Integración de Colecciones Heterogéneas en Bibliotecas Digitales

M.M. Martínez*, C. E. Cuesta*, P. de la Fuente* y J. C. Lamirel+

* Universidad de Valladolid
Departamento de Informática (ATC, CCIA, LSI)
Campus Miguel Delibes, 47011 Valladolid, España
email: {mercedes,cecuesta,pfuente}@infor.uva.es

+ LORIA, BP 239
54506 Vandoeuvre Cedex, France
email: lamirel@loria.fr

Resumen. La heterogeneidad en las bibliotecas digitales dificulta la integración de colecciones documentales destinadas a constituir una nueva biblioteca digital. Dicha heterogeneidad puede deberse, entre otras, a las siguientes causas: distintos modelos de documentos, formatos diferentes, distintos modelos en los atributos, y distintas sintaxis en los lenguajes de consulta y en los datos que circulan por la biblioteca. La biblioteca que acoge una nueva colección debe garantizar la interoperabilidad entre la recién llegada y el resto del sistema. Las soluciones para bibliotecas heterogéneas se apoyan en la utilización de protocolos creados específicamente para la Recuperación de Información y/o mediadores que se integran en la biblioteca, asumiendo el papel de "traductores". En este trabajo presentamos y comparamos dos modelos de biblioteca digital donde se trata el problema de la heterogeneidad: una arquitectura donde los mediadores se implementan como objetos, y una solución utilizando el protocolo Z39.50.

1 Introducción

La integración de colecciones heterogéneas en una biblioteca digital es una tarea que puede resultar tan complicada que la integración conseguida sea meramente parcial. Dicha integración supone resolver de forma transparente al usuario los problemas de interoperabilidad que genera la heterogeneidad entre las colecciones que se pretende cooperen en la biblioteca. Si bien los problemas de heterogeneidad relacionados con las plataformas y lenguajes de implementación han sido abordados de forma satisfactoria gracias al soporte que proporcionan plataformas como CORBA [15], el problema de la interoperabilidad semántica [13] no ha encontrado solución de forma tan sencilla, y no está claro que se esté cerca de una solución general a dicho problema.

La interoperabilidad semántica en el contexto de las bibliotecas digitales significa ser capaces de conseguir entendimiento, entre otros, en la semántica de las consultas en el sistema y en la semántica de las respuestas. También encontramos en esta categoría la heterogeneidad en los modelos de documentos en la biblioteca. Por otro lado, tenemos los problemas de heterogeneidad sintáctica propios de las bibliotecas digitales, como por ejemplo, la diversidad en los lenguajes de consulta. Las soluciones para bibliotecas heterogéneas se apoyan en la utilización de protocolos creados específicamente para la Recuperación de Información -como Z39.50- y/o mediadores que se integran en

la biblioteca, asumiendo el papel de "traductores".

En este trabajo mostramos cómo se ve afectada la arquitectura de una biblioteca digital por la heterogeneidad en sus colecciones. Presentamos una solución basada en el uso de mediadores, y la comparamos con otra solución implementada previamente utilizando Z39.50, incidiendo en cuáles son los aspectos de Z39.50 que consideramos conveniente aprovechar en una biblioteca digital que persiga una buena interoperabilidad semántica.

Bibliotecas Digitales Heterogéneas

Las **bibliotecas digitales** ofrecen a los usuarios información, agrupada en colecciones *organizadas*. Permiten acceder a la información que almacenan, consultarla, y proporcionan otros servicios tendentes a facilitar la interacción del usuario con el sistema. La *distribución* y la *heterogeneidad*, en caso de que existan, son transparentes al usuario, que percibe la biblioteca como una colección única y homogénea.

Las bibliotecas digitales a menudo están formadas por varias colecciones. La biblioteca consta de un punto de acceso, que proporciona la interfaz de usuario y redirige las consultas de usuario a los servidores de datos. Una vez recibe las respuesta de todos los servidores de datos, integra los resultados individuales y los presenta al usuario como una sola respuesta.

A la hora de integrar un nuevo repositorio al sistema pueden ocurrir dos cosas: primera, la nueva colección se integra al sistema aportando sus propios métodos de acceso a los datos. En este caso debe realizarse en algún momento la “traducción” del lenguaje de consulta local, criterios de búsqueda locales y esquema de la colección, a los equivalentes que el sistema es capaz de manipular [11]. Segunda, en el momento de la integración en el sistema, cada servidor de datos adopta un módulo de indexación y búsqueda común con la totalidad del sistema. En cualquier caso, se corre el riesgo de conseguir interoperabilidad semántica a costa de pérdida de riqueza semántica en la colección.

Heterogeneidad e Interoperabilidad

Las causas de heterogeneidad en una biblioteca son múltiples, aunque se pueden establecer dos grandes categorías: *heterogeneidad en los* **datos** (incluidos los metadatos) y *heterogeneidad en las* **aplicaciones** que los crean y los manipulan.

En lo que concierne a la **heterogeneidad de los datos**, puede deberse a la heterogeneidad de formatos (.doc, .tex, .xml, .gif, .tiff, etc), a la coexistencia de distintos modelos de documentos, y a la diversidad en el vocabulario. Los aspectos relacionados con el vocabulario son especialmente relevantes e incluyen la heterogeneidad en los criterios de búsqueda y la heterogeneidad en la semántica de los resultados. También nos encontramos la heterogeneidad en los esquemas que describen las colecciones.

Otra causa de heterogeneidad es la utilización de modelos de atributos diferentes en cada colección [2], como por ejemplo, *Dublin Core* [6] (estándar para la descripción de recursos Web), las colecciones de atributos *MARC* [9] (los estándares MARC definen un conjunto de atributos que describen registros bibliográficos), o el conjunto *Bib-1* de *Z39.50* [20] (conjunto de atributos definido dentro del protocolo Z39.50).

En cualquier caso, es necesario disponer y manipular correctamente la información sobre cuáles son los conjuntos de atributos permitidos en cada colección. También se requiere ser capaz de obtener la equivalencia entre un atributo -y el conjunto de valores posibles para dicho atributo- en un cierto modelo de atributos y el atributo o conjunto de atributos equivalente en otro modelo. Este cálculo puede ser complejo cuando la equivalencia entre atributos no es uno a uno. Por ejemplo, mientras que *Dublin Core* define un único atributo para caracterizar cualquier autor de un documento, los estándares MARC distinguen entre varios tipos de autores.

Si se trata de las **aplicaciones**, la causa de heterogeneidad puede estar en el tipo de aplicaciones, el modo de tratar los datos (de los sistemas de gestión de bases de datos a las herramientas de recuperación de información textual), y en las plataformas o lenguajes de implementación.

La variedad en las herramientas de búsqueda utilizadas en el acceso a cada colección implica que existirá heterogeneidad en la sintaxis de los lenguajes de consulta, así como heterogeneidad en los resultados de las consultas. Heterogeneidad sintáctica en las consultas es la situación de varias consultas con distinta sintaxis, pero semántica común. Por ejemplo, la consulta "*buscar la aparición simultánea de los términos X e Y en los documentos de la base*", puede traducirse en "`X and Y`", "`X Y`", "`+X +Y`", etc. Sin embargo, todas ellas son representaciones de la misma consulta.

Solucionar los problemas de heterogeneidad no consiste en limitarse a presentar de modo uniforme los datos al usuario. Se debe conseguir que los servidores que componen la biblioteca interaccionen correctamente, a pesar de la heterogeneidad en el sistema. Esto es, debe existir *interoperabilidad* entre los elementos del sistema.

Una situación ideal (interoperabilidad máxima) sería aquella en la cual cada aplicación se crearía de modo independiente, atendiendo a los criterios particulares de cada creador. Sería suficiente conectar esta aplicación a un bus virtual, que se encargaría de la traducción de los formatos y servicios de forma transparente. Dado que este ideal no se ha conseguido aún, las soluciones actuales se apoyan en la utilización de protocolos, y/o en elementos intermedios que cumplen la función de "traductores". Esto supone incorporar una capa en la biblioteca que se ocupe de las traducciones mencionadas. La arquitectura de la biblioteca digital se ve así afectada por la introducción de nuevos elementos que asumirán dicha tarea. Dichos elementos pueden ser implementados como *mediadores* [19].

En caso de optar por utilizar un protocolo, se cuenta con protocolos pensados expresamente para la Recuperación de Información (IR), como *Dienst* [5], Z39.50 [21], *DLIOP* [16], *STARTS* [7], etc. La principal aportación de los protocolos IR es que abordan el problema de la interoperabilidad semántica, que, en ningún caso, resuelven los protocolos de las capas inferiores. De un modo u otro, la adopción de uno de estos protocolos implicará la incorporación al sistema de nuevos elementos, que implementan los servicios del protocolo.

2 Una Biblioteca Digital sin Heterogeneidad

Para apreciar mejor cómo afecta la heterogeneidad a una biblioteca digital, vamos a comenzar con una biblioteca digital en la cual no hay heterogeneidad. En esta

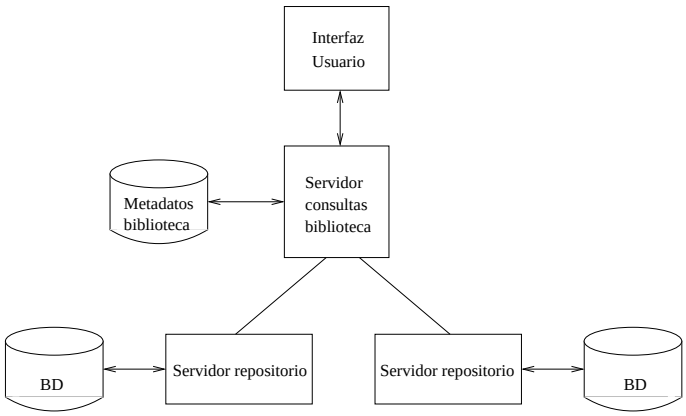

Figura 1: Distribución de consultas sin heterogeneidad

biblioteca existe un conjunto de elementos, así como un conjunto de datos y metadatos que el sistema utiliza para redirigir las consultas a las colecciones de la biblioteca. Los componentes del sistema se implementan como objetos, a los cuales se accede a través de las interfaces que exportan.

Arquitectura

La arquitectura que permite redistribuir la consulta en un sistema de este tipo es la que aparece en la figura 1. El servidor de consultas redirecciona la consulta del usuario a los servidores de repositorios. Las consultas que reciben todos los servidores de datos son idénticas. Tras recibir los resultados de todas las colecciones, lo siguiente que hará el servidor de consultas es fusionarlos.

Los elementos que forman parte de esta arquitectura son:

1. La **interfaz de usuario**. Facilita la interacción con el usuario, que podrá recuperar documentos de los servidores de repositorios y consultar la biblioteca.
2. El **servidor de consultas**. Encargado de recoger las consultas que le proporciona el servidor de usuario y efectuar aquellas operaciones necesarias para devolver una lista de registros respuesta.

   Concretamente, se ocupa de:

   (a) Redireccionar la consulta a los servidores de datos.
   (b) Integrar los resultados que recibe de los servidores de datos y devolver el conjunto resultante al servidor de usuario.
3. Los **servidores de datos** (o **servidores de los repositorios**). Uno por cada base de datos del sistema. Permiten la recuperación de registros y realizar búsquedas.

Metadatos

Los metadatos que necesita el **servidor de consultas** para redireccionar la consulta que recibe como entrada son pocos. Básicamente, se trata de la lista de colecciones en el sistema y el modo de acceder a cada una.

3 Introducción de Heterogeneidad en la Biblioteca

Si pretendemos incorporar a esta biblioteca nuevas colecciones cuyos documentos se ajusten a esquemas distintos, donde los atributos de búsqueda y el lenguaje de búsqueda difieran de los que hasta ahora tenemos, debemos considerar nuevos metadatos e introducir nuevos componentes que faciliten dicha integración. Como dijimos en la Introducción, las tareas de traducción en una biblioteca heterogénea pueden ser asumidas por agentes mediadores.

3.1 Los Mediadores y las Bibliotecas Digitales

Los **mediadores** [19] aparecen como una de las soluciones más utilizadas en las bibliotecas digitales para obtener la deseada interoperabilidad [11]. También se les conoce como *wrappers* o *proxies*. Son componentes de software que asumen las tareas de traducción necesarias para incorporar un nuevo servidor al sistema. Facilitan la autonomía de los sistemas locales, porque recae sobre estos mediadores la traducción entre los formatos de datos y los modos de interacción. Algunos ejemplos de agentes mediadores son los gateways SQL - XML o HTTP - Z39.50.

3.2 La Solución con Objetos Mediadores

Los mediadores que se ocupan de la traducción pueden ser implementados como objetos accesibles por el resto de los componentes de la biblioteca a través de sus interfaces. Lo significativo de los cambios respecto a la biblioteca homogénea de la sección 2 se encuentra en los nuevos componentes (sección 3.2.2) y nuevos metadatos que describen la heterogeneidad del sistema (sección 3.2.1). Una descripción más completa de esta solución se puede consultar en [10].

3.2.1 Metadatos

Puesto que nos encontramos con heterogeneidad en los modelos de atributos y los lenguajes de consulta, necesitaremos metadatos en cada colección que la describan así como metadatos que informen sobre la heterogeneidad en el conjunto del sistema.
Los metadatos correspondientes al sistema son:

- Lenguajes de consulta en el sistema y traductor asociado a cada lenguaje.
- Modelos de atributos que existen en el sistema. Se utilizará para decidir en cada caso cuáles son los traductores que se deben emplear durante la interacción con cada colección.

Cada colección debe exportar información sobre si misma. Esta información será utilizada por el servidor de consultas:

- Modelo de atributos en la colección.
- Descripción de dicho modelo.

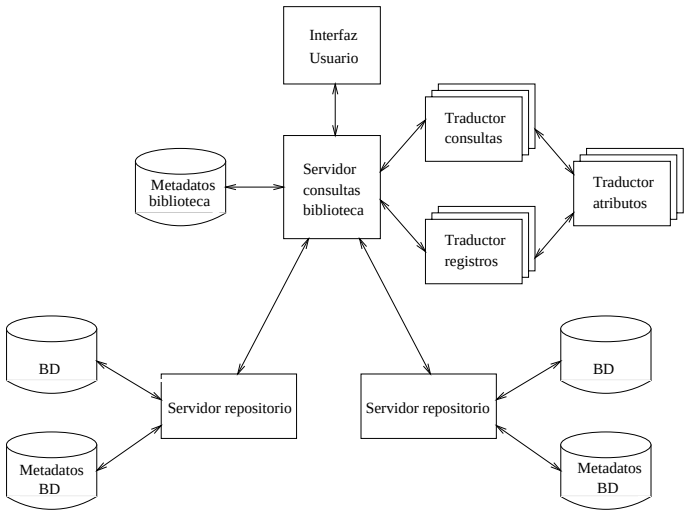

Figura 2: Objetos mediadores en una biblioteca heterogénea

- Lenguaje de consulta en la colección.

Además, cada colección debe proporcionar un conjunto de *facilidades* que posibiliten el acceso a sus metadatos.

3.2.2 La Arquitectura Modificada

La arquitectura de la biblioteca se ve, pues, ampliada con la introducción de nuevos componentes. Estos componentes son los **traductores** (mediadores), que aparecen en la figura 2:

- Los **traductores de consultas**. Tantos como lenguajes de consulta diferentes haya en el sistema. Son traductores 1:1, del lenguaje de consulta "genérico" en el cual recibe la consulta el servidor de consultas, al lenguaje de consulta local de cada herramienta de consulta.
- Los **traductores de formato**. Traducen los resultados del formato local de respuesta de cada servidor de datos al formato común, que entiende el servidor de consultas.
- Los **traductores de atributos**. Son traductores uno a uno entre los modelos de atributos. Habrá tantos como modelos de atributos diferentes hay en el sistema; cada traductor realiza la conversión entre un cierto modelo y el modelo común utilizado para la interacción en el sistema.

4 La Solución con Z39.50

Una solución alternativa para la heterogeneidad es la utilización del protocolo Z39.50, concebido para abordar la interoperabilidad semántica entre distintas bibliote-

cas. La comparación entre esta solución y la propuesta descrita en la sección anterior se encuentra en las conclusiones de este trabajo.

4.1 El Protocolo Z39.50

Z39.50 [20] es un protocolo ANSI/NISO, específico para el contexto de Recuperación de Información. Surgió para resolver el problema de la interoperabilidad en las bases de datos bibliográficas, aunque cuenta con perfiles adaptados a otros tipos de comunidades.

Z39.50 se apoya en un modelo cliente/servidor. El *cliente* puede interrogar varias bases de datos, conectándose a un *servidor* Z39.50, que es el que se ocupa de acceder a dichas bases y proporcionar las respuestas al cliente. El protocolo regula la interacción entre ambos, y define un conjunto de reglas sintácticas y semánticas sobre los datos y los resultados intercambiados entre cliente y servidor.

En el aspecto sintáctico, Z39.50 admite varias sintaxis para las consultas. Asimismo, en los registros se aceptan varias sintaxis; entre ellas los registros MARC[1]. En lo referente a la semántica, el estándar define varios conjuntos de atributos. Cada conjunto de atributos expresa la semántica de un área particular. El conjunto más elaborado es el *Bib-1*, que consiste en la colección de campos que pueden caracterizar los registros bibliográficos.

4.2 Arquitectura

La implementación de la biblioteca usando Z39.50 supone incorporar en cada servidor de colecciones que pretende integrarse en la biblioteca un servidor Z39.50, que asumirá las tareas de traducción de los formatos y criterios locales de la colección a los criterios y formatos Z39.50. En este caso el estándar de intercambio de información en el sistema es el definido por el protocolo Z39.50. Las tareas de traducción se realizan siempre en cada servidor de colecciones antes de consultar o responder a los demás nodos del sistema.

La arquitectura de la biblioteca se ve, pues, influenciada por las directrices que marca el protocolo Z39.50. Esta arquitectura se muestra en la figura 3, y consta de los elementos que se citan:

- **Interfaz de usuario**.
  Este elemento no varía, en lo que se refiere a su funcionamiento.
- **Servidor de consultas**.
  En este caso debe incorporar un cliente Z39.50, que será el que le permita dialogar con las colecciones de la biblioteca.
- **Servidor Z39.50**.
  Cara a los demás elementos del sistema, cada colección es visible a través de un servidor Z39.50, que oculta todas las tareas de traducción y acceso locales a la colección. Es decir, este servidor dialoga con el sistema de acceso local

[1] Machine-Readable Cataloging.

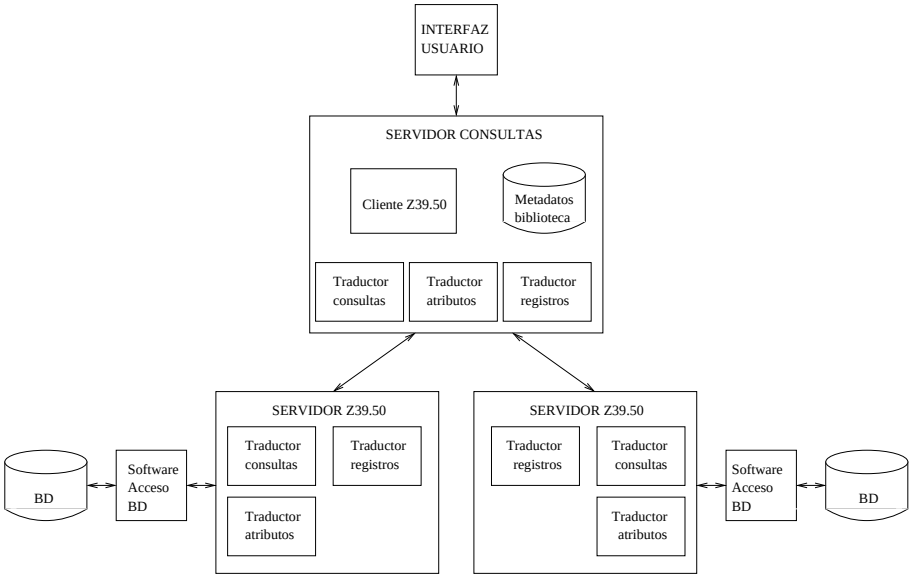

Figura 3: Una biblioteca heterogénea con Z39.50

a la Base de Datos, asumiendo todas las tareas de traducción necesarias para ello. Los traductores y las facilidades de acceso asociadas a los metadatos de las colecciones quedan también ocultos tras los servidores Z39.50 de cada colección.

5 Trabajos Relacionados

Algunos ejemplos de bibliotecas digitales en las cuales se utilizan mediadores para resolver los problemas causados por la heterogeneidad en la biblioteca son la biblioteca digital de ERCIM [3], NDLTD [14], el proyecto de Bibliotecas Digitales de Stanford [12], *Aleph* [8] y AQUARELLE [1]. Stanford propone la utilización de mediadores (*Library Service Proxies*) que se ocupan de las labores de traducción necesarias para integrar servidores implementados inicialmente sobre protocolos diferentes. En NDLTD también ponen especial atención en los metadatos que describen los servidores; sin embargo, uno de sus fines primordiales es permitir consultas multilingües. El proyecto AQUARELLE propone una solución para bibliotecas distribuidas, utilizando el protocolo Z39.50; cada servidor de datos que se incorpora al sistema debe implementar un servidor Z39.50.

Por último, el proyecto CICYT “Biblioteca Virtual de Textos Españoles de Siglo de Oro accesibles via Internet” [4] -en el cual participan tres de los autores de este trabajo- consiste en la implementación de una biblioteca digital distribuida federada, uno de cuyos objetivos es conseguir la fácil integración de servidores heterogéneos, como son bases de datos relacionales y servidores de documentos XML.

6 Conclusiones

Las conclusiones de este trabajo consisten en una comparación entre la solución Z39.50 y la solución con los objetos mediadores.

Como se ha visto, la decisión de utilizar Z39.50 obliga a incorporar con cada colección un servidor Z39.50, que asuma las tareas de integración semántica y las tareas de comunicación de servidor con otros servidores o clientes Z39.50. Si utilizamos los objetos mediadores, lo único que se requiere a cada nueva colección es que permita acceder a ella mediante las interfaces definidas con este propósito. Las tareas de comunicación entre objetos las asume la capa que implementa la interacción entre objetos. Es decir, en la solución basada en objetos la incorporación de nuevas colecciones requiere un esfuerzo mucho menor de implementación y aprendizaje por parte del proveedor. Z9.50 es un protocolo complejo, que requiere un considerable esfuerzo de aprendizaje; la programación orientada a objeto está muy extendida, con lo cual se puede considerar nulo el tiempo de aprendizaje necesario. Una razón más a favor de una solución con objetos mediadores es que facilita la incorporación de nodos poco colaborativos. Los proveedores no deben aportar los traductores; es el sistema quien los proporciona. En resumen, desde el punto de vista del proveedor, la solución con objetos mediadores es más cómoda.

Sin embargo, recaen sobre los responsables del sistema todas las tareas que garantizan la interoperabilidad en la biblioteca. Estas tareas van desde la implementación de los traductores, a la elección de un lenguaje de consulta y modelo de atributos común al sistema. La interoperabilidad sintáctica se puede resolver cómodamente, utilizando estándares como XML [17], RDF [18] en la decripción de los datos y metadatos del sistema. Es distinto el caso de la interoperabilidad semántica, donde el estándar más popular en las bibliotecas en la Web -*Dublin Core*- es objeto de numerosas "extensiones", o adaptaciones a las necesidades particulares de cada comunidad. Los 15 campos que proporciona distan de proporcionar la riqueza semántica del conjunto *Bib-1* (más de 100 atributos), lo cual lo hace insuficiente en casos como el de las colecciones bibliográficas accesibles en los OPACs[2].

En conclusión, si bien en los aspectos arquitectónicos se pueden encontrar soluciones más sencillas, las aportaciones de estándares como Z39.50 en lo que a semántica se refiere -conjuntos de atributos y perfiles- son valiosas y merecen ser tenidas en cuenta en el momento de tratar la heterogeneidad semántica en las bibliotecas digitales.

Referencias

[1] AQUARELLE. The Information Network on the Cultural Heritage. http://aqua.inria.fr/Aquarelle/.

[2] M. Baldonado, C. Chang, L. Gravano, y A. Paepcke. The Stanford Digital Library Metadata Architecture. *International Journal of Digital Libraries*, 2(1), Febrero 1997.

[3] S. Biagioni, J.L.Borbinha, R. Ferber, P. Hansen, S. Kapidakis, L. Kovacs, F. Roos, y A. M. Vercoustre. The ERCIM Technical Reference Digital Library. En *Second European Conference on Research and Advanced Technology for Digital Libraries*, Septiembre 1998.

[2] Online Public Access Catalogs.

[4] Nieves R. Brisaboa, M. José Durán, Charlo Lalín, Juan Ramón López, José R. Paramá, Miguel R. Penabad, y Ángeles S. Places. Propuesta de Arquitectura para un Sistema de Bases de Datos Documentales. En *IV Jornadas de Ingeniería del Software y Bases de Datos*, Noviembre 1999.

[5] J. R. Davis y C. Lagoze. A protocol and server for a distributed digital technical report library. Technical Report TR94-1418, Computer Science Department, Cornell University, 1994.

[6] Dublin Core Element Set: Reference Description. http://purl.org/dc.

[7] Luis Gravano, Kevin Chang, Hector Garcia-Molina, Carl Lagoze, y Andreas Paepcke. STARTS. Stanford Protocol Proposal for Internet Retrieval and Search. Technical report SIDL-WP-1996-0043, Stanford University, Agosto 1996. http://www-diglib.stanford.edu/cgi-bin/WP/get/SIDL-WP-1996-0043.

[8] Gerard Rodríguez i Mulà. *Federation Mechanisms for Large Scale Cooperative Systems*. PhD thesis, Department of Computer Architecture. Universitat Politècnica de Catalunya.

[9] MARC STANDARDS. http://lcweb.loc.gov/marc/.

[10] M. M. Martínez, C. E. Cuesta, P. de la Fuente, y J. C. Lamirel. Consultas heterogéneas en bibliotecas digitales distribuidas. En *Simposio Español de Informática Distribuida (SEID 2000)*, Ourense, septiembre 2000.

[11] Andreas Paepcke, Che Chuan K. Chang, Héctor García Molina, y Terry Winograd. Interoperability for Digital Libraries Worldwide. *Communications of the ACM*, Abril 1998.

[12] Andreas Paepcke, Steve B. Cousins, Scott W. Hassan, Ketchpel Steven P, Martin Roscheisen, Héctor García Molina, y Terry Winograd. Using distributed objects for digital library interoperability. *Computer*, Mayo 1996.

[13] Margaret St. Pierre. Z39.50 and semantic interoperability. En *Distributed Indexing/Searching Workshop*, Mayo 1996.

[14] James Powell y Edward A. Fox. Multilingual Federated Searching Across Heterogeneous Collections. *D-Lib Magazine*, Septiembre 1998.

[15] Jon Siegel. *CORBA. Fundamentals and Programming*. Wiley Computer Publishing. John Wiley & Sons, Inc., 1996.

[16] Stanford Digital Library Group, {hassan,paepecke}@cs.stanford.edu. *Stanford Digital Library Interoperability Protocol*, 1996.

[17] W3C. *Extensible Markup Language (XML) 1.0*, Febrero 1998. W3C Recommendation 10-February-1998, http://www.w3.org/TR/REC-xml.

[18] W3C. *Resource Description Framework (RDF) Schema Specification 1.0*, Marzo 2000. W3C Candidate Recommendation 27 March 2000, http://www.w3.org/TR/rdf-schema.

[19] G. Weiderhold. Mediators in the architecture of future information systems. *IEEE Computer*, págs. 38–49, Marzo 1992.

[20] Z39.50 Maintenance Agency. *ANSI/NISO Z39.50-1995. Information Retrieval (Z39.50): Application Service Definition and Protocol Specification*, Julio 1995.

[21] Z39.50 Maintenance Agency. *Attribute set Bib-1 (Z39.50-1995): Semantics*, Septiembre 1995. ftp://ftp.loc.gov/pub/z3950/defs/bib1.txt.