Gestión de Bibliotecas Digitales de Publicaciones de Investigación

J.A. Royo* and E. Mena

Dpto. de IIS, Univ. de Zaragoza
María de Luna 3, 50018 Zaragoza, España
{joalroyo,emena}@posta.unizar.es

Resumen Una de las tareas más comunes en la elaboración de artículos de investigación es el mantenimiento de un depósito bibliográfico que se utilice tanto para su consulta como para su referencia desde los nuevos artículos de investigación que se desarrollen.
Sin embargo, debido a la propia naturaleza dinámica de los trabajos de investigación, estos depósitos tienen un alto coste de mantenimiento ya que deben ser actualizados de forma periódica. No sólo hay que dar solución al almacenamiento sino también a la integración con los editores de texto utilizados. Este problema es aún mayor cuando son varios miembros del mismo grupo de investigación quienes actualizan y consultan las publicaciones manejadas por dicho grupo.
En este artículo presentamos una aproximación que, mediante la creación de servicios web, soluciona muchos de los problemas para el mantenimiento de una biblioteca digital de artículos de investigación a la vez que facilita su integración con un procesador de texto

Keywords: Interoperabilidad, Bibliotecas digitales de documentos científicos.

1 Introducción

Actualmente el volumen de información bibliográfica es muy elevado y crece día a día. Por ello es necesaria una gestión adecuada de las publicaciones y la automatización de tareas como la generación automática de la bibliografía asociada a un documento. En el contexto de la elaboración de artículos de investigación no es suficiente con referenciar el documento sino que éste debe estar accesible para poder consultarlo, por ejemplo a través de la Web.

Uno de los problemas que aparecen en contextos dinámicos como el de la investigación es la continua necesidad de actualización como consecuencia de contrastar el trabajo desarrollado con el de otros investigadores. Muchas veces se referencian trabajos en espera de su evaluación, o aceptados para su publicación pero que la misma tardará meses en producirse. Esto lleva a la utilización de referencias incompletas cuyos datos de publicación (páginas, ISBN, etc) pueden tardarse meses en conocer. Además ciertas referencias pueden dejar de ser

* Trabajo financiado por la beca B131/2002 del Gobierno de Aragón y el Fondo Social Europeo.

útiles al aparecer referencias mejores sobre el mismo tema (nuevos artículos más elaborados y/o publicados en eventos de más prestigio).Por otra parte hay que considerar cuestiones como la utilización de diversas fuentes de datos bibliográficos, lo que puede dar lugar a la aparición de inconsistencias.

Existen trabajos que permiten la integración de referencias bibliográficas de forma automática. En [17] se presenta una aproximación basada en agentes móviles [15] que permite la integración de distintos depósitos BibTEX [10] en uno único, concretamente en una base de datos relacional. En [3] se presenta un sistema para crear y generar bibliografía para Microsoft Word con una base de datos bibliográfica propia, aunque se trata de una solución de difícil adaptación a otros procesadores de texto. Además existen herramientas comerciales como Procite [4] que permiten el manejo de referencias bibliográficas pero no permite la gestión de documentos digitales. Igualmente existen diversos depósitos de referencias de trabajos científicos como *Research Index* [8] (que realiza búsquedas bibliográficas a través de la Web), DBLP [11], y *The Computer Science Bibliography* [1]. Sin embargo estos sistemas no permiten la integración con un procesador de texto para generar automáticamente la bibliografía de la publicación que se está editando.

El resto del artículo tiene la siguiente estructura: en la Sección 2 presentamos la arquitectura del sistema propuesto. En la Sección 3 se describe el mecanismo de inserción de nuevas publicaciones. Los procedimientos de consulta y actualización son tratados en la Sección 4. En la Sección 5 tratamos la generación de bibliografía para un procesador de textos, en nuestro caso LATEX [10]. Finalmente la Sección 6 recoge las conclusiones y trabajo futuro.

2 Arquitectura del Sistema

Las ideas presentadas en este trabajo han surgido como solución a las necesidades de gestión de publicaciones científicas del grupo de Bases de Datos Interoperantes (BDI) de la Universidad del País Vasco, cuyos miembros están distribuidos en dos universidades. En [17] se presentó un mecanismo, basado en la tecnología de agentes móviles [15], para la creación automática de depósitos de publicaciones de investigación a partir de un conjunto de depósitos de referencias distribuidos. Concretamente se proponía la utilización de un agente móvil para la recopilación de información bibliográfica a partir de distintos ficheros BibTEX, visitando uno tras otro los ordenadores que contienen dichos ficheros e integrando los datos incrementalmente, para finalmente almacenarlos en una base de datos relacional. Esta aproximación presentaba las siguientes ventajas:

- *Recopilación de información distribuida*, mediante la utilización de un agente móvil. Este agente viajará al nodo que contiene el primer depósito de datos y, después de que la información haya sido analizada, el agente viajará al siguiente nodo, integrará los nuevos datos con los anteriores, y así sucesivamente hasta haber completado el análisis de todos los nodos.
- *Generación de información web a partir de la base de datos*, permitiendo una gestión automatizada del sitio web de publicaciones del grupo.

- *Detección automática de inconsistencias* entre la información disponible en los depósitos de datos distribuidos. La misma referencia[1] puede haber sido introducida por distintos usuarios con deficiencias en su información.
- *Robustez frente a desconexiones*, el agente móvil es capaz de reaccionar ante los fallos de red que se produzcan, según cierta política de reintentos. Así completará su tarea sin que sea necesaria la intervención del usuario.
- *Integración de la información bibliográfica* en una base de datos con la idea de hacer accesible a otras aplicaciones cualquier publicación disponible en los depósitos BibTEX.

A pesar de las mejoras, con el tiempo se han detectado otros problemas: 1) *Problemas de actualización*, ya que deben actualizarse todos los ficheros BibTEX que contengan dicha referencia; 2) *Problemas de inserción*, las nuevas publicaciones deben introducirse en alguno de los ficheros BibTEX y debe existir algún mecanismo automático que lance el agente móvil para añadirlas al depósito centralizado; y 3) *Baja interoperabilidad*, el sistema únicamente puede ser accedido mediante las aplicaciones creadas, por lo que su funcionalidad no puede reutilizarse fácilmente desde nuevas aplicaciones.

A continuación presentamos una arquitectura que soluciona los problemas presentados anteriormente. Para permitir una mejor integración de la biblioteca digital con otras aplicaciones, el sistema se ha diseñado como una serie de servicios Web [9], permitiendo una fácil interoperación con páginas HTML dinámicas y con otras aplicaciones mediante el uso de XML [12] y del protocolo SOAP [18]. La utilización de servicios web facilita la interoperación porque permite interconectar aplicaciones utilizando un formato de intercambio de datos estándar, XML. Los servicios web debido a que están basados en HTTP no presentan los problemas de CORBA a la hora de atravesar cortafuegos. Arquitecturas distribuidas o *peer to peer* podrían dar lugar a la aparición de inconsistencias entre los datos de las publicaciones.

La arquitectura propuesta (ver Figura 1) presenta los siguientes tres niveles:

1. *Presentación*: desarrollada mediante páginas web dinámicas utilizando JSP [6], lo que permite la separación de los datos de su visualización. En esta capa se encuentran servicios de inserción, edición y consulta de datos (secciones 3 y 4, respectivamente).
2. *Lógica del sistema*: desarrollada mediante servicios web para permitir la interacción con otras aplicaciones, ya que la entrada y salida de esta capa son datos en XML siguiendo el protocolo SOAP (basado en HTTP).
3. *Acceso a datos*: es la encargada del acceso a los distintos depósitos de datos. Esta capa esta formada por un conjunto de *wrappers* [7] que permiten el acceso a la base de datos de publicaciones y a los ficheros Bib. El wrapper que permite el acceso a los ficheros BibTEX se encuentra integrado dentro del agente móvil que se puede ver en la Figura 1, este wrapper solamente es utilizado en la inserción masiva de datos.

[1] En nuestro sistema de información por publicaciones entendemos tanto referencias como su documento digital asociado.

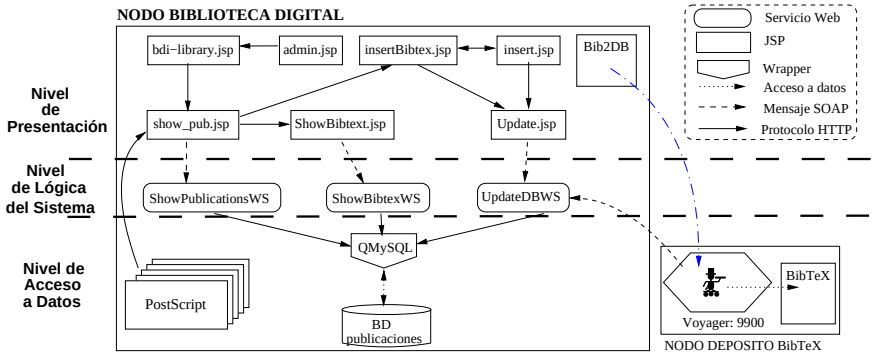

Figura 1. Arquitectura del Sistema

En lo referente a seguridad, los nombres de usuario y contraseñas se almacenan en la base de datos. En las secciones que siguen presentaremos en detalle los módulos principales de la arquitectura presentada.

3 Inserción de Publicaciones

En esta sección se presenta la forma en la que se introducen las nuevas publicaciones en el sistema. Existen dos mecanismos diferenciados:

- *Inserción masiva* mediante el agente móvil desarrollado en [17], para cuando quieren introducirse un gran número de publicaciones, posiblemente almacenadas en varios ficheros de referencias.
- *Inserción manual* que permite la inserción de publicaciones mediante la utilización de dos formularios web: 1) *Modo registro*[2], que permite la inserción de publicaciones a partir de su registro BibTEX (ver Figura 2); y 2) *Modo campo a campo*, para introducir cada uno de los datos asociados a la publicación de forma separada (ver Figura 3). Ambos métodos pueden ser utilizados indistintamente debido a un mecanismo automático de traducción que permite pasar de un formulario/modo de inserción a otro.

Durante el proceso de inserción de una nueva publicación, por cualquiera de los métodos descritos anteriormente, se utiliza el sistema de verificación de inconsistencias desarrollado en [17]. Dicho en pocas palabras, este mecanismo permite detectar: 1) si dos publicaciones con el mismo identificador tienen información distinta en alguno de sus campos, y 2) si dos referencias con distinto identificador pueden estar describiendo a la misma publicación. En [17] se explica de forma detallada la casuística completa que presenta este problema.

[2] El modo registro está asociado al JSP "insertBibtex" y el modo campo a campo al JSP "insert".

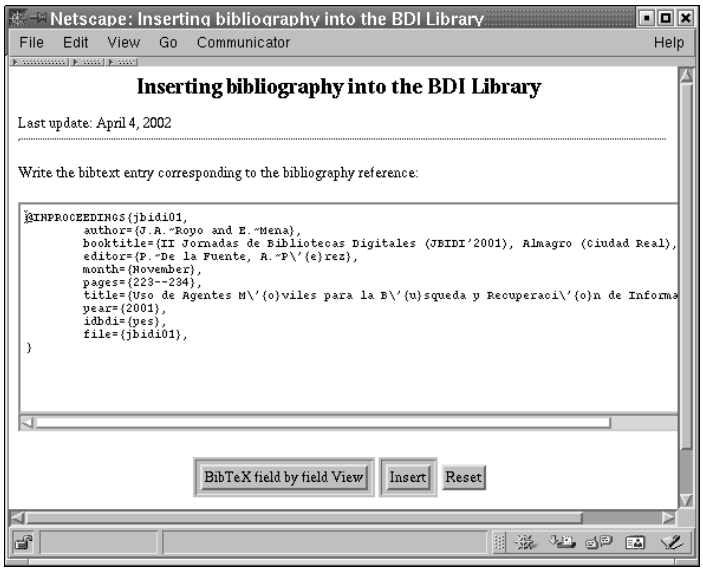

Figura 2. Inserción modo registro

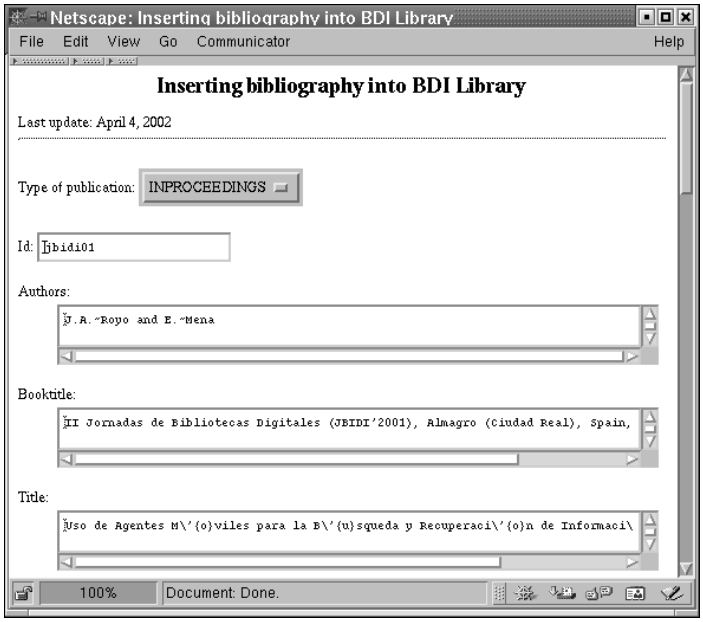

Figura 3. Inserción modo campo a campo

4 Consulta y Actualización de Publicaciones

En la biblioteca digital desarrollada se permite la realización de búsquedas a través de un formulario construido dinámicamente. En dicho formulario se per-

mite la búsqueda mediante la selección de los temas que tratan, tipo de publicación[3], o autores de las publicaciones (ver Figura 4). También es posible imponer restricciones de tipo subcadena a cualquiera de los campos de una publicación. En la Figura 5 puede observarse el resultado de la consulta realizada.

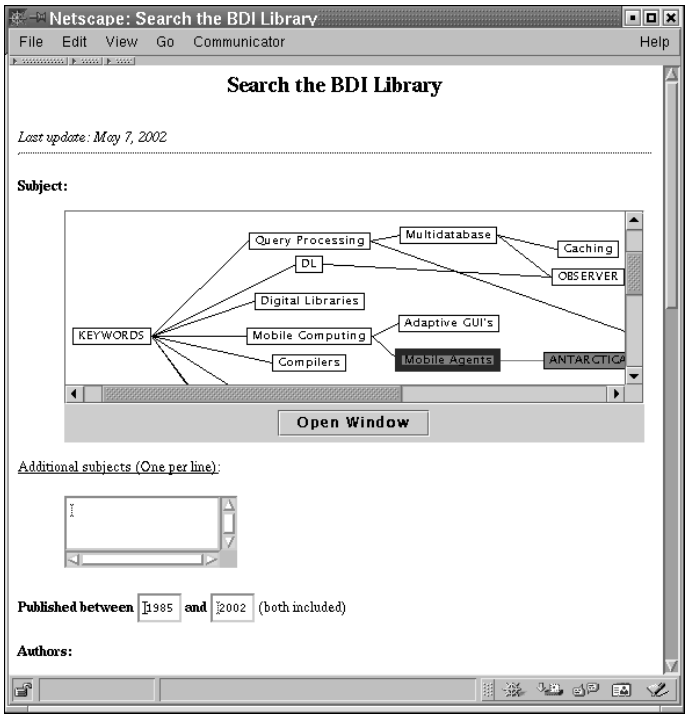

Figura 4. Formulario de consulta

Respecto a la actualización de publicaciones, el proceso sería el siguiente: 1) Búsqueda de las publicaciones a modificar (usando el mismo formulario de la Figura 4); 2) Selección de la publicación a modificar (en una pantalla similar a la de la Figura 5 aparecerá un enlace "Edit" para cada referencia); y 3) Actualización de la publicación, tras pulsar "Edit", mediante formularios similares a los de inserción (Figuras 2 y 3) que mostrarán los datos actuales de la publicación seleccionada, permitiendo cambiarlos.

Utilización de Ontologías

En [13] se describe un problema asociado a la relación entre los posibles valores de las entidades y atributos representadas en un sistema de información. Entre los

[3] Por tipo de publicación nos referimos a si estamos ante un libro, un artículo en revista, etc.

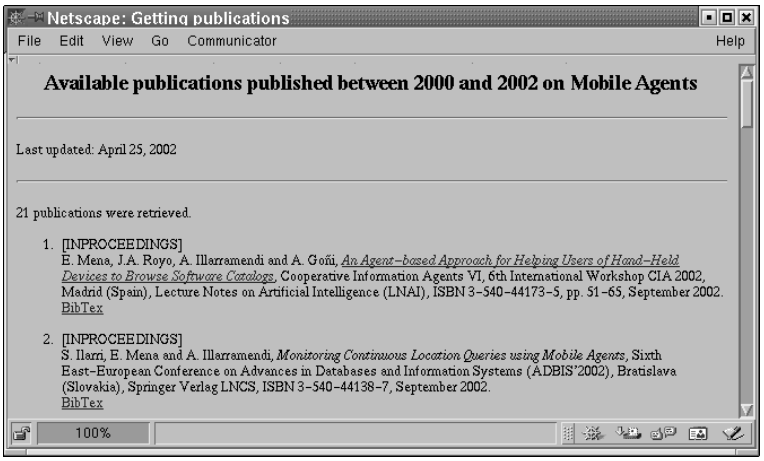

Figura 5. Resultado de la búsqueda

distintos valores puede haber relaciones de sinonímia, hiponímia e hiperonímia; es decir, puede existir una relación semántica entre los posibles valores de los objetos representados. La representación de este tipo de conocimiento puede realizarse mediante la utilización de un sistema terminológico [2].

Por ejemplo, en el caso de las palabras clave, al buscar las publicaciones referentes a "bases de datos distribuidas" no se incluirán en el resultado aquellas publicaciones clasificadas solamente como de "bases de datos federadas", a pesar de que también tratan sobre cierto tipo de bases de datos distribuidas. Por tanto, los usuarios deberían especificar *todas* las palabras clave de una publicación al introducir sus datos, recordando la relación semántica entre ellas; si olvidan alguna, los resultados de las búsquedas no serán semánticamente correctos.

Por tanto, para algunos de los campos de una publicación es muy interesante no ofrecer los posibles valores como una lista plana sino como una ontología [5]. La utilización de ontologías además de permitir la representación de conocimiento, permiten almacenarlo de forma organizada. Por todo esto hay que destacar que la utilización de este mecanismo de representación no sirve únicamente para facilitar las operaciones de búsqueda, sino también para organizar los datos que se almacenan en la biblioteca digital. En la Figura 6 se muestra las ontologías de palabras clave y de tipos de publicaciones manejadas por el grupo de investigación al que pertenecen los autores de este trabajo, pero podría utilizarse un tesaurus ya preexistente como WordNet [14].

En nuestro prototipo del sistema, los formularios de inserción, edición y búsqueda incluyen mediante *applets* una ontología de palabras clave (ver Figura 4). El sistema incluye la conexión a un sistema terminológico [2] que convertirá automáticamente el conjunto de palabras clave (o tipos de publicación) indicadas en un conjunto mutuamente independiente, lo cual optimizará tanto su almacenamiento como las búsquedas. Por ejemplo, si se introduce una publica-

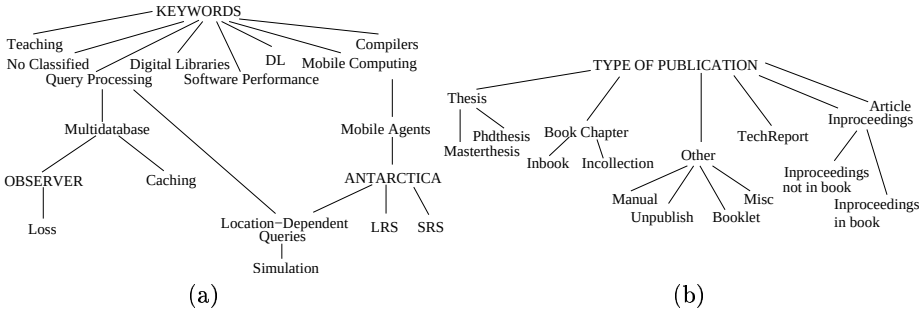

Figura 6. Ontologías de palabras clave (a) y de tipos de publicaciones (b)

ción con "Caching" como única palabra clave, automáticamente será clasificada como una publicación que trata los temas de "Caching", "Multidatabase" y "Query Processing" (ver Figura 6).

5 Generación Automática de Bibliografía

Como ya dijimos en la introducción, una biblioteca digital de publicaciones de investigación no debe tener la finalidad única de servir para la búsqueda de documentos sino que también debe ser posible su conexión con los procesadores de texto que se utilicen. En nuestro caso nos centraremos en el editor LaTeX [10], probablemente el más utilizado para la elaboración de artículos de investigación.

La elección de LaTeX se basa en que incluye "bibtex", una aplicación para generar automáticamente la bibliografía asociada a un artículo tomando como base un fichero de texto en formato BibTeX [10]. Por otra parte, LaTeX no siempre incluye un entorno de trabajo al que añadirle nuevas funcionalidades, como podríamos hacer en un procesador tipo Microsoft Word, sino que muchos de sus usuarios invocan las distintas aplicaciones desde el *prompt* del sistema.

Por tanto hemos desarrollado un servicio que permite seleccionar un fichero LaTeX y, tras analizarlo, generar automáticamente el correspondiente fichero de bibliografía en formato BibTeX. Así se hace necesario un cambio de filosofía de trabajo: en vez de preocuparnos de mantener actualizados varios ficheros BibTeX, las referencias se almacenan siempre en un depósito centralizado que nos ofrece el servicio de generar cuando queramos el fichero de bibliografía en formato BibTeX correspondiente a un documento LaTeX. De la misma manera podría generarse bibliografía en otros formatos sin ninguna dificultad. Como consecuencia de la instalación de la aplicación en una página web se permite que cualquier persona que lo desee pueda utilizar dicho servicio para generar sus ficheros de bibliografía, mediante la utilización de hojas de estilo para transformar los datos XML al formato deseado.

El análisis de documentos LaTeX se realiza mediante una gramática (ver Figura 7) que permite el análisis de los comandos LaTeX identificados por las palabras clave *cite* y *nocite*. El comando *cite* se utiliza para citar una publicación y que su

referencia aparezca en la bibliografía del documento; *nocite* se usa para que una referencia no aparezca en la bibliografía aunque no sea citada desde el texto. El fichero de bibliografía se actualiza cada vez que se ejecuta esta aplicación.

```
<documento> ::= <lista_citas> <bibliografia>
              |

<lista_citas> ::= <cita> <lista_citas>
                | <cita>

<cita> ::= "\cite" "{" <lista_referencias> "}"
         | "\nocite" "{" <lista_referencias> "}"

<lista_referencias> ::= cadena "," <lista_referencias>
                      | cadena
                      | *

<bibliografia> ::= "\bibliography" "{" cadena "}"
```

Figura 7. Gramática para Ficheros LaTeX

En el análisis de ficheros LaTeX se han considerado los siguientes tipos de errores:

- *Error tipográfico*: cuando el identificador de una cita se encuentra en la base de datos pero no coinciden las mayúsculas y minúsculas, se mostrará un mensaje de aviso (tal y como hace la aplicación "bibtex").
- *Identificador erróneo*: cuando en el texto LaTeX aparece una cita a un documento que no existe en la base de datos.
- *Error de seguridad*: cuando no se han dado los permisos Java necesarios para la ejecución del servicio web.
- *Error de conexión a la base de datos*: cuando la base de datos no está accesible por problemas de red.

6 Conclusiones y Trabajo Futuro

En este artículo detallamos una arquitectura que permite integrar una biblioteca digital de publicaciones de investigación con los procesadores de texto utilizados en el desarrollo de los mismos. También hemos desarrollado mecanismos automáticos de detección de inconsistencias entre las publicaciones existentes y las que se desean introducir. La biblioteca digital desarrollada se encuentra en [16].

La arquitectura propuesta presenta características deseables para las bibliotecas digitales de publicaciones científicas como: 1) *Alta interoperabilidad* con otros sistemas al haber sido desarrollado como una serie de servicios web; 2) *Unificación del proceso de inserción y búsqueda de publicaciones*, basado en un depósito de datos centralizado; 3) *Gestión de valores jerarquizados*, mediante el uso de ontologías y un sistema terminológico, que facilita la inserción y búsqueda de datos; 4) *Generación automática de bibliografía* en formato BibTeX y XML; y

5) *Minimización del software instalado*, disminuyendo el uso de recursos de los usuarios y evitando el problema de la actualización de versiones.

Actualmente estamos trabajando en: 1) Integrar el sistema con otros procesadores de texto como Microsoft Word; 2) Permitir el acceso transparente a otras bibliotecas digitales científicas, como la versión actual de BibWord [3]; y 3) Mejorar el mecanismo de detección de inconsistencias.

Referencias

1. A. C. Achilles. The computer science bibliography. http://liinwww.ira.uka.de/bibliography.
2. A. Borgida. From type systems to knowledge representation: Natural semantics specifications for description logics. *International Journal on Intelligent and Cooperative Information Systems*, 1(1), mar 1992.
3. J.H. Canos. A Bibliography Manager for Microsoft Word. *ACM Crossroads Special Issue on Windows Programming, ISSN=1528-4980*, 6.4, June 2000.
4. IIS Research Soft Corp. ProCite. Your information toolbox. http://www.procite.com.
5. T. Gruber. http://www-ksl.stanford.edu/kst/what-is-an-ontology.html.
6. M. Hall. *Core Servlets and Java Server Pages(JSP).* Prentice Hall PTR/Sun Microsystems Press, May 2000.
7. J. Hammer, M. Breunig, H. Garcia-Molina, S.Ñestorov, V. Vassalos, and R. Yerneni. Template-based wrappers in the tsimmis system. In *Proceedings of the Twenty-Sixth SIGMOD International Conference on Management of Data, Tucson, Arizona*, May 1997.
8. NEC Research Institute. Research index. http://citeseer.nj.nec.com/.
9. T. Jewell and D.A. Chappell. *Java Web Services.* O'Reilly & Associates, March 2002.
10. L. Lamport. *LaTeX.A Document Preparation System. User's guide and reference manual.* Addison-Wesley, 1994.
11. Michael Ley. Computer science bibliography. http://dblp.uni-trier.de/.
12. B. McLaughlin. *Java & XML, 2nd Edition: Solutions to Real-World Problems.* O'Reilly & Associates, September 2001.
13. E. Mena and A. Illarramendi. *Ontology-Based Query Processing for Global Information Systems.* Kluwer Academic Publishers, ISBN 0-7923-7375-8, 2001. June 2001.
14. G. Miller. WordNet: A Lexical Database for English. *Communications of the ACM*, 38(11), nov 1995.
15. D. Milojicic, M. Breugst, I. Busse, J. Campbell, S. Covaci, B. Friedman, K. Kosaka, D. Lange, K. Ono, M. Oshima, C. Tham, S. Virdhagriswaran, and J. White. MASIF, the OMG mobile agent system interoperability facility. In *Proceedings of Mobile Agents '98*, September 1998.
16. J.A. Royo. BDI Digital Library, 2002. http://sol1.cps.unizar.es:5080/PUBLICATIONS.
17. J.A. Royo and E. Mena. Uso de agentes móviles para la búsqueda y recuperación de información bibliográfica. In A. Pérez P. De la Fuente, editor, *II Jornadas de Bibliotecas Digitales (JBIDI'2001), Almagro (Ciudad Real), Spain, ISBN 84-699-6276-0*, pages 223–234, November 2001.
18. J. Snell, D. Tidwell, and P. Kulchenko. *Programming Web Services With Soap.* O'Reilly & Associates, December 2001.