063
10	93	C	1024	254	0.458483755	0.248046875
10	94	L	1024	25	0.045126354	0.024414063
10	95	C	1024	554	1	0.541015625
10	96	C	1024	554	1	0.541015625
10	97	L	1024	96	0.173285199	0.09375
10	98	C	1024	554	1	0.541015625
10	99	L	1024	26	0.046931408	0.025390625
10	100	L	1024	35	0.063176895	0.034179688
11	101	C	1024	280	0.468227425	0.2734375
11	102	L	1024	58	0.096989967	0.056640625
11	103	L	1024	68	0.113712375	0.06640625
11	104	L	1024	35	0.058528428	0.034179688
11	105	L	1024	153	0.255852843	0.149414063
11	106	L	1024	89	0.148829431	0.086914063
11	107	C	1024	289	0.483277592	0.282226563
11	108	L	1024	45	0.075250836	0.043945313
11	109	L	1024	35	0.058528428	0.034179688
11	110	L	1024	68	0.113712375	0.06640625
11	111	C	1024	390	0.652173913	0.380859375
11	112	L	1024	56	0.093645485	0.0546875
11	113	C	1024	226	0.377926421	0.220703125
11	114	L	1024	230	0.384615385	0.224609375
11	115	L	1024	56	0.093645485	0.0546875
12	116	C	1024	102	0.174358974	0.099609375
12	117	L	1024	65	0.111111111	0.063476563
12	118	C	1024	26	0.044444444	0.025390625
12	119	C	1024	290	0.495726496	0.283203125
12	120	C	1024	268	0.458119658	0.26171875
12	121	C	1024	269	0.45982906	0.262695313
12	122	L	1024	190	0.324786325	0.185546875
12	123	L	1024	16	0.027350427	0.015625
12	124	L	1024	26	0.044444444	0.025390625
12	125	L	1024	19	0.032478632	0.018554688
12	126	L	1024	59	0.100854701	0.057617188
12	127	L	1024	69	0.117948718	0.067382813
12	128	L	1024	98	0.167521368	0.095703125
12	129	C	1024	585	1	0.571289063

Tabla X. Consultas creadas, cantidad de metadatos recuperados y relevantes, tipo de consulta y valores de Precisión y Exhaustividad para cada una de las consultas creadas. (Continuación)

13	130	C	1024	577	1	0.563476563
13	131	L	1024	56	0.097053726	0.0546875
13	132	L	1024	23	0.039861352	0.022460938
13	133	C	1024	577	1	0.563476563
13	134	L	1024	65	0.112651646	0.063476563
13	135	C	1024	189	0.327556326	0.184570313
13	136	C	1024	205	0.355285962	0.200195313
13	137	L	1024	20	0.034662045	0.01953125
13	138	C	1024	190	0.329289428	0.185546875
13	139	L	1024	16	0.027729636	0.015625
13	140	L	1024	20	0.034662045	0.01953125
13	141	L	1024	29	0.050259965	0.028320313
13	142	L	1024	31	0.05372617	0.030273438
14	143	C	1024	250	0.388198758	0.244140625
14	144	C	1024	395	0.613354037	0.385742188
14	145	C	1024	644	1	0.62890625
14	146	L	1024	39	0.060559006	0.038085938
14	147	L	1024	40	0.062111801	0.0390625
14	148	L	1024	59	0.091614907	0.057617188
14	149	C	1024	365	0.566770186	0.356445313
14	150	L	1024	29	0.045031056	0.028320313
14	151	C	1024	256	0.397515528	0.25
14	152	C	1024	644	1	0.62890625
14	153	C	1024	644	1	0.62890625
14	154	L	1024	30	0.046583851	0.029296875
14	155	L	1024	59	0.091614907	0.057617188
14	156	C	1024	644	1	0.62890625
14	157	L	1024	21	0.032608696	0.020507813
15	158	L	1024	125	0.197160883	0.122070313
15	159	L	1024	95	0.149842271	0.092773438
15	160	L	1024	98	0.154574132	0.095703125
15	161	L	1024	69	0.108832808	0.067382813
15	162	L	1024	56	0.088328076	0.0546875
16	163	C	1024	572	1	0.55859375
16	164	C	1024	572	1	0.55859375
16	165	L	1024	150	0.262237762	0.146484375
16	166	C	1024	190	0.332167832	0.185546875
16	167	C	1024	190	0.332167832	0.185546875
16	168	C	1024	309	0.54020979	0.301757813
16	169	L	1024	29	0.050699301	0.028320313
16	170	C	1024	190	0.332167832	0.185546875
16	171	L	1024	25	0.043706294	0.024414063
16	172	L	1024	63	0.11013986	0.061523438
16	173	L	1024	59	0.103146853	0.057617188

Tabla X. Consultas creadas, cantidad de metadatos recuperados y relevantes, tipo de consulta y valores de Precisión y Exhaustividad para cada una de las consultas creadas. (Continuación)

17	174	C	1024	588	1	0.57421875
17	175	L	1024	10	0.017006803	0.009765625
17	176	L	1024	106	0.180272109	0.103515625
17	177	L	1024	106	0.180272109	0.103515625
17	178	L	1024	25	0.042517007	0.024414063
17	179	C	1024	588	1	0.57421875
17	180	C	1024	155	0.263605442	0.151367188
17	181	L	1024	26	0.044217687	0.025390625
17	182	C	1024	588	1	0.57421875
17	183	L	1024	45	0.076530612	0.043945313
17	184	C	1024	588	1	0.57421875
17	185	C	1024	155	0.263605442	0.151367188
17	186	L	1024	35	0.05952381	0.034179688
17	187	L	1024	22	0.037414966	0.021484375
18	188	L	1024	15	0.026595745	0.014648438
18	189	L	1024	26	0.046099291	0.025390625
18	190	L	1024	35	0.062056738	0.034179688
18	191	L	1024	26	0.046099291	0.025390625
18	192	C	1024	564	1	0.55078125
18	193	L	1024	68	0.120567376	0.06640625
18	194	L	1024	65	0.115248227	0.063476563
18	195	L	1024	84	0.14893617	0.08203125
18	196	L	1024	25	0.044326241	0.024414063
19	197	C	1024	578	1	0.564453125
19	198	C	1024	326	0.564013841	0.318359375
19	199	L	1024	15	0.025951557	0.014648438
19	200	C	1024	578	1	0.564453125
19	201	L	1024	23	0.039792388	0.022460938
19	202	C	1024	578	1	0.564453125
19	203	L	1024	26	0.044982699	0.025390625
19	204	C	1024	578	1	0.564453125
19	205	L	1024	21	0.03633218	0.020507813
19	206	L	1024	20	0.034602076	0.01953125
19	207	L	1024	32	0.055363322	0.03125
19	208	L	1024	16	0.027681661	0.015625
20	209	C	1024	295	0.503412969	0.288085938
20	210	C	1024	586	1	0.572265625
20	211	C	1024	195	0.332764505	0.190429688
20	212	C	1024	586	1	0.572265625
21	213	C	1024	585	1	0.571289063
21	214	L	1024	25	0.042735043	0.024414063
21	215	C	1024	585	1	0.571289063
21	216	C	1024	268	0.458119658	0.26171875
21	217	C	1024	268	0.458119658	0.26171875

Tabla X. Consultas creadas, cantidad de metadatos recuperados y relevantes, tipo de consulta y valores de Precisión y Exhaustividad para cada una de las consultas creadas. (Continuación)

22	218	C	1024	556	1	0.54296875
22	219	C	1024	556	1	0.54296875
22	220	C	1024	201	0.361510791	0.196289063
22	221	L	1024	23	0.041366906	0.022460938
22	222	C	1024	286	0.514388489	0.279296875
22	223	L	1024	12	0.021582734	0.01171875
23	224	L	1024	23	0.037891269	0.022460938
23	225	L	1024	25	0.041186161	0.024414063
23	226	C	1024	260	0.428336079	0.25390625
23	227	L	1024	35	0.057660626	0.034179688
24	228	C	1024	289	0.496563574	0.282226563
24	229	C	1024	289	0.496563574	0.282226563
24	230	C	1024	295	0.506872852	0.288085938
24	231	L	1024	15	0.025773196	0.014648438
24	232	L	1024	23	0.0395189	0.022460938
24	233	C	1024	582	1	0.568359375
24	234	L	1024	25	0.042955326	0.024414063
25	235	C	1024	152	0.256323777	0.1484375
25	236	C	1024	196	0.330522766	0.19140625
25	237	L	1024	580	0.978077572	0.56640625
25	238	C	1024	250	0.42158516	0.244140625
25	239	L	1024	96	0.161888702	0.09375
25	240	C	1024	344	0.58010118	0.3359375
26	241	C	1024	400	0.722021661	0.390625
26	242	L	1024	500	0.902527076	0.48828125
27	243	C	1024	583	1	0.569335938
27	244	C	1024	583	1	0.569335938
27	245	C	1024	583	1	0.569335938
27	246	C	1024	583	1	0.569335938
27	247	C	1024	583	1	0.569335938
27	248	C	1024	583	1	0.569335938
27	249	C	1024	583	1	0.569335938
27	250	C	1024	15	0.025728988	0.014648438
27	251	C	1024	583	1	0.569335938
27	252	C	1024	490	0.840480274	0.478515625
28	253	C	1024	294	0.481967213	0.287109375
28	254	L	1024	25	0.040983607	0.024414063
28	255	C	1024	250	0.409836066	0.244140625
28	256	C	1024	610	1	0.595703125
28	257	C	1024	190	0.31147541	0.185546875
28	258	L	1024	25	0.040983607	0.024414063
28	259	C	1024	294	0.481967213	0.287109375
28	260	L	1024	25	0.040983607	0.024414063
28	261	L	1024	26	0.042622951	0.025390625

Tabla X. Consultas creadas, cantidad de metadatos recuperados y relevantes, tipo de consulta y valores de Precisión y Exhaustividad para cada una de las consultas creadas. (Continuación)

29	262	C	1024	296	0.512998267	0.2890625
29	263	L	1024	25	0.043327556	0.024414063
29	264	C	1024	295	0.511265165	0.288085938
29	265	L	1024	24	0.041594454	0.0234375
29	266	C	1024	577	1	0.563476563
29	267	C	1024	577	1	0.563476563
29	268	C	1024	490	0.849220104	0.478515625
29	269	L	1024	24	0.041594454	0.0234375
29	270	C	1024	240	0.415944541	0.234375
29	271	C	1024	577	1	0.563476563
29	272	L	1024	24	0.041594454	0.0234375
29	273	C	1024	206	0.357019064	0.201171875
29	274	L	1024	15	0.025996534	0.014648438
29	275	C	1024	577	1	0.563476563
30	276	C	1024	24	0.041811847	0.0234375
30	277	C	1024	356	0.620209059	0.34765625
30	278	C	1024	574	1	0.560546875
30	279	C	1024	274	0.477351916	0.267578125
30	280	C	1024	286	0.49825784	0.279296875
30	281	C	1024	286	0.49825784	0.279296875
31	282	L	1024	39	0.067826087	0.038085938
31	283	C	1024	168	0.292173913	0.1640625
31	284	L	1024	35	0.060869565	0.034179688
31	285	C	1024	168	0.292173913	0.1640625
31	286	L	1024	21	0.036521739	0.020507813
31	287	C	1024	32	0.055652174	0.03125
31	288	L	1024	24	0.04173913	0.0234375
32	289	L	1024	68	0.115449915	0.06640625
32	290	C	1024	98	0.166383701	0.095703125
32	291	C	1024	269	0.456706282	0.262695313
32	292	C	1024	109	0.185059423	0.106445313
32	293	C	1024	249	0.422750424	0.243164063
32	294	L	1024	130	0.220713073	0.126953125
32	295	C	1024	589	1	0.575195313
32	296	L	1024	24	0.040747029	0.0234375
32	297	L	1024	16	0.027164686	0.015625
32	298	C	1024	185	0.314091681	0.180664063
32	299	C	1024	298	0.505942275	0.291015625
33	300	L	1024	29	0.054613936	0.028320313
33	301	C	1024	531	1	0.518554688
33	302	C	1024	531	1	0.518554688
33	303	L	1024	38	0.071563089	0.037109375
33	304	C	1024	531	1	0.518554688
33	305	L	1024	36	0.06779661	0.03515625

Tabla X. Consultas creadas, cantidad de metadatos recuperados y relevantes, tipo de consulta y valores de Precisión y Exhaustividad para cada una de las consultas creadas. (Continuación)

33	306	C	1024	531	1	0.518554688
33	307	L	1024	26	0.048964218	0.025390625
33	308	C	1024	531	1	0.518554688
33	309	C	1024	531	1	0.518554688
33	310	L	1024	76	0.143126177	0.07421875
34	311	C	1024	138	0.242530756	0.134765625
34	312	L	1024	24	0.042179262	0.0234375
34	313	L	1024	26	0.0456942	0.025390625
34	314	L	1024	14	0.024604569	0.013671875
34	315	C	1024	569	1	0.555664063
34	316	L	1024	12	0.021089631	0.01171875
35	317	C	1024	197	0.337907376	0.192382813
35	318	C	1024	583	1	0.569335938
35	319	C	1024	24	0.041166381	0.0234375
35	320	C	1024	583	1	0.569335938
35	321	L	1024	25	0.042881647	0.024414063
36	322	C	1024	572	1	0.55859375
36	323	L	1024	20	0.034965035	0.01953125
36	324	L	1024	24	0.041958042	0.0234375
36	325	C	1024	572	1	0.55859375
36	326	L	1024	35	0.061188811	0.034179688
36	327	C	1024	500	0.874125874	0.48828125
36	328	C	1024	572	1	0.55859375
36	329	C	1024	572	1	0.55859375
36	330	L	1024	26	0.045454545	0.025390625
36	331	L	1024	20	0.034965035	0.01953125
36	332	C	1024	259	0.452797203	0.252929688
36	333	L	1024	21	0.036713287	0.020507813
37	334	C	1024	20	0.032679739	0.01953125
37	335	C	1024	180	0.294117647	0.17578125
37	336	L	1024	20	0.032679739	0.01953125
37	337	C	1024	358	0.58496732	0.349609375
37	338	C	1024	612	1	0.59765625
37	339	L	1024	21	0.034313725	0.020507813
38	340	L	1024	12	0.019639935	0.01171875
38	341	C	1024	611	1	0.596679688
38	342	L	1024	95	0.155482815	0.092773438
38	343	L	1024	12	0.019639935	0.01171875
38	344	C	1024	356	0.582651391	0.34765625
38	345	L	1024	20	0.032733224	0.01953125
38	346	C	1024	611	1	0.596679688
38	347	C	1024	284	0.464811784	0.27734375
39	348	L	1024	20	0.034602076	0.01953125
39	349	L	1024	16	0.027681661	0.015625

Tabla X. Consultas creadas, cantidad de metadatos recuperados y relevantes, tipo de consulta y valores de Precisión y Exhaustividad para cada una de las consultas creadas. (Continuación)

39	350	L	1024	10	0.017301038	0.009765625
39	351	L	1024	10	0.017301038	0.009765625
39	352	C	1024	187	0.323529412	0.182617188
39	353	C	1024	578	1	0.564453125
39	354	L	1024	14	0.024221453	0.013671875
39	355	C	1024	578	1	0.564453125
39	356	C	1024	578	1	0.564453125
39	357	L	1024	13	0.022491349	0.012695313
39	358	C	1024	92	0.15916955	0.08984375
39	359	C	1024	578	1	0.564453125
40	360	C	1024	588	1	0.57421875
40	361	C	1024	588	1	0.57421875
40	362	C	1024	588	1	0.57421875
40	363	C	1024	259	0.44047619	0.252929688
40	364	L	1024	20	0.034013605	0.01953125
40	365	L	1024	588	1	0.57421875
41	366	C	1024	289	0.491496599	0.282226563
41	367	L	1024	10	0.017006803	0.009765625
41	368	L	1024	16	0.027210884	0.015625
41	369	C	1024	294	0.5	0.287109375
41	370	C	1024	15	0.025510204	0.014648438
42	371	C	1024	465	0.782828283	0.454101563
42	372	L	1024	489	0.823232323	0.477539063
42	373	C	1024	389	0.654882155	0.379882813
42	374	L	1024	495	0.833333333	0.483398438
42	375	C	1024	480	0.808080808	0.46875
42	376	L	1024	540	0.909090909	0.52734375
42	377	C	1024	399	0.671717172	0.389648438
42	378	C	1024	298	0.501683502	0.291015625
42	379	L	1024	495	0.833333333	0.483398438
42	380	C	1024	368	0.61952862	0.359375
42	381	L	1024	364	0.612794613	0.35546875
42	382	C	1024	456	0.767676768	0.4453125
42	383	L	1024	480	0.808080808	0.46875

Tabla XI. Valores de Precisión y Exhaustividad referentes a las consultas del tema 3.

Consulta	Precisión	Exhaustividad
24	0.515570934	0.291015625
25	0.560553633	0.31640625
26	0.560553633	0.31640625
27	0.560553633	0.31640625
28	0.117647059	0.06640625
29	0.017301038	0.009765625
30	0.034602076	0.01953125

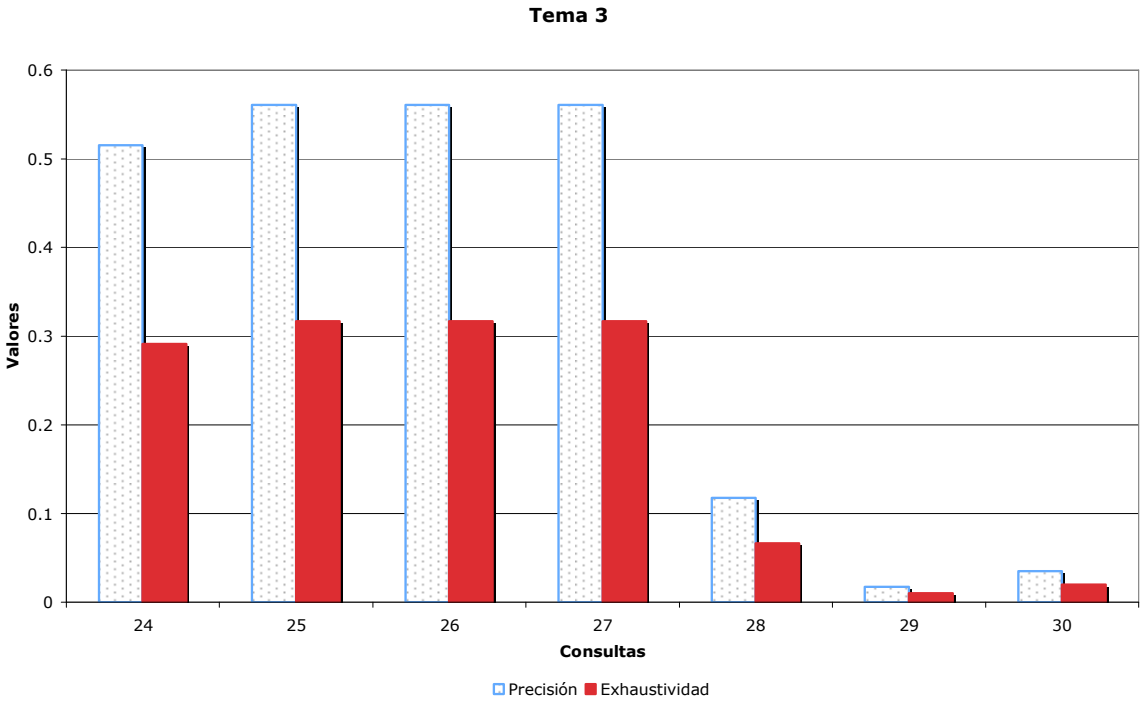

Figura 18. Comparación entre Precisión y Exhaustividad para las consultas referentes al tema 3.

Tabla XII. Valores de Precisión y Exhaustividad referentes a las consultas del tema 7.

Consulta	Precisión	Exhaustividad
53	0.395973154	0.23046875
54	1	0.58203125
55	0.513422819	0.298828125
56	0.426174497	0.248046875
57	0.681208054	0.396484375
58	1	0.58203125
59	0.048657718	0.028320313
60	0.174496644	0.1015625
61	0.093959732	0.0546875

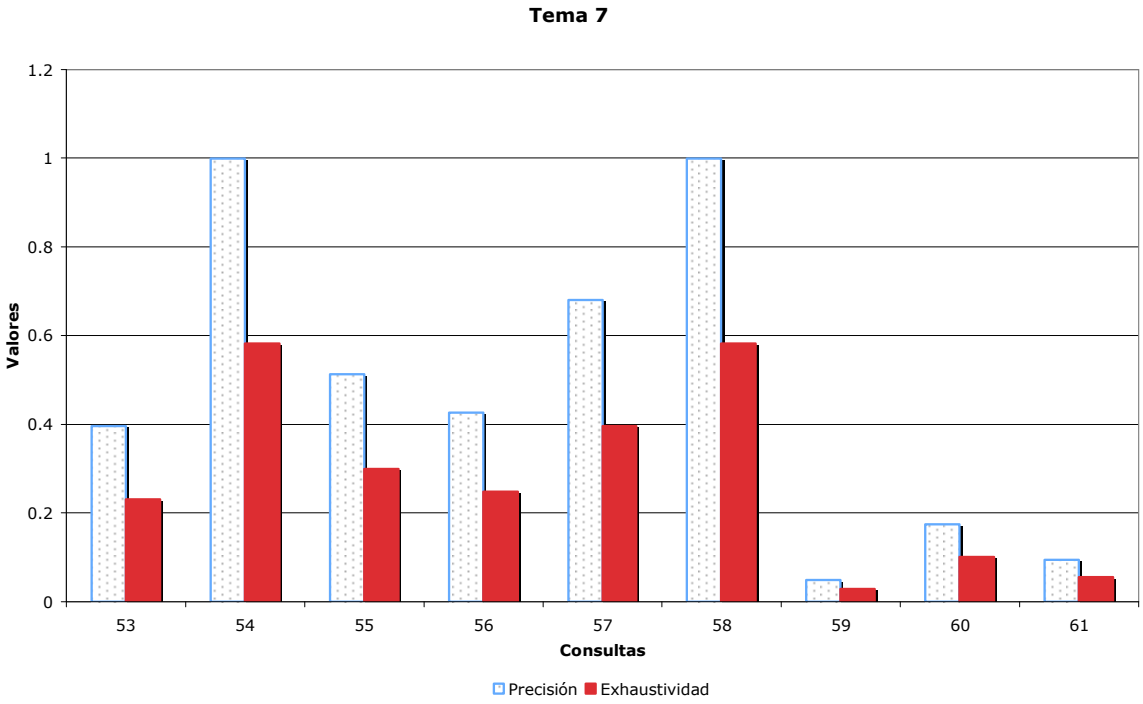

Figura 19. Comparación entre Precisión y Exhaustividad para las consultas referentes al tema 7.

Tabla XIII. Valores de Precisión y Exhaustividad referentes a las consultas del tema 12.

Consulta	Precisión	Exhaustividad
116	0.174358974	0.099609375
117	0.111111111	0.063476563
118	0.044444444	0.025390625
119	0.495726496	0.283203125
120	0.458119658	0.26171875
121	0.45982906	0.262695313
122	0.324786325	0.185546875
123	0.027350427	0.015625
124	0.044444444	0.025390625
125	0.032478632	0.018554688
126	0.100854701	0.057617188
127	0.117948718	0.067382813
128	0.167521368	0.095703125
129	1	0.571289063

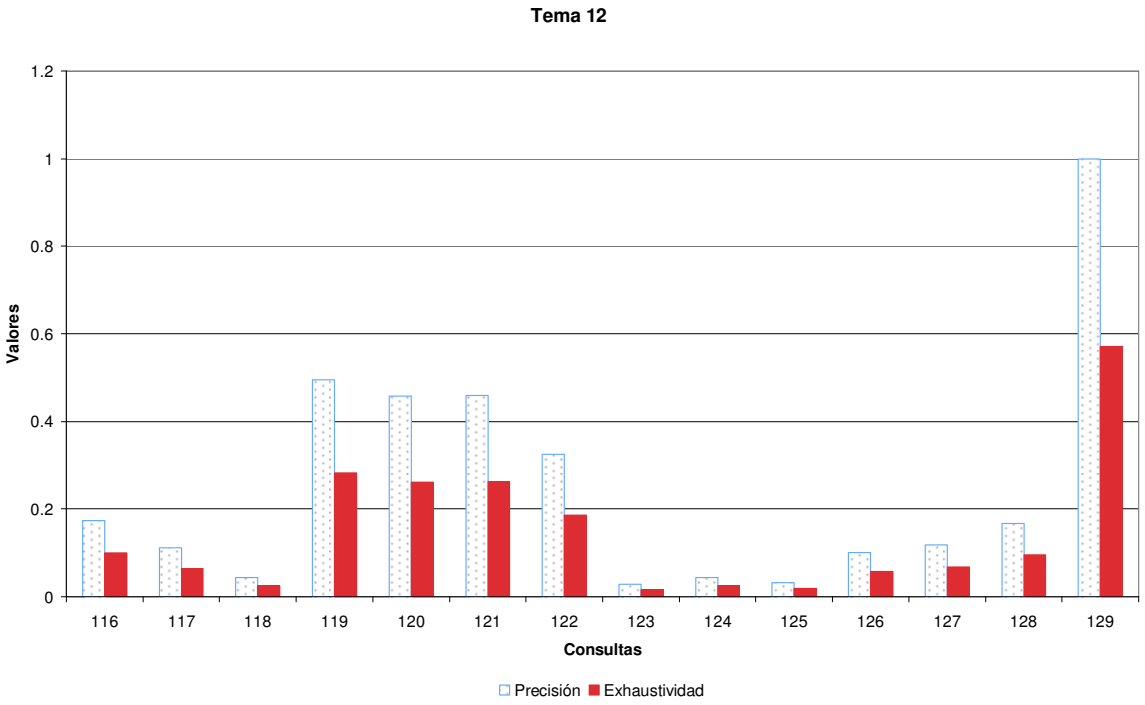

Figura 20. Comparación entre Precisión y Exhaustividad para las consultas referentes al tema 12.

Tabla XIV. Valores de Precisión y Exhaustividad referentes a las consultas del tema 17.

Consulta	Precisión	Exhaustividad
174	1	0.57421875
175	0.017006803	0.009765625
176	0.180272109	0.103515625
177	0.180272109	0.103515625
178	0.042517007	0.024414063
179	1	0.57421875
180	0.263605442	0.151367188
181	0.044217687	0.025390625
182	1	0.57421875
183	0.076530612	0.043945313
184	1	0.57421875
185	0.263605442	0.151367188
186	0.05952381	0.034179688
187	0.037414966	0.021484375

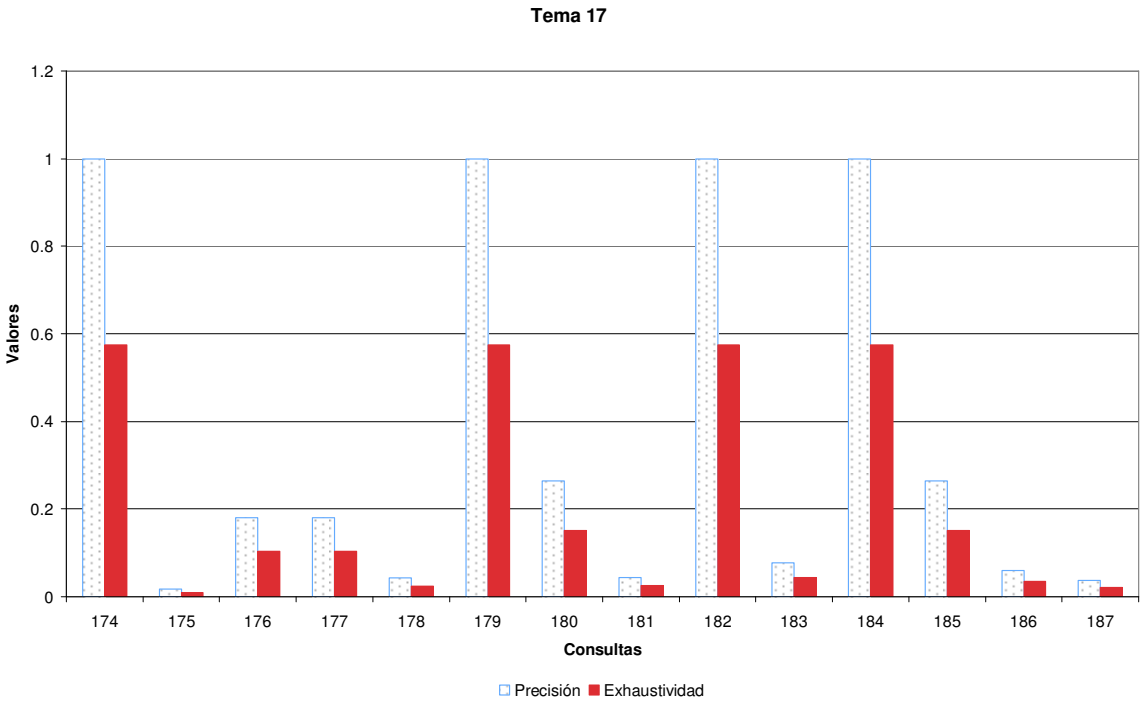

Figura 21. Comparación entre Precisión y Exhaustividad para las consultas referentes al tema 17.

Tabla XV. Valores de Precisión y Exhaustividad referentes a las consultas del tema 27.

Consulta	Precisión	Exhaustividad
243	1	0.569335938
244	1	0.569335938
245	1	0.569335938
246	1	0.569335938
247	1	0.569335938
248	1	0.569335938
249	1	0.569335938
250	0.025728988	0.014648438
251	1	0.569335938
252	0.840480274	0.478515625

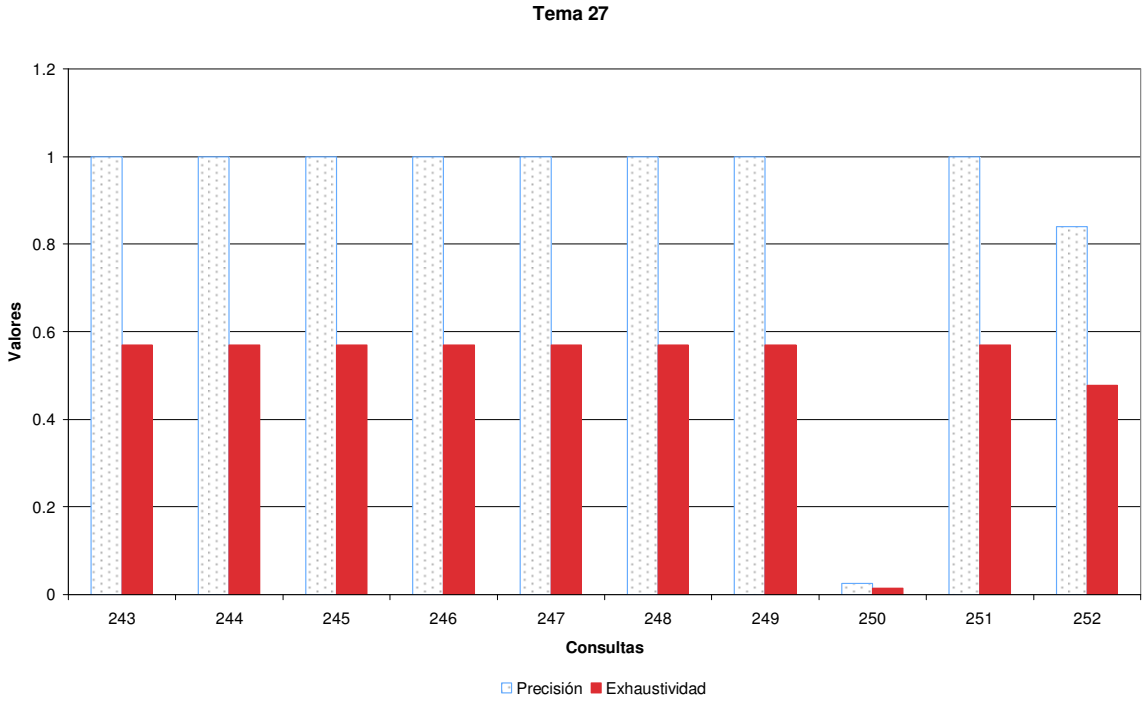

Figura 22. Comparación entre Precisión y Exhaustividad para las consultas referentes al tema 27.

Tabla XVI. Valores de Precisión y Exhaustividad referentes a las consultas del tema 31.

Consulta	Precisión	Exhaustividad
282	0.067826087	0.038085938
283	0.292173913	0.1640625
284	0.060869565	0.034179688
285	0.292173913	0.1640625
286	0.036521739	0.020507813
287	0.055652174	0.03125
288	0.04173913	0.0234375

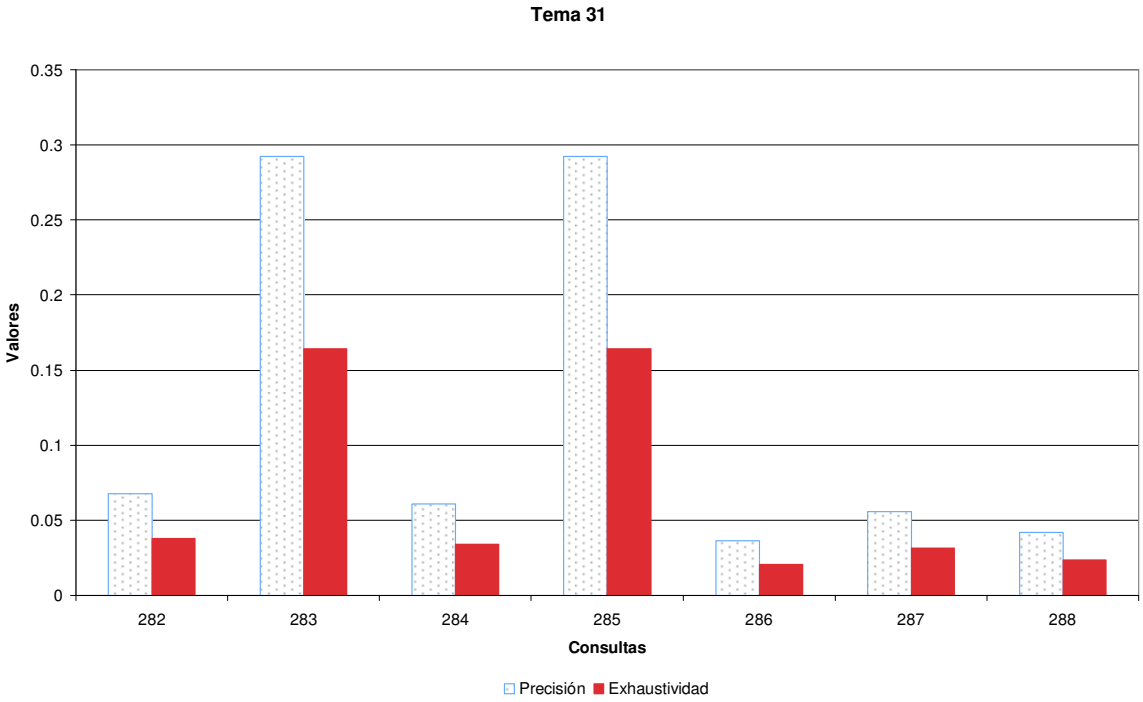

Figura 23. Comparación entre Precisión y Exhaustividad para las consultas referentes al tema 31.

Tabla XVII. Valores de Precisión y Exhaustividad referentes a las consultas del tema 42.

Consulta	Precisión	Exhaustividad
371	0.782828283	0.454101563
372	0.823232323	0.477539063
373	0.654882155	0.379882813
374	0.833333333	0.483398438
375	0.808080808	0.46875
376	0.909090909	0.52734375
377	0.671717172	0.389648438
378	0.501683502	0.291015625
379	0.833333333	0.483398438
380	0.61952862	0.359375
381	0.612794613	0.35546875
382	0.767676768	0.4453125
383	0.808080808	0.46875

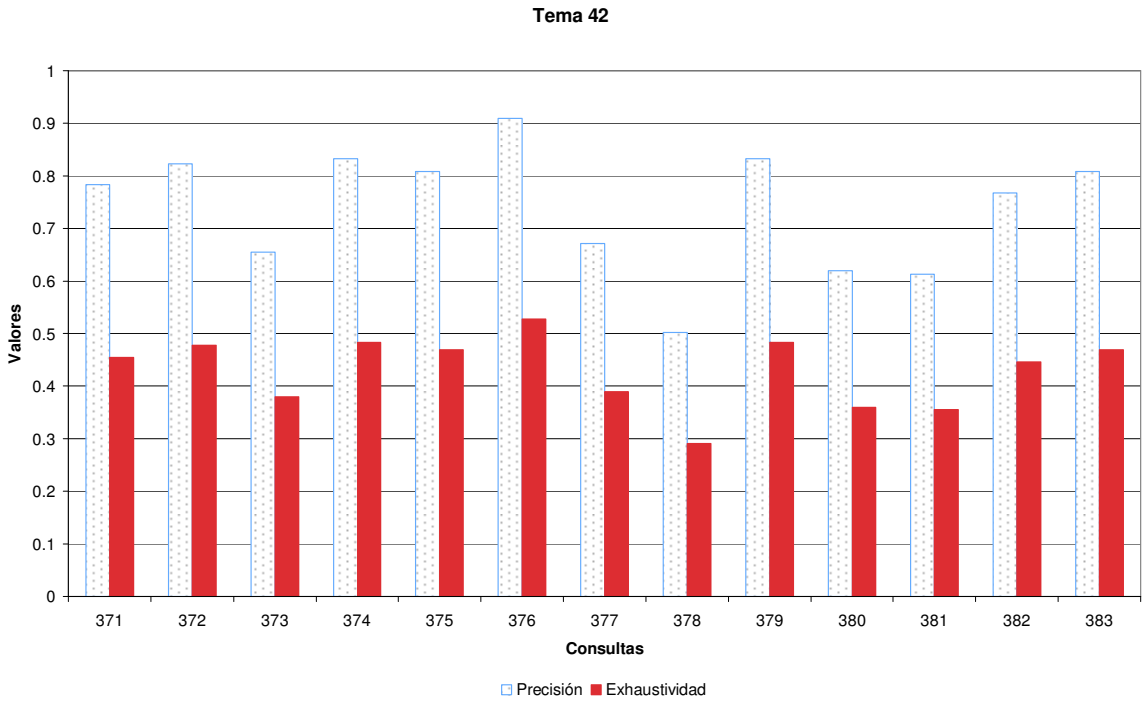

Figura 24. Comparación entre Precisión y Exhaustividad para las consultas referentes al tema 42.

En las gráficas presentadas, se puede apreciar que los niveles de Precisión siempre son elevados, lo que significa que los documentos recuperados en su respectiva consulta, son los más relevantes; sin embargo, las medidas de precisión en algunas consultas se desploman a cero (Figura 21, por ejemplo), esto es debido a que para esa consulta no fue posible encontrar metadatos que saldaran la consulta inicial, regresando un valor nulo, la causa de este valor nula es la creación errónea de la consulta, dado que no fue hecha con la ayuda de expertos en la materia o incluso por una mala escritura de la misma.

Los datos encontrados en todas las tablas, fueron obtenidos mediante la aplicación de las respectivas fórmulas de Precisión y Exhaustividad, aprovechando que la cantidad de metadatos relevantes, así como la cantidad de metadatos almacenados en cada colección, es conocida.