UNIVERSIDADE DA CORUÑA
DEPARTAMENTO DE COMPUTACIÓN

ARQUITECTURA PARA FEDERACIÓN DE BASES DE DATOS DOCUMENTALES BASADA EN ONTOLOGÍAS

TESIS DOCTORAL

Doctoranda: M. Ángeles Saavedra Places
Directores: Nieves Rodríguez Brisaboa, Miguel Rodrígez Penabad

A Coruña, enero 2003

UNIVERSIDADE DA CORUÑA
DEPARTAMENTO DE COMPUTACIÓN

Arquitectura para Federación de Bases de Datos Documentales basada en Ontologías

Tesis Doctoral
Doctoranda: M. Ángeles Saavedra Places
Directores: Nieves Rodríguez Brisaboa, Miguel Rodríguez Penabad
A Coruña, enero 2003

Tesis Doctoral dirigida por:

Dra. Nieves Rodríguez Brisaboa
Departamento de Computación
Facultade de Informática
Universidade da Coruña
15071 A Coruña (España)
Telf. +34 981 16 70 00 Ext. 1243
Fax: +34 981 16 71 60
brisaboa@udc.es

Dr. Miguel Rodríguez Penabad
Departamento de Computación
Facultade de Informática
Universidade da Coruña
15071 A Coruña (España)
Telf. +34 981 16 70 00 Ext. 1254
Fax: +34 981 16 71 60
penabad@udc.es

PRESIDENTE

Dr. D. ISIDRO RAMOS

VOCAL

Dr. D. FELIX SALTOR

VOCAL

Dr. D. VALENTIN VALERO

VOCAL

Dr. D. MARIO PIATTINI

SECRETARIO

Dr. D. ANTONIO BLANCO

Agradecimientos

Quiero agradecer a mi directora de tesis, la doctora Nieves Rodríguez Brisaboa, no sólo todo el tiempo y el esfuerzo que ha dedicado a este trabajo de tesis, sino también el que me haya dado la oportunidad de trabajar en investigación en el Laboratorio de Bases de Datos. Asimismo, agradezco a mi director de tesis, el doctor Miguel Rodríguez Penabad toda la ayuda que me ha prestado siempre.

A los que han sido mis compañeros en el Laboratorio de Bases de Datos, Fran, Jose (Paramá), Miguel (Luaces), Jose (Viqueira), Eva, Fari, Mon, Toni, Mariajo y Charo, les agradezco todo el ánimo y toda la ayuda que me han dado, y todas las risas que me han sacado.

Además, quiero expresar aquí un agradecimiento especial para mis familias: a mi familia de siempre, Julia, Paco, Manuela (y Carlos), por el apoyo incondicional que me ha dado durante toda mi carrera; y a mi familia nueva, Pedro (y los suyos), por la ayuda que me ha prestado, y por los ánimos que me ha dado para seguir adelante.

Gracias a todas mis amigas y amigos, en especial a Nati y a Rosi, por haberme interrumpido tantas veces, y no haberme dejado trabajar en la tesis, consiguiendo así que este tiempo haya resultado mucho más divertido.

En fin, quiero agradecer el interés, los ánimos y la ayuda que me han prestado todas las personas que han estado a mi alrededor a lo largo de esta tesis.

Índice General

ÍNDICE DE FIGURAS **11**

ÍNDICE DE TABLAS **13**

1 INTRODUCCIÓN **15**

1.1 MOTIVACIÓN 15
1.2 OBJETIVOS 17
1.3 ALCANCE E INTERÉS DEL TRABAJO DE ESTA TESIS 19
1.4 ESTRUCTURA DE LA TESIS 21

2 DESCRIPCIÓN DE NUESTRAS BASES DE DATOS **23**

2.1 INTRODUCCIÓN 23
2.2 LA LITERATURA EMBLEMÁTICA 25
2.2.1 La base de datos Libros de Emblemas *27*
2.2.2 La base de datos Libros de Emblemas Traducidos *32*
2.3 LAS RELACIONES DE SUCESOS 33
2.3.1 La base de datos Relaciones de Sucesos *36*
2.4 IMPLEMENTACIÓN DE LAS BASES DE DATOS 38
2.4.1 Estado actual de las bases de datos *39*
2.5 RESUMEN 40

3 INTERFACES DE USUARIO **41**

3.1 INTRODUCCIÓN 41
3.2 INTERFACES DE USUARIO AMIGABLES 41
3.3 USO DE METÁFORAS COGNITIVAS 43
3.4 FRASES EN LENGUAJE NATURAL ACOTADO (LNA) 46
3.4.1 Versatilidad de la técnica del Lenguaje Natural Acotado (LNA) *48*
3.4.2 Implementación y Esqueletos de frases *52*
3.5 APROXIMACIÓN NAVEGACIONAL 54
3.6 VALIDACIÓN DE LAS TÉCNICAS DE DISEÑO DE INTERFACES PROPUESTAS 57
3.6.1 Descripción del primer prototipo *58*
3.6.2 Descripción del segundo prototipo *59*
3.6.3 Descripción del tercer prototipo *62*
3.6.4 Descripción del cuarto prototipo *63*
3.6.5 Interfaz para la Biblioteca Virtual Gallega *69*
3.7 RESUMEN 70

4 ESTADO DEL ARTE EN APROXIMACIONES PREVIAS: SISTEMAS DE INFORMACIÓN FEDERADOS, Z39.50 Y OAI-PMH 71

4.1 INTRODUCCIÓN 71
4.2 SISTEMAS DE INFORMACIÓN FEDERADOS 71
4.2.1 Tipos de Sistemas de Información Federados 73
4.2.2 Sistemas de Información débilmente acoplados 74
4.2.3 Sistemas de Bases de Datos Federadas 74
4.2.4 Sistemas de Información basados en Mediadores 75
4.2.5 Proyectos relevantes 75
4.3 Z39.50 78
4.3.1 Interoperabilidad de Z39.50 81
4.3.2 Una sesión Z39.50 86
4.3.3 Problemas de Z39.50 88
4.4 OPEN ARCHIVES INICIATIVE 89
4.5 ADECUACIÓN DE LAS TRES APROXIMACIONES EXPUESTAS A LA INTEGRACIÓN DE BIBLIOTECAS DIGITALES 92
4.6 RESUMEN 95

5 ARQUITECTURA DEL SISTEMA DE ACCESO INTEGRADO A TRES BASES DE DATOS DOCUMENTALES Y ÁRBOLES DE CONCEPTOS .. 97

5.1 INTRODUCCIÓN 97
5.2 ARQUITECTURA GENERAL 98
5.2.1 Capa 1: Interfaz de Usuario 99
5.2.2 Capa 2: Mediador 100
5.2.3 Capa 3: Sistemas Envoltorio 101
5.2.4 Capa 4: Bases de Datos 101
5.3 ÁRBOLES DE CONCEPTOS Y ÁRBOLES DE CORRESPONDENCIAS 102
5.3.1 Descripción Árboles 103
5.4 ÁRBOLES DE CONCEPTOS 107
5.4.1 Información para construir la Interfaz de Consulta 108
5.4.2 Información relativa a las bases de datos 109
5.4.3 Árbol de Conceptos del Sistema de Acceso Integrado a las tres bases de datos del Siglo de Oro 109
5.5 ÁRBOLES DE CORRESPONDENCIAS (MAPPINGS) 112
5.5.1 Árboles de Correspondencias del Sistema de Acceso Integrado a las bases de datos del Siglo de Oro 113
5.6 LENGUAJES DE COMUNICACIÓN ENTRE CAPAS 115
5.7 RESUMEN 116

6 IMPLEMENTACIÓN DEL SISTEMA 117

6.1 INTRODUCCIÓN 117
6.2 GENERADOR DE LA INTERFAZ DE CONSULTA 117
6.2.1 Algoritmo del Generador de la Interfaz de Consulta 120
6.3 REPRESENTACIÓN, CONSTRUCCIÓN Y DISTRIBUCIÓN DE CONSULTAS 125
6.3.1 Lenguaje de Consulta 125
6.3.2 El módulo Constructor y Distribuidor de Consultas 126
6.4 TRADUCCIÓN DE LAS CONSULTAS 128
6.4.1 Algoritmo 128
6.5 GESTOR DE LA PRESENTACIÓN 130
6.6 RESUMEN 132

7 FUNCIONAMIENTO 133

7.1 INTRODUCCIÓN 133
7.2 GENERADOR DE LA INTERFAZ DE CONSULTA 133
7.3 REPRESENTACIÓN, CONSTRUCCIÓN Y DISTRIBUCIÓN DE CONSULTAS 136
7.3.1 Otros ejemplos 138
7.3.2 Envío de la consulta a los Sistemas Envoltorio implicados 139
7.4 TRADUCCIÓN DE LAS CONSULTAS 139
7.5 GESTOR DE LA PRESENTACIÓN 143
7.6 RESUMEN 145

8 GENERALIZACIÓN DE LA ARQUITECTURA PROPUESTA PARA FEDERACIÓN DE BIBLIOTECAS DIGITALES BASADA EN ONTOLOGÍAS 147

8.1 INTRODUCCIÓN 147
8.2 FEDERACIÓN DE BASES DE DATOS BASADA EN ONTOLOGÍAS 148
8.2.1 Arquitectura para Federación de Bases de Datos basada en Ontologías 151
8.2.2 Integración de las respuestas usando ontologías 152
8.3 ARQUITECTURA INTEGRAR EL ACCESO DE BIBLIOTECAS DIGITALES MULTILINGÜES 153
8.3.1 Motivación 153
8.3.2 Arquitectura 154
8.3.3 Almacenes de Datos 155
8.3.4 Generador de la Interfaz de Consulta 156
8.4 RESUMEN 157

9 CONCLUSIONES Y TRABAJO FUTURO 159

9.1 INTRODUCCIÓN 159
9.2 CONCLUSIONES Y APORTACIONES PRINCIPALES 159
9.3 LÍNEAS DE TRABAJO FUTURO 163

10 BIBLIOGRAFÍA 165

10.1 REFERENCIAS 165
10.2 PUBLICACIONES DE LA DOCTORANDA DERIVADAS DE LA INVESTIGACIÓN REALIZADA EN ESTA TESIS 169
10.2.1 Publicaciones sobre Interfaces de Usuario 169
10.2.2 Publicaciones sobre Integración 170
10.3 PUBLICACIONES DE LA DOCTORANDA NO RELACIONADAS CON ESTA TESIS 171

Anexos

1 ÁRBOL DE CONCEPTOS DEL SISTEMA 173

1.1 INTRODUCCIÓN 173
1.2 DTD DE LOS ÁRBOLES DE CONCEPTOS 174
1.3 XML DEL ÁRBOL DE CONCEPTOS DE NUESTRO SISTEMA 175

2 ÁRBOLES DE CORRESPONDENCIAS DEL SISTEMA 181

2.1 INTRODUCCIÓN 181
2.2 DTD DE LOS ÁRBOLES DE CORRESPONDENCIAS 181
2.3 XML DE LOS ÁRBOLES DE CORRESPONDENCIAS 183
2.3.1 Base de datos de Libros de Emblemas 183
2.3.2 Base de datos de Relaciones de Sucesos 189

3 ESQUELETOS DE FRASE Y METÁFORAS COGNITIVAS 193

3.1 INTRODUCCIÓN 193
3.2 FRASES EN LENGUAJE NATURAL ACOTADO 193

4 LENGUAJE DE CONSULTA 197

4.1 INTRODUCCIÓN 197
4.2 DTD DEL LENGUAJE DE CONSULTA 197

Índice de Figuras

Fig. 1. Sistema de Acceso Integrado a tres Bibliotecas Digitales 19
Fig. 2. Emblema 26
Fig. 3. Modelo conceptual de Libros de Emblemas 28
Fig. 4. Modelo conceptual de Libros de Emblemas Traducidos 33
Fig. 5. Ilustración de una Relación de Sucesos 35
Fig. 6. Modelo Entidad - Relación 38
Fig. 7. Metáfora de Biblioteca 45
Fig. 8. Metáforas cognitivas: portada de libro 45
Fig. 9. Metáforas cognitivas: libro abierto 46
Fig. 10. Metáforas cognitivas: archivo y fichas 46
Fig. 11. Frases en LNA antes de rellenar los huecos 47
Fig. 12. Frases en LNA después de rellenar los huecos 48
Fig. 13. Frase en LNA para cadenas de caracteres cortas 49
Fig. 14. Frase en LNA para cadenas de caracteres largas 49
Fig. 15. Frase en LNA para atributos multivaluados de tipo cadena de caracteres 50
Fig. 16. Frase en LNA para un atributo de tipo Fecha 50
Fig. 17. Frase en LNA para un atributo de tipo numérico 50
Fig. 18. Frase en LNA para el tipo de dato Texto 51
Fig. 19. Frase en LNA para el uso de tesauros 51
Fig. 20. Frase en LNA para búsqueda aproximada 51
Fig. 21. Frase en LNA para otras técnicas de recuperación de textos 52
Fig. 22. Esqueleto de Frase para cadenas de caracteres cortas 53
Fig. 23. Esqueleto de Frase para cadenas de caracteres largas 53
Fig. 24. Esqueleto de Frase cadenas de caracteres multivaluadas 53
Fig. 25. Esqueleto de Frase para fechas 54
Fig. 26. Esqueleto de Frase para numéricos 54
Fig. 27. Esqueleto de Frase para textos 54
Fig. 28. Aproximación Navegacional en consulta 56
Fig. 29. Aproximación Navegacional en respuesta 56
Fig. 30. Aproximación Navegacional en SIG 57
Fig. 31. Primer prototipo 59
Fig. 32. Segundo prototipo 61
Fig. 33. Tercer prototipo 63
Fig. 34. Página principal de la Biblioteca Virtual de Emblemática 65
Fig. 35. Consulta por título de la obra y autor 65
Fig. 36. Consulta por los principales datos de los emblemas 66
Fig. 37. Búsqueda por contenido 66
Fig. 38. Libro Virtual 67
Fig. 39. Resultados en Google 68
Fig. 40. Arquitectura de un sistema basado en Z39.50 79
Fig. 41. Modelo abstracto de bases de datos 82
Fig. 42. Conversión de los resultados a las Sintaxis de Registro 86
Fig. 43. Sesión Z39.50 88
Fig. 44. Arquitectura de PMH 91
Fig. 45. Sistema de Acceso Integrado a bases de datos documentales 98

Fig. 46. Ubicación de los Árboles de Conceptos y los Árboles de Correspondencias 102
Fig. 47. Un concepto y sus atributos 104
Fig. 48. Relación de Descripción 104
Fig. 49. Relación de Generalización / Especialización 105
Fig. 50. Dos Árboles de Conceptos 107
Fig. 51. Árbol de Conceptos del Sistema de Acceso Integrado 110
Fig. 52. Parte General del Árbol de Conceptos del Sistema 111
Fig. 53. Metáfora Cognitiva y Aproximación Navegacional 112
Fig. 54. Esqueleto asociado al atributo "Lugar de Edición" 112
Fig. 55. Árbol de Correspondencias para Libros de Emblemas 114
Fig. 56. Árbol de Correspondencias para Relaciones de Sucesos 115
Fig. 57. Módulos implicados en la Generación de la Interfaz de Consulta 118
Fig. 58. Esqueleto Frase de Especialización 119
Fig. 59. Ejemplo de Frase de Especialización 119
Fig. 60. Esqueleto de Frase de Descripción 120
Fig. 61. Ejemplo de Frase de Descripción 120
Fig. 62. Diagrama de flujo para la construcción de la Interfaz de Consulta 122
Fig. 63. Ejemplo de pantalla de consulta 123
Fig. 64. Módulos implicados en el envío de las consultas del usuario a las bases de datos 126
Fig. 65. Módulos implicados en la traducción de las consultas 129
Fig. 66. Diagrama de flujo del algoritmo traductor de consultas 130
Fig. 67. Interfaz Web basada en la Metáfora Cognitiva de Biblioteca 134
Fig. 68. Representación gráfica de la consulta de ejemplo 137
Fig. 69. Representación gráfica de la consulta de ejemplo. 139
Fig. 70. Fragmento del Árbol de Correspondencias de Libros de Emblemas 140
Fig. 71. Primera pantalla de respuesta 144
Fig. 72. Interfaz de Respuesta de la base de datos Libros de Emblemas 145
Fig. 73. Ejemplos de Árboles de Conceptos derivados de una ontología 150
Fig. 74. Arquitectura del Sistema de Federación de Bases de Datos basada en Ontologías 152
Fig. 75. Sistema de Acceso Integrado Multilingüe 155

Índice de Tablas

Tabla 1. Taxonomía de Objetos 31
Tabla 2. Características de los Tipos de Sistemas de Información Federados 73
Tabla 3. Clasificación de los Sistemas de Información Federados 73
Tabla 4. Fragmento XML que expresa el primer paso de la consulta 136
Tabla 5. Fragmento XML que expresa el segundo paso de la consulta 136
Tabla 6. XML que expresa la consulta completa 137
Tabla 7. Consulta en XML 138
Tabla 8. Primer paso 141
Tabla 9. Paso dos 141
Tabla 10. Paso 3: Consulta para la bd de Libros de Emblemas 142
Tabla 11. Consulta para la bd Libros de Emblemas Traducidos 143
Tabla 12. Consulta para la bd de Relaciones de Sucesos 143
Tabla 13. Conceptos 155
Tabla 14. Valores de conceptos 156
Tabla 15. DTD de los Árboles de Conceptos 174
Tabla 16. DTD de los Árboles de Correspondencias 181
Tabla 17. Frases en LNA para la Parte General 193
Tabla 18. Frases en LNA para el subárbol de *Relaciones de Sucesos* 194
Tabla 19. Frases en LNA para el subárbol de *Libros de Emblemas* 194
Tabla 20. DTD del Lenguaje de Consulta 197

Capítulo 1

Introducción

1.1 Motivación

Desde su constitución en el año 1995, el Laboratorio de Bases de Datos [41] de la Universidade da Coruña ha venido desarrollando una línea de investigación dedicada a Bibliotecas Digitales. El trabajo en este ámbito ha sido realizado con diferentes equipos de humanidades, como el dirigido por Sagrario López Poza, que está formado por investigadores procedentes de las áreas de Filología española, Historia del Arte y Filología latina.

La colaboración con dicho grupo se materializó en diversos proyectos financiados por la CICYT[1] [42] y la Xunta de Galicia[2] [53]. Gracias a estos proyectos se desarrollaron dos bases de datos (Libros de Emblemas y Relaciones de Sucesos) de documentos del Siglo de Oro español (siglos XVI, XVII y XVIII). Las bases de datos, o Bibliotecas Digitales si se ven desde la óptica de su utilización, almacenan textos, imágenes y una gran cantidad de información que ha sido extraída de los documentos por especialistas de humanidades.

Para cada Biblioteca Digital fue necesario diseñar e implementar Interfaces de Usuario Web lo suficientemente fáciles de usar para que permitiesen realizar consultas más simples o más complejas dependiendo del usuario.

Más recientemente, en un nuevo proyecto CICYT[3] abordado, además de con el equipo de humanidades, con dos equipos de informática de las Universidades de Valladolid y de Vigo, se planteó el desarrollo de una nueva

---

[1] Proyecto con referencia TEL96-1390-C02.

[2] Proyectos con referencias XUGA10504A96 y PGIDT99PXI10502A.

[3] Proyecto con referencia TEL99-0335-C04.

Biblioteca Digital de Libros de Emblemas Traducidos[4], y la integración de las tres Bibliotecas Digitales de documentos del Siglo de Oro. De estas tres Bibliotecas Digitales, las dos primeras son accesibles vía Web en [42] y [53], mientras que la tercera está siendo aún alimentada por los investigadores de humanidades que han necesitado pedir una prórroga de su subproyecto TEL99-0335-C04-01.

Además, a lo largo de los años 2001–2002 se trabajó con el Departamento de Filología Gallega de la Universidade da Coruña en la creación de una Biblioteca Digital de Literatura Gallega [13, 73, 79, 82] en la que también se ha tenido que cuidar la amigabilidad y la facilidad de uso de su Interfaz de Usuario, ya que en dicha biblioteca, dirigida al conjunto de la sociedad gallega, se trataba de maximizar la interacción entre lectores y autores.

Nuestra experiencia en el desarrollo de estos proyectos nos permite afirmar que el diseño e implementación de una Interfaz de Usuario Web para acceder a una Biblioteca Digital es el problema tecnológico fundamental por su impacto en el éxito de dicha Biblioteca, y esta ha sido, precisamente, la primera línea de trabajo abordada en esta tesis. Efectivamente, el diseño, implementación y validación de las diferentes Interfaces de Usuario realizadas para las Bibliotecas Digitales antes citadas, ha constituido la responsabilidad de esta doctoranda durante el desarrollo de estos proyectos.

Por otro lado, en este trabajo de tesis se presenta el sistema desarrollado para integrar el acceso a las tres bases de datos documentales, autónomas e independientes que almacenan, respectivamente, los tres corpus de documentos del Siglo de Oro español ya citados. Dicho sistema está ya terminado pero se encuentra en fase de validación mientras los investigadores filólogos del equipo terminan de validar y alimentar la base de datos de Libros de Emblemas Traducidos. Una vez finalizadas estas tareas, el sistema será accesible vía Web. Estimamos que el tiempo necesario para terminar la validación y alimentación de la base de datos será de entre 6 y 9 meses.

Es necesario destacar que el Sistema de Acceso Integrado a las tres Bibliotecas Digitales del Siglo de Oro se ha realizado a partir de una arquitectura previa, diseñada específicamente para facilitar el acceso integrado a Bibliotecas Digitales. Dicha arquitectura constituye quizá la aportación fundamental de esta tesis.

---

[4] Libros de Emblemas escritos en otras lenguas europeas y traducidos al español durante el Siglo de Oro, y que tuvieron gran influencia en los autores españoles de la época.

1.2 Objetivos

Como ya se ha expuesto, los objetivos de investigación de esta tesis han sido básicamente dos: estudio de estrategias de diseño de Interfaces de Usuario y desarrollo de un Sistema (y su Arquitectura) de Acceso Integrado a tres Bibliotecas Digitales del Siglo de Oro. Además, dicha arquitectura se ha generalizado para que pueda ser usada en la federación de bases de datos documentales utilizando una ontología de descripción del conocimiento almacenado en las mismas.

Interfaces de Usuario

Se ha realizado investigación y obtenido conclusiones sobre cómo diseñar Interfaces de Usuario amigables e intuitivas para el acceso a Bibliotecas Digitales, en particular, y a cualquier tipo de aplicaciones Web, en general.

Como resultado del trabajo realizado proponemos tres técnicas de diseño de Interfaces de Usuario. Se trata de las técnicas del Lenguaje Natural Acotado, la técnica de las Metáforas Cognitivas y el uso sistemático de la Aproximación Navegacional.

Sistema de Acceso Integrado a tres Bibliotecas Digitales

En la Fig. 1 se presenta el diseño básico de la arquitectura del Sistema de Acceso Integrado que se describirá completamente más adelante. En el diseño de este sistema para integrar el acceso a estas tres Bibliotecas Digitales del Siglo de Oro nos hemos preocupado por tres aspectos fundamentales:

- **El sistema debe ser escalable**. El aumento del número de bases de datos que estén integradas no debe bajar el rendimiento global del sistema.
- **El sistema debe acomodar los cambios fácilmente**. Se debe adaptar con gran facilidad a los cambios y evolución continuos típicos de este tipo de bases de datos. Existen dos situaciones que se deben tener en cuenta:
  - Los cambios que se producen por la modificación de las bases de datos ya integradas y por la incorporación de nuevas bases de datos en el sistema, desarrolladas por cualquier otro equipo de investigación, de modo autónomo.
  - Los cambios que se producen en las necesidades de búsqueda de los usuarios, de modo que conceptos, que no habían sido considerados inicialmente, se vuelvan interesantes. De igual forma, el uso del sistema puede llegar a demostrar que conceptos, que en un primer momento se consideraron interesantes para realizar consultas, no lo son en realidad.

- **La Interfaz de Usuario debe ser intuitiva, fácil de usar, potente y flexible** para satisfacer las necesidades de los usuarios. Por un lado, debe permitir realizar consultas a las bases de datos componentes en conjunto o a una base de datos específica. Por otro lado, la interfaz debe adaptarse al nivel de especialización de los usuarios permitiendo consultas muy simples y generales pero también consultas más complejas y específicas para usuarios que sean especialistas en algún dominio. En cualquier caso, el éxito del sistema depende de que los usuarios encuentren la interfaz cómoda y fácil de usar. Indudablemente, este último requisito entra de lleno en la investigación realizada como primer objetivo de esta tesis.

Aunque nos hemos centrado en la realización, diseño e implementación de un Sistema de Acceso Integrado a las tres Bibliotecas Digitales del Siglo de Oro antes citadas. Evidentemente, la arquitectura diseñada es generalizable para la integración de Bibliotecas Digitales de diferente tipo, tal y como se detalla en el Capítulo 8.

Se trata de una arquitectura en cuatro capas, como se muestra en la Fig. 1. La primera capa está formada por las bases de datos y sus respectivos sistemas gestores de bases de datos. La segunda capa está formada por los Sistemas Envoltorio (Wrappers). Existe uno por cada base de datos. Los Sistemas Envoltorio actúan como puentes entre el sistema federado y las bases de datos, haciendo transparentes las diferencias entre las mismas. La tercera capa es el Mediador (o Sistema Federado) y, por último, la cuarta capa es la Interfaz de Usuario (capa conceptual).

La mayor aportación de la arquitectura diseñada para la implementación del Sistema de Acceso Integrado a Bibliotecas Digitales, y que es perfectamente generalizable a otros sistemas que impliquen la federación de bases de datos, es que todo el software del sistema se guía en su ejecución por los Árboles de Conceptos y los Árboles de Correspondencias, que son ficheros XML fácilmente modificables. De este modo, cualquier cambio en las bases de datos ***no supone ningún cambio en el código de ningún módulo*** del sistema, ni su recompilación.

Esta característica hace que la arquitectura pueda proporcionar independencia física y lógica de las bases de datos. Es decir, la arquitectura está preparada para acomodar fácilmente, sin necesidad de realizar ***ninguna recodificación de ningún programa***, cualquier cambio que se produzca, bien en las bases de datos, o bien en las necesidades de información de los usuarios.

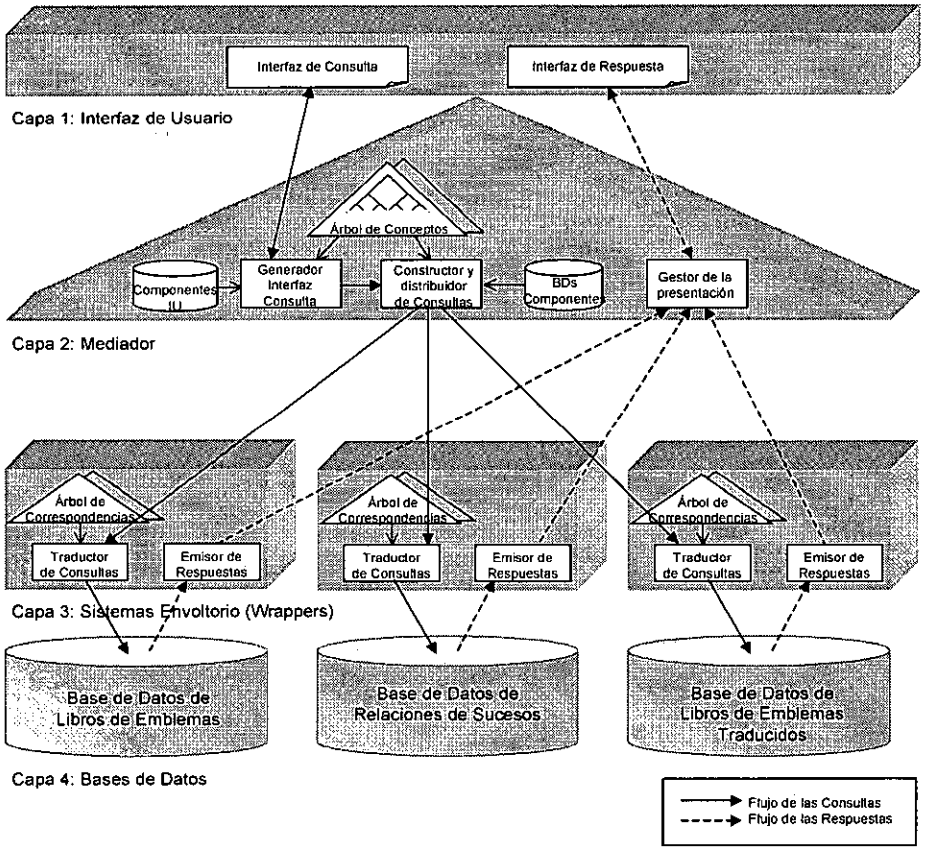


Fig. 1. Sistema de Acceso Integrado a tres Bibliotecas Digitales

1.3 Alcance e interés del trabajo de esta tesis

El problema de integración de fuentes de datos al que hemos tenido que enfrentarnos no es un problema aislado. La enorme expansión de fuentes documentales a la que estamos asistiendo durante los últimos años en Internet hace que existan bases de datos con todo tipo de documentos tales como documentos oficiales, catálogos bibliográficos, documentos literarios, documentos históricos.

Un usuario interesado en un tema específico debe conocer la URL de cada base de datos documental que sea relevante para dicho tema. Además, cada una de estas bases de datos posee su propia interfaz de usuario y su propio lenguaje de consulta. Por lo tanto, el usuario debe adaptar la misma consulta a

las características de cada una de estas interfaces, teniendo que aprender la sintaxis en la que se expresan las consultas para cada una de ellas.

El inconveniente que supone consultar una a una cada base de datos relevante hace imprescindible implementar sistemas que integren varias bases de datos bajo una única interfaz de usuario y a través de una única URL. Sistemas de estas características permitirán al usuario obtener información de todas las bases de datos integradas sin más que escribir una consulta. Serán dichos sistemas los que redirijan las consultas a todas las bases de datos implicadas y guíen al usuario en la navegación a través de las respuestas obtenidas de las distintas bases de datos.

En el diseño de nuestra Arquitectura de Acceso Integrado a Bibliotecas Digitales hemos buscado que fuese generalizable, de manera que se pueda usar para construir sistemas que integren el acceso a fuentes documentales de otros dominios o a bases de datos en general.

Por otro lado, como ya se señaló, consideramos que el diseño de Interfaces de Usuario adecuadas es un problema de gran interés por su impacto en el éxito de cualquier Sistema Web y, en especial, de aquellos destinados al gran público, como Bibliotecas Digitales o Sistemas de Comercio Electrónico. Por ello, creemos que las orientaciones sobre diseño de interfaces web que se ofrecen en esta tesis son una aportación relevante para el diseño de dichos sitemas.

Para justificar el interés de los dos tópicos de investigación abordados en esta tesis (Interfaces de Usuario Web a Bibliotecas Digitales y Acceso Integrado a las mismas) basta señalar que constituyen dos de los tópicos destacados en el informe Bertino. Dicho informe fue elaborado en una importante reunión celebrada en junio de 2001, en la que se dieron cita 23 de los más importantes investigadores internacionales en Bibliotecas Digitales (Bertino, Fox, etc.) con el objetivo de realizar una profunda reflexión sobre los retos que se abrían en el área de cara al Sexto Programa Marco. Como resultado de dicha reunión se elaboró un informe [11] en el que se señala que las líneas de investivación más importantes en Bibliotecas Digitales en los próximos años serán:

- Federación de múltiples fuentes de información
- Acceso "Cross-lingual"
- Acceso a documentos por su contenido
- Interfaces y servicios de usuario
- Construcción de grandes colecciones de datos multimedia

Como veremos, los dos objetivos principales de este trabajo de tesis caen completamente en la primera y cuarta de las líneas de investigación propuestas. Sin embargo, como se verá a lo largo de la tesis, nos hemos preocupado también por facilitar el acceso a documentos por su contenido y por facilitar el acceso a documentos escritos en diferentes lenguas (líneas 2 y 3 de la lista anterior).

1.4 Estructura de la tesis

En el Capítulo 2 se describen detalladamente las tres bases de datos documentales del Siglo de Oro para las que se ha desarrollado el Sistema de Acceso Integrado a través de Web.

En el Capítulo 3, se da cuenta del trabajo realizado en la primera línea de investigación de esta tesis. Es decir, se describen las técnicas de diseño de Interfaces de Usuario amigables para el acceso a Bibliotecas Digitales.

En el Capítulo 4 se describen las tres aproximaciones relevantes para la integración de fuentes documentales heterogéneas: el protocolo Z39.50, la propuesta de *Open Archives Iniciative* y algunos trabajos representativos en federación de bases de datos. Además, se describe lo que dichas aproximaciones aportan y aquello en lo que no se adecuan al problema que nosotros enfrentamos.

Los Capítulos 5, 6 y 7 son los capítulos centrales de la segunda línea de trabajo de la tesis. En el Capítulo 5, se describe de forma global el sistema para proporcionar acceso integrado a nuestras tres Bibliotecas Digitales. Se describe su arquitectura desde el punto de vista de las capas por las que está formada y los módulos de los que se compone cada capa. Se describen también los Árboles de Conceptos y los Árboles de Correspondencias, así como el Lenguaje de Consulta del sistema.

En los Capítulos 6 y 7 se describe detalladamente el funcionamiento de los módulos clave de la arquitectura que presentamos en este trabajo, y se detalla el funcionamiento de los mismos.

En el capítulo 8, se explica cómo se generaliza la arquitectura de nuestro Sistema de Acceso Integrado a nuestras tres Bibliotecas Digitales del Siglo de Oro, para su utilización en la federación de bases de datos basada en ontologías.

Finalmente, en el capítulo 9 se presentan las conclusiones y las líneas de trabajo futuro que quedan abiertas.

Adjuntos a este trabajo de tesis se presentan 4 anexos en los que se dan detalles concretos de implementación que se evitan en las páginas principales de la tesis, manteniendo así en ellas las descripciones y explicaciones en un nivel conceptual para facilitar su lectura. En dichos anexos se describen los DTDs de los Árboles de Conceptos y de Correspondencias; los Esqueletos de Frase y Metáforas Cognitivas concretas usados en el Sistema de Acceso Integrado a nuestras tres Bibliotecas Digitales del Siglo de Oro; y, un cuarto anexo en el que se describe el DTD del Lenguaje de Consulta del sistema.

Capítulo 2

Descripción de nuestras bases de datos

2.1 Introducción

Inicialmente, nuestro interés se centró en proporcionar acceso integrado a tres bases de datos documentales e independientes. Este capítulo está completamente dedicado a su descripción.

Estas tres bases de datos documentales almacenan datos, textos y las páginas digitalizadas de documentos del Siglo de Oro Español (siglos XVI, XVII y XVIII). Estos documentos, aparte de su gran belleza, son una fuente de información magnífica sobre la cultura española del Barroco. En estas obras los investigadores pueden encontrar información sobre la moral, las costumbres, la tecnología, la educación y otros aspectos que son de interés para un amplio rango de investigadores de muy distintas áreas del conocimiento: Antropología, Historia, Sociología, Periodismo, Lengua, Política, Filosofía, etc. Sin embargo, estas obras magníficas son, además, antiguas y valiosas por lo que están dispersas y son de difícil acceso.

Con el objetivo de ofrecer a investigadores, y público en general, la posibilidad de acceder a esta literatura, en el Laboratorio de Bases de Datos de la Universidade de A Coruña [41] se han desarrollado dos proyectos que han dado lugar a dos grandes bases de datos:

- Base de datos de Libros de Emblemas: esta base de datos fue creada gracias al proyecto *Bases de datos y emblemática hispánica bajo Internet*, financiado por la Comisión Interministerial de Ciencia y Tecnología (CICYT ref. TEL96-1390-C02-02). Esta base de datos se encuentra disponible a través de Internet en [42].
- Base de datos de Relaciones de Sucesos: esta base de datos fue creada como principal objetivo del proyecto *Catálogo informatizado de Relacións de Sucesos dos séculos XVI-XVIII en bibliotecas de Galicia e Portugal*,

financiado por la Xunta de Galicia (ref. XUGA10504A96). Esta primera base de datos es un catálogo de Relaciones de Sucesos, es decir, almacena los datos bibliográficos de numerosas Relaciones de Sucesos encontradas en varias bibliotecas de Galicia y Portugal.

Un proyecto posterior *Biblioteca Dixital de Relacións de Sucesos (Séculos XVI-XVIII) en bibliotecas de Galicia e Portugal* que también recibió financiación de la Xunta de Galicia (ref. PGIDT99PXI10502A) permitió ampliar los datos recogidos de las Relaciones de Sucesos con datos resultantes de su análisis literario, artístico y latino, así como con las páginas digitalizadas de las relaciones. Esta gran base de datos de Relaciones de Sucesos está también disponible a través de Internet en [53].

En los últimos meses, se han descubierto nuevas Relaciones de Sucesos en otros países, como Estados Unidos, de manera que la doctora Sagrario López Poza ha solicitado, en la actual convocatoria del Plan Nacional de I+D, un proyecto, en el que incluye a un investigador de nuestro Laboratorio, para ampliar la base de datos de Relaciones de Sucesos, que hoy en día tenemos, tanto en cuanto a su fondo como en cuanto a la cantidad de datos almacenados sobre cada relación.

Un cuarto proyecto, también subvencionado por la CICYT (ref. TEL99-0334-C04-02), ha permitido emprender la creación de una nueva base de datos que contiene Libros de Emblemas escritos originalmente en un idioma distinto al español, pero que ya en su época fueron traducidos por el especial interés que suscitaban. A partir de ahora, nos referiremos a esta base de datos por Libros de Emblemas Traducidos. Actualmente, los datos ya recogidos todavía no han sido validados ni introducidos en la base de datos.

Como ya se ha comentado, dos de las tres bases de datos, Libros de Emblemas y Relaciones de Sucesos, ya están disponibles para consultar a través de Internet. Sin embargo, nuestra intención es construir un sistema que permita a los usuarios acceder a los tres corpus a través de un único punto de entrada, aunque, en realidad, el corpus de Libros de Emblemas Traducidos todavía no esté completo.

Los siguientes subapartados están dedicados a estas tres bases de datos, describiendo para cada una de ellas, tanto la información y los documentos que almacenan, como el modelo conceptual concreto que las soporta.

2.2 La Literatura Emblemática

En el siglo XVI, aprovechando, por un lado, la popularidad de los emblemas, (pictogramas representativos de una idea) como medio de comunicación, y por otro lado la expansión de la imprenta, se comenzaron a editar en toda Europa libros de lo que se ha llamado Literatura Emblemática.

Los libros de Literatura Emblemática trataban de difundir ideas sobre moralidad usando para representar cada principio moral o ético un pictograma, o emblema propiamente dicho, y texto en el que se desarrollaba la idea mediante ejemplos, citas clásicas y bíblicas, etc. A la intención moralizante inicial se le unieron posteriormente otras motivaciones, tales como la divulgación religiosa o la formación de los jóvenes. Así, hay Libros de Emblemas de educación para mujeres, para príncipes, para formación de los futuros clérigos, etc.

En general, los libros de emblemas se dividen en capítulos, también denominados emblemas en este contexto. Cada capítulo–emblema suele estar formado por una imagen (el pictograma) o "emblema" propiamente dicho, por un "epigrama" –pequeño poema– y por un texto en prosa, denominado "glosa", donde el autor desarrollaba la idea o principio moral, citando autoridades clásicas y/o episodios bíblicos que sirvieran como ejemplos, y dieran fuerza a sus argumentos. En la imagen suele aparecer un lema escrito sobre un bando. Este pequeño texto, que frecuentemente aparecía en Latín, se denomina "mote" y en él se trataba de expresar la idea central del emblema.

En su conjunto, la misión de los emblemas era la de fijar en los lectores una idea, un principio moral a adoptar, grabándolo de forma duradera en su memoria. Los tres componentes (imagen, mote y texto – epigrama y glosa) actuaban conjuntamente para producir el efecto deseado. La idea consistía en que el lector trataría primero de interpretar el significado del grabado, partiendo de la pista contenida en el mensaje del mote. El epigrama solía ofrecer, en verso, al estilo de las moralejas de las fábulas, una versión condensada de dicho principio moral, por lo que podía servir al lector para descifrar la profusa simbología barroca de la imagen. Finalmente, se suponía que el lector contrastaría su interpretación con la explicación ofrecida tanto en el epigrama como en la glosa. Tanto en el acierto como en el error, lo que se pretendía se habría logrado: hacer que el mensaje moral llegase a su destino.

En la Fig. 2 aparece una imagen de un emblema en la están ilustradas estas secciones.

Fig. 2. Emblema

En la imagen se puede ver a *"Un hombre joven ataviado a la época, se dispone montando a un caballo en corveta"* (Descripción han hecho nuestros filólogos para este emblema de la base de datos). El mote es una banda que aparece a la derecha de la imagen con el texto *"PARCE PUER STIMULIS"*, que traducen por "*No abuses, muchacho, de las espuelas"*. El epigrama dice:

El mancebo y el potro son briosos,
y más ha menester freno, que espuela,
con poca edad lozanos, y furiosos
en su carrera, el uno, y el otro, vuela.
Fatigadlos, no estén jamás ociosos,
domadlos, en el campo y en la escuela,
el hombre con razón y con doctrina,
y al caballo, con vara y disciplina".

Finalmente en la glosa aparece un texto que en resumen dice: *"El hombre mozo, especialmente si es rico, debe refrenarse y templarse con prudencia. Como el potro, no tiene necesidad de espuela, sino de freno".*

La Literatura Emblemática, aparte de su interés puramente literario, constituye una fuente compleja de información sobre la sociedad, moral, conocimientos o costumbres de los siglos XVI - XVIII, pero, a pesar del gran interés que reviste su estudio y consulta para filólogos, antropólogos o historiadores (estudiosos de la cultura, de las creencias y prejuicios del sistema de valores de aquella época tan compleja de nuestra historia común), resulta realmente difícil acceder actualmente a estas obras ya que son ejemplares valiosos, antiguos y muy dispersos en bibliotecas de monasterios, fundaciones y museos internacionales, donde son objeto de un inflexible cuidado.

La posibilidad de obtener el acceso a los documentos bajo algún tipo de soporte, como por ejemplo microfilmes, o incluso fotocopias, es complicado y, a veces, ciertamente imposible. Por ello posibilitar a los investigadores de tan diversos campos el acceso a una edición digitalizada de dichas obras profusamente indexadas por multitud de campos de búsqueda es de gran interés.

2.2.1 La base de datos *Libros de Emblemas*

Esta base de datos, almacena, por un lado, los libros de emblemas digitalizados y, por otro, una gran cantidad de datos resultantes del análisis de las obras realizado por un equipo de especialistas en Latín, Arte y Filología, entre otras disciplinas.

Entre los datos que se han extraído de los libros de emblemas se encuentran las metáforas más utilizadas para representar determinado concepto, tipos de métrica usados, palabras clave, símiles utilizados, lugares o fechas de impresión más prolíficos, autores clásicos y episodios bíblicos más citados (correctamente o, de forma intencionada, incorrectamente), y la calidad de la traducción del mote latino (frecuentemente tergiversada en la dirección del principio moral del emblema concreto).

En la Fig. 3 se presenta el esquema conceptual de la base de datos Libros de Emblemas.

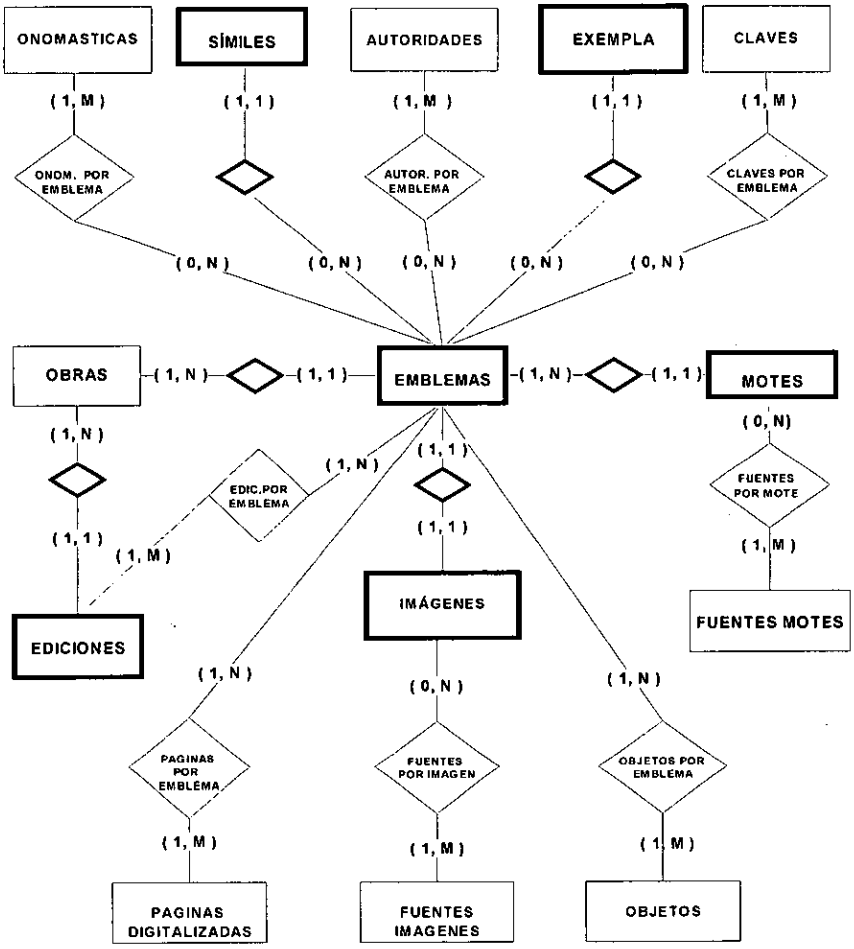


Fig. 3. Modelo conceptual de Libros de Emblemas

Obsérvese que la entidad *Páginas Digitalizadas* almacena, como su nombre indica, las páginas completas, y no sólo los pictogramas. Cada Página Digitalizada puede corresponderse a una página o a una doble página del libro, según el tamaño del mismo. Debido a que cada emblema puede ocupar un número indeterminado de páginas es necesario relacionar a cada emblema con sus páginas para poder ofrecer a los usuarios, no sólo la posibilidad de leer un libro completo página a página, sino también la posibilidad de leer las páginas correspondientes a emblemas seleccionados mediante cualquier conjunto de condiciones de búsqueda.

Se describen a continuación algunas entidades de interés agrupadas según se utilicen para almacenar datos filológicos, datos de la imagen o datos del mote.

Descripción de las relaciones con datos filológicos:

Las entidades en las que se almacenan datos de carácter filológico son las que hemos llamado *Obras*, *Ediciones*, *Emblemas* y *Emblemas por edición*. Dichas entidades dan lugar en la base de datos a las correspondientes relaciones que, lógicamente, identificamos con el mismo nombre.

- *Obras:* Esta relación almacena datos relevantes de los libros, como: el autor, el título o la *Editio Optima* (fecha de la edición tomada como representativa)
- *Ediciones:* Una obra pudo haber sido objeto de diversas ediciones con el paso del tiempo. Esta entidad almacena los datos relevantes de las ediciones, como: año de edición, lugar de edición, promotor, editor o impresor.
- *Emblemas:* Esta relación almacena los datos de cada capítulo o emblema: el epigrama completo, un resumen del epigrama, número y tipo de versos, estrofa, idioma del epigrama, idioma del mote y un resumen de la glosa.
- *Emblemas por edición:* Esta relación almacena datos propios de cada emblema que podían variar de una edición a otra de la obra. Entre dichos datos cabe destacar: el nombre del diseñador que es el artista que diseñó la imagen; nombre del grabador, artista que realizo el grabado; o posición que ocupa el emblema en la edición. Este último atributo tiene sentido porque con frecuencia en las diferentes ediciones los emblemas eran reordenados. Es frecuente, incluso, que algunos emblemas no aparezcan en todas las ediciones de una obra, lo cual se refleja también en esta relación. Este dato es de gran interés pues, habitualmente, se debía a la censura de la Inquisición.
- *Símiles:* En esta relación se recogen las metáforas utilizadas por el autor en cada emblema. Se almacena la metáfora en sí y su significado.
- *Exempla:* Recoge los ejemplos expuestos por el autor para explicar la idea que trata de transmitir con el emblema.

Aparte de estas relaciones que hemos descrito, en la base de datos de emblemas tenemos unas tablas que se constituyen auténticos *Thesaurus* útiles para realizar las búsquedas. Obsérvese que cada una de estas relaciones contiene un único campo. En realidad, más que constituir relaciones propiamente dichas, pueden considerarse relaciones derivadas, pues pueden

extraerse por proyección del dato implicado de las relaciones siguientes: *Onomásticas por emblema*, *Autoridades por emblema* y *Claves por emblema*, respectivamente. Sin embargo, hemos decidido mantenerlas en el modelo de datos como entidades independientes para mayor claridad.

- *Índice onomástico:* Nombres propios que aparecen en cualquier lugar del emblema. Suelen referirse a personajes míticos y/o bíblicos, y también a topónimos.
- *Autoridades:* Citas realizadas por el autor. No se trata sólo del nombre de la autoridad citada sino de la referencia completa que aparece.
- *Palabras clave:* Lista de palabras clave.

Descripción de relaciones con datos sobre la imagen:

Los datos almacenados en estas tablas son fruto del estudio que especialistas en arte han realizado de los libros de emblemas. Las relaciones que se usan para recoger los datos relativos a las imágenes o pictogramas son *Imágenes*, *Fuentes de la imagen* y *Objetos*. Estas tablas se describen a continuación:

- *Imágenes:* Recoge datos sobre cada una de las imágenes o escenas que aparecen en un emblema o pictograma. Se almacena el número de imagen, el motivo (o descripción de la escena) y su significado (interpretación que el autor buscaba que se le diera a la imagen).

Además, existen otras dos relaciones que almacenan datos sobre las imágenes y que también constituyen auténticos *Thesaurus*. Obsérvese que más que constituir relaciones propiamente dichas podrían extraerse por proyección del dato implicado de las relaciones de *Fuentes por imagen* y *Objetos por imagen*, respectivamente.

- *Fuentes de la imagen:* Se almacenan en esta relación los antecedentes que se han encontrado de la imagen. En ocasiones dichos antecedentes son citados por el autor, en otras, si existen, han sido especialistas en arte los que las han proporcionado. Por otro lado, se almacena el tipo de las fuentes: Emblemáticas, Iconográficas o Literarias.
- *Objetos:* Mantiene una lista de objetos que aparecen en cada imagen y su clasificación en una taxonomía de dos niveles creada por los expertos en arte para este fin. Una muestra de esta taxonomía de más de 6300 objetos se muestra en la Tabla 1.

Tabla 1. Taxonomía de Objetos

Clase	Subclase	Objeto
...	...	...
Animales		
	Aves	Cuervo
		Fénix
		Funicuro
...	...	...
Humano		
	Armas	Abrojo
		Arcabuz
		Arco
	Embarcaciones	Mástil
		Nao
		Navío
...	...	...
Mitología		
	Atributos	Flecha rota
		Guirnalda de hojas
		Guirnalda de rosas
...	...	...

Descripción de relaciones con datos sobre el mote:

El estudio de los motes en sí mismo reviste gran interés. Los errores ortográficos y sintácticos que presentan, tanto el latín original como su traducción al español, a veces con gran tergiversación del significado real de la frase latina, pueden ser interpretados en ocasiones como indicativos del nivel cultural del autor, pero, en otras ocasiones, como intentos claros de manipulación, que indican que los autores eran conscientes de que los lectores ya no eran capaces de entender el texto latino. Para almacenar los datos relativos a los motes tenemos dos tablas:

- *Mote:* Esta relación almacena el mote propiamente dicho, así como si su escritura es correcta o no. Es necesario almacenar también si el autor reescribe o no el mote en el texto y cómo lo reescribe: si lo traduce del latín, y, en ese caso, si la traducción es correcta o no. En caso de que su escritura latina o su traducción hayan sido incorrectas, se almacena además el mote correctamente escrito en latín y correctamente traducido.

  Los campos que utilizamos para almacenar esta información son: el identificador del tipo de mote, el mote (original de la imagen, traducido por el autor o por nuestro especialista en latín, etc. dependiendo del código que se le asocie al tipo de mote), correcto/incorrecto (Este campo lógico sólo utilizado cuando el mote es escrito o traducido por el autor.

Lógicamente, si la escritura latina o la traducción es realizada por nuestro especialista, el campo deberá adoptar el valor verdadero siempre).

- *Fuentes del mote:* En esta relación se describen los antecedentes de la frase que constituye el mote. Se almacena entonces el identificador del mote y la fuente (cita a los textos anteriores al emblema en los que se puede encontrar esa misma frase) y el tipo de fuente, que indica si la fuente fue citada por el autor en el texto y es correcta, si la fuente fue citada por el autor en el texto pero incorrectamente o si la fuente la proporciona nuestro especialista.

2.2.2 La base de datos *Libros de Emblemas Traducidos*

Estos libros de emblemas, en particular, fueron muy populares en su momento, de manera que su estudio es muy interesante, porque han inspirado a otros autores españoles para escribir sus propios libros de emblemas. Conceptualmente, la base de datos Libros de Emblemas Traducidos y Libros de Emblemas son muy parecidos. Ambas almacenan datos, texto y documentos de obras del mismo tipo.

En la Fig. 4 se muestra el esquema conceptual de la base de datos Libros de Emblemas Traducidos. A primera vista, en esta base de datos se almacenan los mismos datos que en la base de datos Libros de Emblemas. Sin embargo, a pesar de que los conceptos que almacenan son los mismos, existen ciertas diferencias que analizamos a continuación:

- *Epigrama:* en los libros de emblemas traducidos cada emblema o capítulo podía tener varios epigramas, a diferencia de los libros de emblemas españoles que siempre tenían únicamente uno.
- *Similes*, *Objetos*: no se recogen datos sobre estas relaciones.

El interés que tiene el estudio de estos libros es permitir rastrear su influencia en los autores españoles de la época. Es decir, para estudiar ciertos Libros de Emblemas españoles es muy interesante saber cuáles eran los Libros de Emblemas (o de otro tipo) a los que el autor tuvo acceso para inspirarse al escribir su obra. Esta es precisamente la función que cumple esta biblioteca digital.

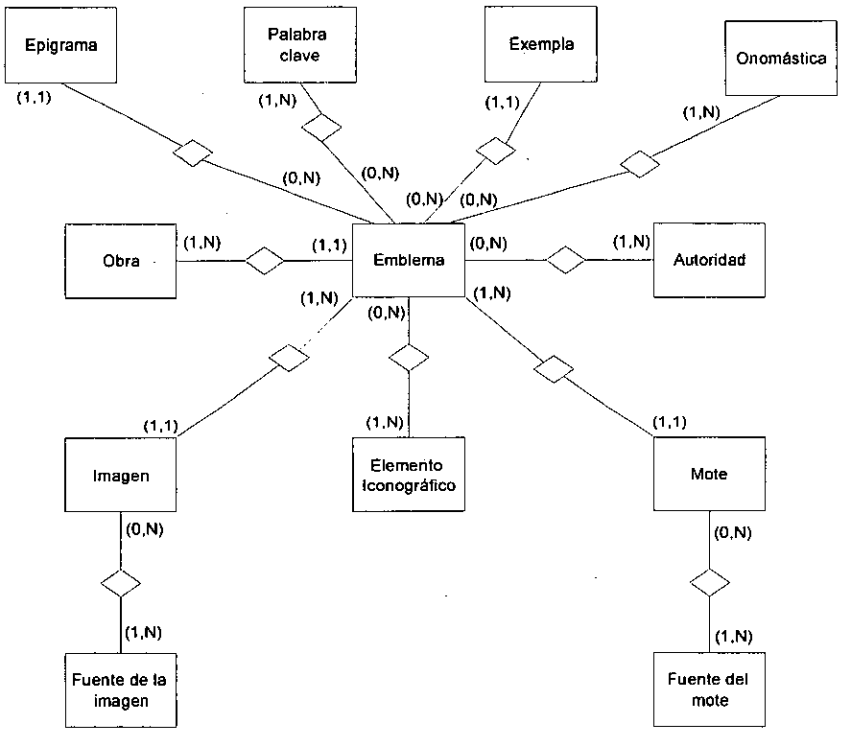


Fig. 4. Modelo conceptual de Libros de Emblemas Traducidos

2.3 Las Relaciones de Sucesos

Las *Relaciones de Sucesos* son documentos que narran un acontecimiento ocurrido o, en algunas ocasiones, inventado, con el fin de informar, entretener y conmover al público –bien sea lector u oyente–. Tratan de muy diversos temas, entre otros: acontecimientos histórico–políticos (por ejemplo, guerras o autos de fe), sucesos monárquicos, fiestas religiosas o cortesanas, viajes o sucesos extraordinarios como catástrofes naturales, milagros o desgracias personales.

Su forma es también variada: pueden ser manuscritas o impresas, estar en verso o prosa, y constar de un solo pliego (la mayor parte tienen esta forma de pliego suelto compuesto por dos o cuatro hojas) o llegar a tener las dimensiones de un libro voluminoso.

Las *Relaciones de Sucesos* surgen en el siglo XV vinculadas al género epistolar: la carta-relación, que informa generalmente a un particular de algún

acontecimiento del que fue testigo el emisor. Su uso se va extendiendo en el siglo XVI, en el que aparece ya la *Relación de Sucesos* de forma autónoma (aunque convivirá siempre con la carta) dirigida a un público más amplio, para alcanzar su apogeo en el siglo XVII, sobre todo en los reinados de Felipe IV y Carlos II. Su desaparición vendrá condicionada por el nacimiento y éxito de las Gacetas, ya en el siglo XVIII, que amplían el mundo informativo al contar las noticias periódicamente, y no de manera ocasional como lo hacían las *Relaciones*. Son, por tanto, las auténticas antecesoras del género periodístico actual y ya en las Relaciones de Sucesos se escribían los antecedentes de cada uno de los subgéneros del periodismo que hoy existen.

La existencia de *Relaciones* se constata en toda Europa, si bien su producción decae con el auge de las gacetas, mientras que en España y sus dominios la forma característica de *relación*, como relato de un acontecimiento ocasional y no periódico, perdura largamente en convivencia con el nuevo género.

Las *Relaciones de Sucesos* pueden clasificar, según su contenido, en:

- *Sucesos Histórico-Políticos*: Narran sucesos sobre la historia o la política de un estado, o de sus monarcas (guerras o celebraciones de autos de fe, por ejemplo) generalmente satirizándolos. Este tipo de relaciones se puede asociar con la actual Prensa Política.
- *Sucesos Festivos*: Narran acontecimientos, monárquicos o religiosos, que dan lugar a fiestas (por ejemplo, bodas reales, bautizos, exequias, beatificaciones). Muchas se proponían halagar a la familia real, o los agasajados en la fiesta, probablemente con la intención de ensalzar sus esfuerzos y gastos. Equivalen a la actual Prensa del Corazón.
- *Sucesos Extraordinarios*: Relataban sucesos milagrosos (como apariciones), desgracias provocadas por la naturaleza o sucesos extraños (como el nacimiento de monstruos). Estos temas son los que, hoy en día, son tratados por la Prensa Sensacionalista.
- *Sucesos Taurinos:* Relataban lo acontecido durante las corridas de toros que se celebraban. Son los antecesores de la actual Prensa Deportiva.
- *Viajes*: Son un tipo especial de *Relaciones*, donde se narran acontecimientos vividos por personas que han viajado y que lo cuentan sobre todo a través de cartas.

En la Fig. 5 se muestra una ilustración de una Relación de Sucesos titulada: *"Relación: y copia de carta escrita por un caballero residente en la ciudad de Paris a otro correspondiente suyo en esta Corte, en que le da cuenta del Monstruoso Pez, que hallaron unos Pescadores en el Rio Sena de*

Francia el día 16 de Enero de 1684. Refiere la maravillosa forma y señales que tenía."

Fig. 5. Ilustración de una Relación de Sucesos

A través de estos documentos se reflejan muchos aspectos de la cultura de la Edad Moderna europea. Se trata de un material de valor inapreciable para los estudiosos de la Historia, Literatura, Historia de las mentalidades, Antropología, Historia del Arte, Sociología y muchos aspectos de la cultura del Siglo de Oro, incluida la imprenta y la tecnología de la edición. A pesar de que la dudosa calidad literaria de estas *Relaciones* haya justificado hasta ahora la escasez de investigaciones profundas sobre ellas, la variada y rica información que ofrecen invita al estudio de estos documentos.

Existieron muchísimas *Relaciones* y bastantes se han conservado hasta nuestros días. Pero, pese a ser numerosas, se encuentran muy dispersas entre instituciones, monasterios y Bibliotecas de fondo antiguo por lo que, en la gran mayoría de los casos, son de muy difícil acceso o de acceso restringido. Por ello, es, lógicamente, de gran interés contar con una auténtica biblioteca

digital de Relaciones de Sucesos como la que estamos permanentemente construyendo[5].

En los siguientes subapartados se describen los datos almacenados actualmente en la base de datos Relaciones de Sucesos.

2.3.1 La base de datos *Relaciones de Sucesos*

Después de un acercamiento a las *Relaciones de Sucesos,* y un estudio detallado de la información que contienen, por parte de los expertos en Filología, y tras múltiples sesiones de trabajo conjuntas, se elaboró el modelo conceptual de la base de datos cuyo esquema Entidad–Relación se muestra en la Fig. 6.

A continuación, se describen las relaciones implementadas y se mencionan sus atributos principales:

- *Relación de Sucesos*: Cada tupla recoge información general de una *Relación de Sucesos* (su identificación, título completo, autor, localización donde tienen lugar los hechos narrados, si está en prosa o en verso, el género y subgénero donde se engloba, etc.)
- *Catálogos Previos:* Resulta de interés, para posteriores estudios, conocer si las *Relaciones* que se estudian en este proyecto ya han sido catalogadas previamente en otros catálogos bibliográficos o de *Relaciones*. Esta tabla guarda información sobre dichos catálogos.
- *Catalogaciones por Relación:* Asocia cada *Relación* con el catálogo(s) donde ya ha sido referenciada indicándose, además, la identificación que se le ha asignado en el catálogo concreto.
- *Ilustración:* Cada edición puede contener una ilustración característica. En esta tabla se guarda información que describe textualmente en que consiste la misma.
- *Edición:* Esta tabla recoge información general sobre las ediciones publicadas de cada *Relación de Sucesos* estudiada. Atributos de esta tabla son el editor, promotor, impresor, lugar de edición, idioma en el que se ha escrito (en aquella época el latín aún era una lengua muy utilizada), su tamaño (folio, gran folio, etc.), si se trata de una edición manuscrita o

[5] Como ya hemos dicho, la actual biblioteca digital de Relaciones de Sucesos está formada por un catálogo completo que se está ampliando con datos resultantes del estudio de las relaciones desde diferentes puntos de vista (literario, artístico e histórico), y que se pretende seguir ampliando. Tanto es así que ha sido solicitado un proyecto que ayude a financiar dicha ampliación.

impresa, nombre de la persona(s) a la(s) que se dedica la edición, tipo de portada (orlada, historiada o grabada), etc.

- *Ejemplar:* Esta tabla contiene los datos referentes a los ejemplares de que se dispone en una edición concreta de cada *Relación de Sucesos*. Para cada ejemplar se guarda información sobre la signatura dada por la biblioteca donde se encuentra en la actualidad, tipo de encuadernación (pergamino, holandesa, pasta o rústica), nombre de la persona o biblioteca a la que perteneció en primer lugar el ejemplar, etc.
- *Biblioteca:* Contiene datos generales sobre las diferentes bibliotecas que guardan las *Relaciones de Sucesos* estudiadas (sus nombres, direcciones, teléfonos, si son públicas o privadas, etc.)
- *Página Digitalizada:* Los ejemplares han sido digitalizados. Cada página o doble página está digitalizada en un fichero independiente. Dichos ficheros se almacenan como campos BLOB en la Base de Datos general.
- *Composición:* En las Relaciones de Sucesos festivas suelen aparecer todos los géneros que surgían en la fiesta que describe la relación. A cada uno de estos géneros se le llama Composición. En la base de datos se incluyen los poemas, los sermones y las piezas teatrales:
  - *Poema:* almacena el autor, el tipo de estrofa y el tipo de verso del poema.
  - *Sermón:* almacena las características más relevantes de este tipo de composiciones, como son: quién pronuncia el sermón, dónde y cuando se pronuncia, y el motivo de dicho sermón.
  - *Pieza teatral:* se almacena el autor, el título y los personajes que intervienen.
- *Estampa:* se almacena su título, su descripción, la posición, el inventor, el pintor, el grabador, etc.
- *Referencia Bibliográfica:* los datos más importantes que almacena son el lugar, año, editorial, etc. de cualquier tipo de referencia bibliográfica.
- *Thesaurus de Epítetos:* Los epítetos son adjetivos, de gran interés para los filólogos, que forman parte del título de la *Relación* y que eran empleadas por el autor para convencer al lector de que dicha *Relación* era la más "verídica" y "autentificada". En esta tabla se guardan todos los epítetos que aparecen en los títulos de las *Relaciones* estudiadas.
- *Epítetos por Relación:* Asocia las relaciones con los epítetos que contiene y que se encuentran en la tabla anterior

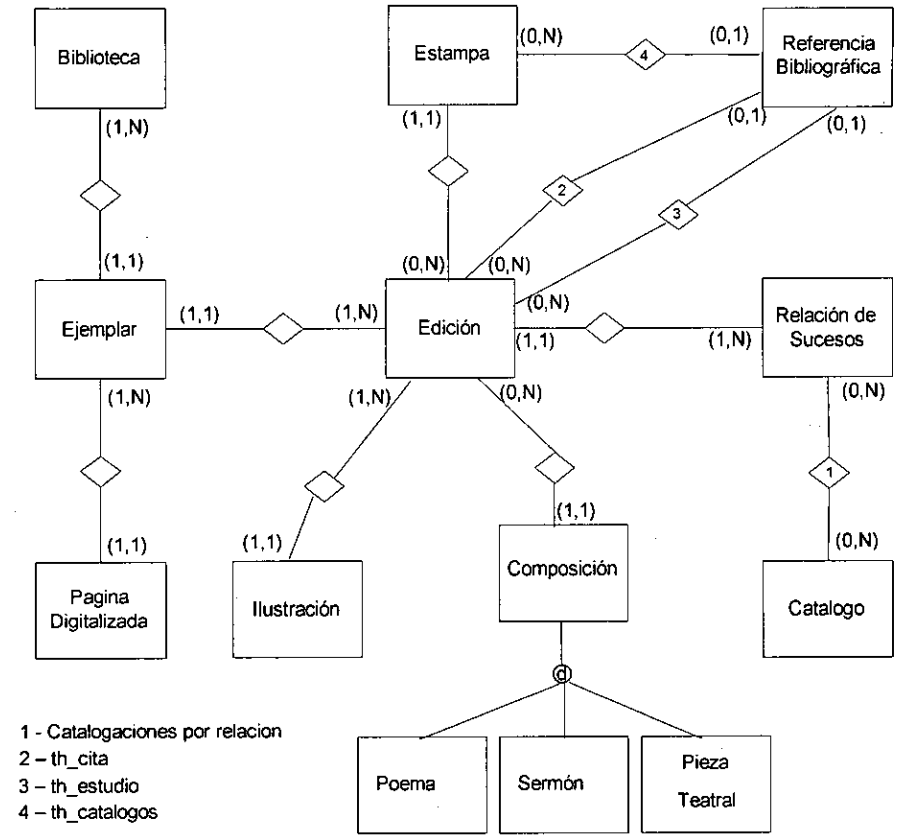


Fig. 6. Modelo Entidad - Relación

2.4 Implementación de las bases de datos

Las tres bases de datos fueron creadas, inicialmente para Informix Online Dynamic Server (versión 7) para Unix. Debido a problemas encontrados en el manejo de campos `blob`, que almacenaban campos de texto (los campos blob en Informix no permiten el uso del operador `like`), la base de datos Relaciones de Sucesos se migró a Borland Interbase Server. Sin embargo, debido a que hemos querido dotar con capacidades de Recuperación de Textos al sistema, las tres bases de datos van a ser migradas a Oracle 9i, porque este sistema gestor de bases de datos incluye la cláusula ***contains***, que es el estándar que SQL:1999 incluye para Recuperación de Textos.

Es necesario comentar, que el DataBlade *Excalibur* de Informix también ofrece una implementación de la cláusula contains, pero, en nuestra opinión,

la implementación que hace Oracle es más completa y eficiente, además de ofrecer ciertas capacidades de búsqueda que Informix no contempla [101].

2.4.1 Estado actual de las bases de datos

Las tres bases de datos han pasado (o están actualmente) por un proceso que va desde el diseño de la base de datos hasta la introducción de los datos, ya extraídos (de las aplicaciones locales de los portátiles de los especialistas en Filología, Arte y Latín) y validados, en la base de datos final accesible en Internet a través de una aplicación Web.

Base de datos de Libros de Emblemas

Esta es la única base de datos que en estos momentos está completa. El diseño de la base de datos fue realizado en el Laboratorio de Bases de Datos antes de mi incorporación, así como el desarrollo de las aplicaciones de alimentación de las bases de datos locales para portátiles de los investigadores.

En cuanto me incorporé al laboratorio entré a formar parte del equipo encargado de la validación de los datos almacenados en las bases de datos locales, y de su volcado a la base de datos centralizada.

El trabajo realizado en mi proyecto de tesina tuvo como principal objetivo el diseño e implementación de la actual interfaz Web de la base de datos, la Biblioteca Virtual de Literatura Emblemática [11].

Base de datos de Relaciones de Sucesos

Recientemente, se ha planificado la ampliación de esta base de datos, tanto de su fondo, como de la cantidad de información recogida de cada relación, así como de la Interfaz de Usuario para el acceso Web a la misma.

Actualmente, se está llevando a cabo el desarrollo de una nueva Interfaz Web (de la cual soy responsable) que permite acceder a los datos resultantes del Análisis Literario, Histórico y Artístico que están siendo volcados a la base de datos central de Relaciones de Sucesos.

Base de datos de Libros de Emblemas Traducidos

Esta base de datos está ahora en fase de validación de los datos de las bases de datos locales y alimentación de la base de datos centralizada.

He participado en todas las tareas relacionadas con la creación de esta Biblioteca Digital. Me he encargado del desarrollo de las aplicaciones locales

y soy responsable del diseño e implementación de la Interfaz Web de esta base de datos, que está siendo construida siguiendo las ideas de diseño de Interfaces de Usuario fruto del trabajo de investigación desarrollado en el marco de esta tesis.

2.5 Resumen

En este capítulo se han descrito las tres bases de datos que, inicialmente, están integradas en el Sistema de Acceso Integrado a bases de datos documentales.

Como se ha visto, estas bases de datos almacenan textos transcritos, páginas digitalizadas y una gran cantidad de datos estructurados, fruto del análisis de los documentos por especialistas en distintas áreas de la filología, sobre documentos de los Siglos de Oro español (XVI-XVIII).

Los Libros de Emblemas y las Relaciones de sucesos, aun siendo todos ellos documentos antiguos, son colecciones muy heterogéneas. Como se verá más adelante, esta heterogeneidad ha sido resuelta por el sistema que construimos, y las soluciones allí aplicadas pueden ser fácilmente generalizadas a la federación de bases de datos en general.

Capítulo 3

Interfaces de Usuario

3.1 Introducción

Los sistemas Web son especialmente dependientes de la calidad de la Interfaz de Usuario que presenten. Es bien sabido que los usuarios Web no van a leer ningún manual de utilización y que apenas harán uso de las "ayudas". Por tanto, es necesario que sepan usar el sistema desde el primer momento y, por ello, las interfaces deben ser muy sencillas e intuitivas.

En este capítulo se aborda el primer objetivo planteado en esta tesis, el diseño e implementación de Interfaces de Usuario intuitivas que propicien el éxito de cualquier sistema Web. En este sentido se presentan tres técnicas de diseño de interfaces de usuario que son útiles para construir Interfaces de Usuario (tanto de consulta como de respuesta) a bases de datos en general y, especialmente, a bases de datos documentales. Así lo demuestra la aceptación que han tenido las interfaces Web que hemos desarrollado usando dichas técnicas [11, 13], además de diversos estudios realizados en el campo de la Interacción Persona-Ordenador (Human Computer Interaction) [52].

3.2 Interfaces de Usuario Amigables

La "usabilidad" de la Interfaz de Usuario es un aspecto crucial de una aplicación Web. El diseño de "buenas" interfaces de usuario, es decir, que sean amigables y fáciles de utilizar por los usuarios, ha sufrido una enorme transformación en los últimos años. Desde la opinión generalizada de que la interfaz de usuario es una "capa superficial" que se añade al diseño de una aplicación en el último momento, se ha pasado a integrar el diseño de la interfaz con el diseño de la aplicación global.

Lo que ha hecho que se le dé tanta importancia a un buen diseño de la Interfaz de Usuario es que, si los usuarios no encuentran el sistema amigable y cómodo para realizar las tareas que deben hacer, no lo usarán. El Dr. Donald Norman ilustra acertadamente en su libro *The Psichology of Everyday Things* [48] un aspecto a tener en cuenta en el diseño de interfaces: "Cuando las cosas simples necesitan etiquetas o instrucciones, el diseño ha fallado". C. Zetie [67] ilustra con un ejemplo otro aspecto importante de las Interfaces de Usuario: La gente suele decir "voy a lavar la ropa", no "voy a usar la lavadora", mientras que dice "voy a programar el vídeo" en vez de "voy a grabar una película". Este ejemplo nos muestra que un buen diseño de interfaz en una aplicación informática hará que los usuarios *realicen una tarea*, y no que *usen la aplicación*.

Para conseguir este ambicioso objetivo, y como resultado del trabajo de investigación desarrollado en esta tesis, se propone el uso sistemático y simultáneo de tres técnicas, alguna de las cuales procede del campo de *Human Computer Interaction*, mientras que otras han sido desarrolladas en el curso de este trabajo de investigación. Estas técnicas son el uso de Analogías o Metáforas Cognitivas, la técnica del Lenguaje Natural Acotado y la Aproximación Navegacional, que hemos presentado también en [71, 72, 72] y consisten en:

- Uso de ***Metáforas Cognitivas***. Consiste en construir páginas Web usando escenarios del mundo real, de manera que los elementos representados en la página Web sean similares, tanto en aspecto como en funcionamiento, a los elementos del mundo real. Esta técnica es muy conocida y utilizada, no sólo en Interfaces de Usuario Web, sino en cualquier sistema informático. Un buen ejemplo, es la Calculadora de Windows [47].

- Uso de ***Lenguaje Natural Acotado***. Esta técnica consiste en ofrecer al usuario un conjunto de frases (en lenguaje natural) con huecos. El usuario debe elegir las frases en las que está interesado y llenar los huecos. Finalmente, el conjunto de frases seleccionadas con los huecos cubiertos representa la consulta del usuario.

  Esta técnica ha sido desarrollada en el marco de este trabajo de tesis. Su gran utilidad ha provocado que haya sido objeto de numerosas publicaciones [71, 72, 72] y que, a su vez, han sido citadas por otros investigadores españoles [25, 26, 27].

- ***Aproximación Navegacional***. Esta aproximación es útil tanto en las Interfaces de Consulta como en las de Respuesta. En las Interfaces de Consulta, en vez de presentar al usuario formularios en los que, para expresar condiciones sobre cada atributo, tenga que introducir el valor que ha de tener dicho atributo, la Aproximación Navegacional permite

expresar dichas restricciones a través de simples clicks de ratón sobre una pantalla (normalmente creada a partir de una Metáfora Cognitiva) en la que están explícitos los posibles valores que puede tomar un atributo o conjunto de atributos. Del mismo modo, al usar esta aproximación, se puede evitar que el usuario introduzca el valor de un cierto atributo si las respuestas se presentan ordenadas por dicho atributo, de modo que el usuario pueda fácilmente navegar por el conjunto de resultados y localizar aquellos de su interés.

Por otro lado, también hemos aplicado la idea de diseñar Interfaces con diferentes niveles de complejidad para diferentes tipos de usuario. Es decir, por un lado se construyen Interfaces sencillas que permiten expresar consultas muy simples y, por otro lado, Interfaces con un nivel de dificultad sólo un poco más alto que, en cambio, permiten expresar consultas más complejas. Como veremos al final de este capítulo, nuestra experiencia en el desarrollo y uso de Interfaces de Usuario nos ha permitido concluir que la construcción de diferentes Interfaces de Usuario para usuarios "comunes" y usuarios "expertos" garantiza un mayor éxito en la difusión y uso de una Biblioteca Digital.

A continuación se explica el uso de las tres técnicas de diseño de Interfaces de Usuario y las ventajas que cada una de ellas proporciona. Se debe destacar que no se trata una metodología de diseño de Interfaces de Usuario, sino que son técnicas que, usadas de modo sistemático y combinado, permiten mejorar el aspecto y la facilidad de uso de cualquier Interfaz Web.

3.3 Uso de Metáforas Cognitivas

Esta técnica está basada en el uso de algo conocido por el usuario que es trasladado a otro dominio. En algunos casos está basado en la similitud del aspecto físico y en otros casos en la similitud de los objetivos y tareas que lleva a cabo. Los primeros ejemplos de Metáforas Cognitivas fueron los procesadores de textos usando como metáfora una máquina de escribir o las carpetas o directorios de ficheros que usan una carpeta de archivador como metáfora. El primer ejemplo se basaba en la similitud del aspecto (el usuario ve una hoja de papel en la que hay que escribir), mientras que el segundo ejemplo se basa en la similitud de los objetivos que se consiguen tanto con los ficheros como con las carpetas, almacenar y recuperar información.

Una de las metáforas más conocidas y usadas es la calculadora que está incluida en todos los sistemas operativos Windows. Probablemente exista una forma más eficiente de desarrollar una interfaz para realizar operaciones matemáticas, pero no sería tan intuitiva y fácil de usar. Esto es debido a que la

similitud entre una calculadora real y la Interfaz de Usuario que ofrece este programa Windows permite que los usuarios usen el conocimiento que ya tienen sobre cómo usar calculadoras en el mundo real para manejar la calculadora de Windows.

En cuanto a la Web, una de las metáforas más exitosas es el extendido *carrito de la compra* que se usa en la mayoría –sino en todas– las tiendas on-line. El éxito de las tiendas on-line, y, por lo tanto, el éxito de las metáforas que querían imitar los carritos de la compra, es una buena indicación de las ventajas de usar una buena metáfora.

Para el acceso a bases de datos documentales, consideramos necesario construir Interfaces de Usuario que usasen esta aproximación tanto sea posible. A continuación ofrecemos varias metáforas diseñadas y desarrolladas en el marco de esta tesis, que son útiles en la implementación de Interfaces de Usuario a bases de datos documentales, y que han sido usadas en nuestras bibliotecas.

Por ejemplo, en la Fig. 7 se muestra una Metáfora Cognitiva que representa una biblioteca real. Esta Metáfora está siendo usada en la interfaz de la Biblioteca Virtual de Literatura Emblemática [42]. El desarrollo de esta interfaz fue el objetivo principal de mi tesina y ha sido objeto de varias publicaciones [68, 69, 83].

El usuario de la página Web, construida a partir de esta metáfora, sabrá, sin necesidad de explicación alguna, que pinchando en

- el tablón que aparece a la izquierda podrá acceder al tablón de las últimas novedades de la biblioteca,
- en la puerta podrá acceder a un foro de discusión,
- en la mesa, podrá inscribirse como socio de la biblioteca,
- en el archivador podrá ver un catálogo,
- y pinchando en las estanterías podrá acceder a las obras de la temática especificada en el cartel de la estantería elegida.

Esto es posible debido a que esta página representa un entorno familiar para los usuarios. Los usuarios asocian de forma natural el conocimiento que tienen sobre el funcionamiento de una biblioteca real con el funcionamiento de la página Web.

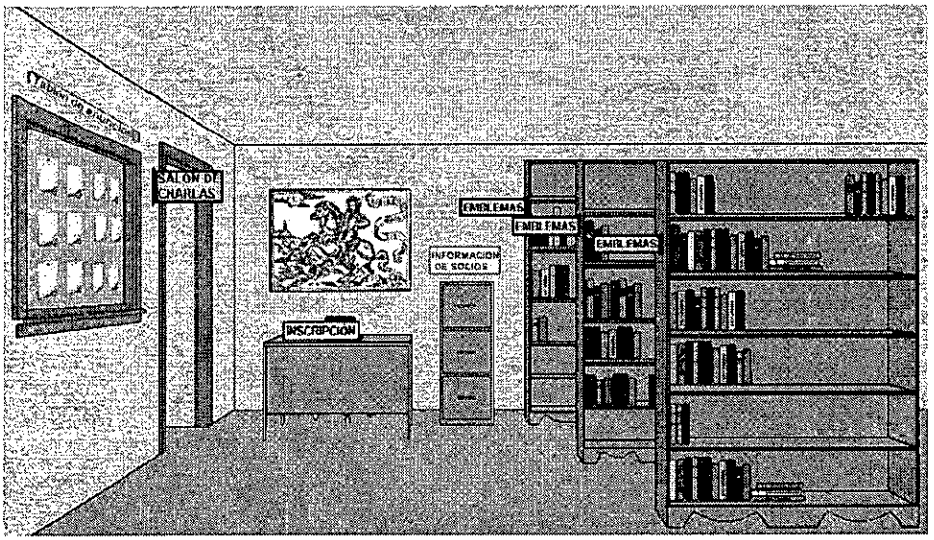


Fig. 7. Metáfora de Biblioteca

En la Fig. 8 y en la Fig. 9 se muestran dos metáforas que también han sido usadas en la Biblioteca Virtual de Literatura Emblemática. Se trata de un libro cerrado y un libro abierto, respectivamente. El usuario sabe que puede abrir el libro, en un caso, y pasar las páginas, en el otro caso, pinchando en las esquinas inferiores de los libros.

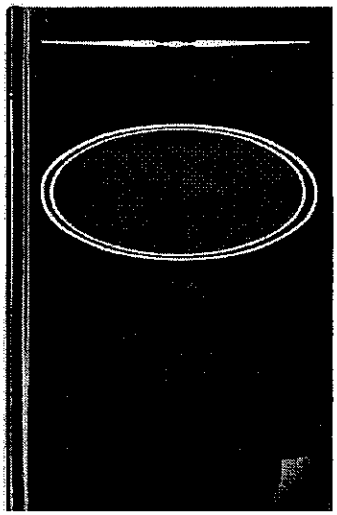


Fig. 8. Metáforas cognitivas: portada de libro

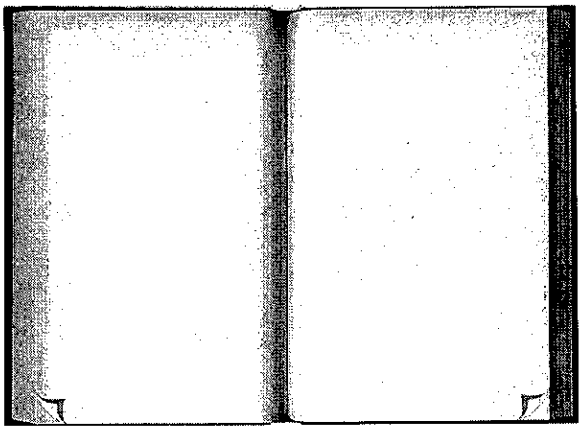


Fig. 9. Metáforas cognitivas: libro abierto

Del mismo modo, algunas otras metáforas que han sido diseñadas en el marco de la investigación sobre Interfaces de Usuario realizada en esta tesis, son las que se presentan en la Fig. 10. En esta figura se muestra un archivador y fichas para mostrar listados de elementos ordenados alfabéticamente. Estas dos metáforas están siendo usadas con mucho éxito en la Biblioteca Virtual Galega [13] y han sido objeto de varias publicaciones [73, 79, 82].

Todas estas metáforas pueden ser usadas tanto durante la consulta como en la fase de presentación de resultados de cualquier Interfaz de Usuario a bases de datos.

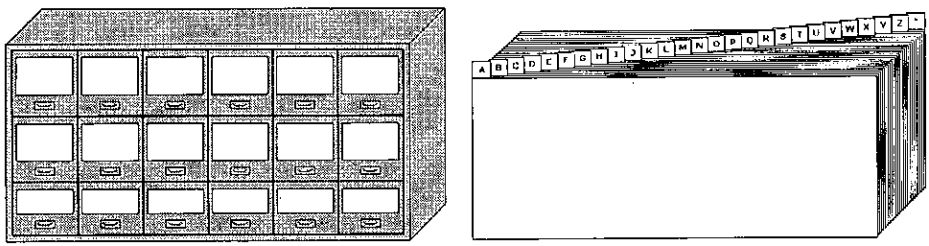


Fig. 10. Metáforas cognitivas: archivo y fichas

3.4 Frases en Lenguaje Natural Acotado (LNA)

Las Interfaces Web se construyen usando desde simples cuadros de texto hasta técnicas mucho más complejas y potentes. Algunos buscadores Web ofrecen la posibilidad de introducir una pregunta que es contestada por el sistema mostrando una lista de páginas Web que, supuestamente, pueden

ayudar al usuario a encontrar la respuesta a su pregunta. Es decir, la interfaz está basada en el uso de lenguaje natural para definir la consulta.

El uso de lenguaje natural es, indudablemente, muy potente. Sin embargo, tiene también muchos inconvenientes, debidos a la propia naturaleza del lenguaje natural. Los lenguajes naturales no siguen una gramática estricta (como lo hacen los lenguajes de programación), tienen sinónimos, frases hechas, y, además, las frases pueden ser ambiguas. Por lo tanto, es muy difícil encontrar una técnica que transforme correctamente una frase en lenguaje natural a un lenguaje formal en el que se pueda interrogar la base de datos en la que está almacenada la información. Parece claro, por tanto, que el uso de lenguaje natural puro para construir interfaces Web presenta muchos problemas para gestionar las consultas.

Nuestra propuesta es usar una técnica diferente que llamamos Lenguaje Natural Acotado (LNA). Esta técnica pretende aprovechar la expresividad y la facilidad de uso para los usuarios del lenguaje natural, evitando los inconvenientes que su decodificación lleva implícitos.

Una frase en Lenguaje Natural Acotado (LNA) es una frase en lenguaje natural con huecos que el usuario debe rellenar para expresar las condiciones de búsqueda. La técnica consiste en presentarle al usuario varias frases para poner restricciones. El usuario debe elegir qué frases va a usar y llenar los huecos. Finalmente, el conjunto de frases seleccionadas y completadas expresará, en lenguaje natural, la consulta completa que el usuario está haciendo.

En la Fig. 11 se muestra un conjunto de frases en Lenguaje Natural Acotado. La idea es que el usuario elija qué frase (o frases) se adecua(n) más para expresar la consulta que desea y rellene los huecos con los valores concretos de su consulta. Finalmente, las frases completadas (Fig. 12) expresan la consulta del usuario.

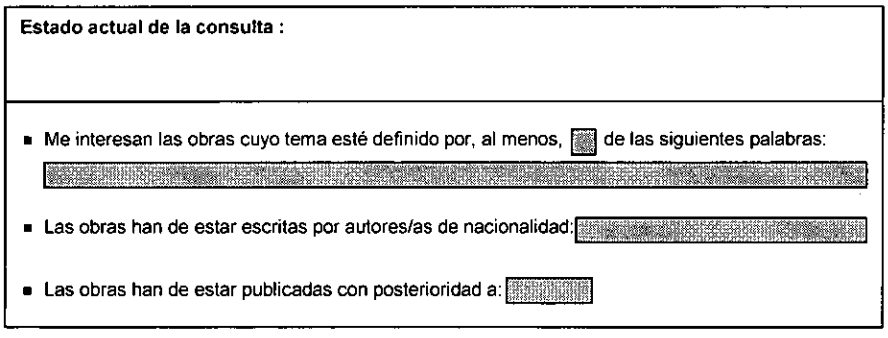


Fig. 11. Frases en LNA antes de rellenar los huecos

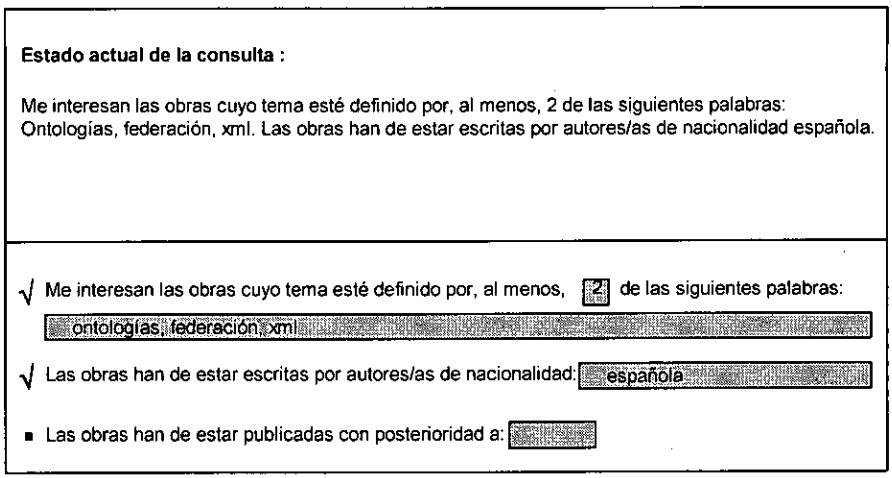


Fig. 12. Frases en LNA después de rellenar los huecos

En la frase en LNA del ejemplo (Fig. 11) el tipo de huecos que el usuario ha de rellenar son cuadros de texto. Como veremos en sucesivos ejemplos, hay diferentes tipos huecos que se pueden usar, proporcionando así una mayor facilidad para completar las frases. Por ejemplo: cuadros combinados (combo boxes), cuadros de lista (list boxes), botones de opción (radio buttons) o casillas de verificación (check boxes).

3.4.1 Versatilidad de la técnica del Lenguaje Natural Acotado (LNA)

Esta es la técnica de diseño de Interfaces de Usuario más usada en nuestro Sistema de Acceso Integrado a bases de datos. Esto es debido a que, a su gran flexibilidad para construir Interfaces de Usuario intuitivas (recordemos, que al fin y al cabo es lenguaje natural), se le une su gran versatilidad, es decir, puede usarse para construir Interfaces de Usuario que permitan expresar restricciones sobre cualquier atributo independientemente del tipo de dato que sea y de la condición que queramos permitir expresar.

Con esta técnica, no sólo se pueden construir frases en LNA para expresar restricciones sobre datos estructurados (fechas, números, etc.), sino también sobre datos no estructurados (contenido de los documentos) [71], permitiendo así usar este tipo de consultas para hacer Recuperación de Textos (Text Retrieval). Por ejemplo, se puede construir una frase en LNA que permita que el usuario escriba el conjunto de palabras que describen el texto que los documentos deben incluir para ser recuperados.

A continuación, se describen ejemplos de frases en Lenguaje Natural Acotado que permiten mostrar la gran versatilidad que presentan para ser usadas como medio de expresión de restricciones sobre cualquier tipo de

atributo. Presentamos una propuesta que incluye frases en LNA para permitir expresar restricciones sobre atributos de los tipos de datos principales. Se presenta una frase en Lenguaje Natural Acotado para atributos de tipo cadena de caracteres monovaluadas; una frase para cadenas de caracteres multivaluadas y otra para cadenas de caracteres largas. También presentamos nuestra propuesta de frases en LNA para atributos de tipo fecha y numérico. Por último, se presenta nuestra propuesta de frase en LNA para atributos de tipo texto, en donde se demuestra cómo con la técnica del Lenguaje Natural Acotado permite también expresar consultas por contenido.

Hay que destacar que esta es solamente una propuesta y que las frases en Lenguaje Natural se pueden construir totalmente adaptadas a las condiciones que se quieran expresar sobre cada atributo.

Atributos de tipo cadenas de caracteres

La frase en LNA que proponemos para permitir expresar restricciones sobre atributos de tipo cadena de caracteres se presenta en la Fig. 13. Esta frase en LNA sirve para expresar restricciones sobre el atributo "Nombre del autor".

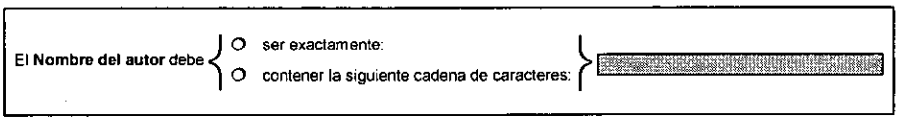


Fig. 13. Frase en LNA para cadenas de caracteres cortas

Aunque para cadenas de caracteres cortas, como lo es el "Nombre del autor", la frase en LNA adecuada podría ser la que hemos presentado en la Fig. 13, para cadenas de caracteres largas, como lo es el "Título de una Relación de Sucesos", que puede llegar a tener más de 2000 caracteres, sería más apropiada una frase en LNA como la que se muestra en la Fig. 14.

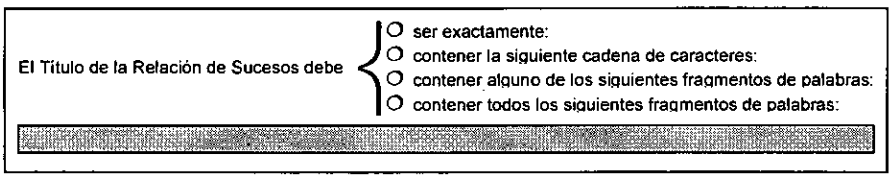


Fig. 14. Frase en LNA para cadenas de caracteres largas

Por otro lado, la frase en LNA de la Fig. 15, permite expresar restricciones sobre "Palabras Clave", que se trata de un atributo cadena de caracteres que es además multivaluado.

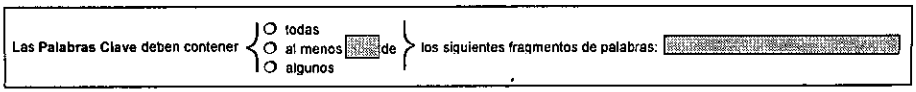


Fig. 15. Frase en LNA para atributos multivaluados de tipo cadena de caracteres

Hay que destacar que usamos "fragmento de palabras" para evitar el uso de comodines (como *, % o ¿) y facilitar al usuario la posibilidad de buscar palabras de la misma familia léxica. El significado de "fragmentos de palabras" y "cadena de caracteres" tendrá que ser explicado al usuario en la interfaz.

Atributos de tipo fecha

Para atributos de tipo fecha, como "Fecha de edición", podría ser útil la frase en LNA de la Fig. 16.

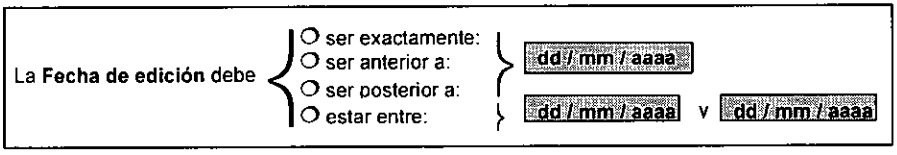


Fig. 16. Frase en LNA para un atributo de tipo Fecha

Atributos de tipo numérico

Asimismo, la frase en LNA útil para expresar condiciones sobre atributos de tipo numérico, como "Número de páginas" podría ser la que se presenta en la Fig. 17.

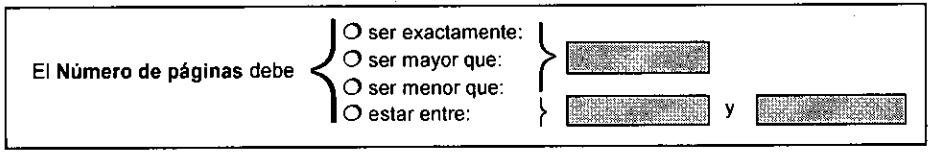


Fig. 17. Frase en LNA para un atributo de tipo numérico

Atributos de tipo texto y uso de Recuperación de Textos

Los atributos de tipo texto, como la "Glosa" de los Libros de emblemas, se pueden restringir con la frase en Lenguaje Natural Acotado de la Fig. 18.

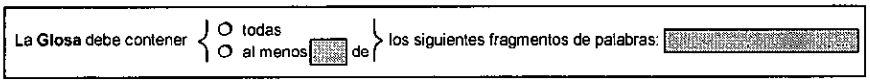


Fig. 18. Frase en LNA para el tipo de dato Texto

Las condiciones expresadas con esta frase pueden aplicarse aunque la base de datos no tenga disponible ninguna técnica de Recuperación de Textos específica (usando la sentencia *LIKE* de SQL). Sin embargo, con la técnica del Lenguaje Natural Acotado se puede sacar partido de cualquier técnica de recuperación de textos que esté implementada. A continuación, se muestran frases en LNA que permiten expresar consultas que necesiten alguna técnica de recuperación de textos para ser ejecutadas.

Por ejemplo, si la técnica de recuperación de textos implementada puede usar un tesauro de sinónimos y de palabras relacionadas, el sistema presentará al usuario la frase de la Fig. 19 a continuación de la frase de la Fig. 18. Rellenando los huecos de la frase de la Fig. 19, el usuario habilitará o deshabilitara el uso de dichos tesauros.

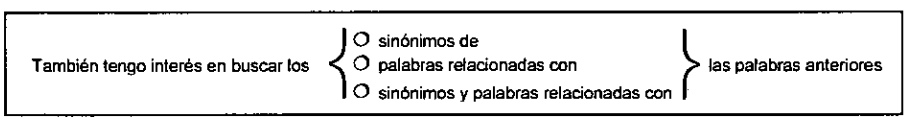


Fig. 19. Frase en LNA para el uso de tesauros

Otro ejemplo que muestra cómo la técnica del Lenguaje Natural Acotado puede ayudar al usuario a sacar partido de las diferentes técnicas de recuperación de textos que estén disponibles es la frase de la Fig. 20. Esta frase (presentada al usuario a continuación de la frase de la Fig. 18) permitirá realizar búsquedas aproximadas. Es decir, permitirá que se recuperaren aquellos documentos que contengan palabras que difieran en uno o dos caracteres de las escritas por el usuario en la frase de la Fig. 18.

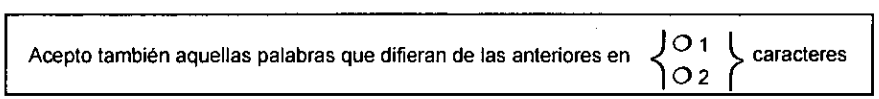


Fig. 20. Frase en LNA para búsqueda aproximada

Si la técnica de recuperación de textos implementada permite establecer diferentes pesos para las palabras de la consulta, la frase en LNA adecuada para guiar al usuario en el proceso de dar diferentes pesos a las palabras podría ser la que aparece en la Fig. 21 (el sistema presentará esta frase en vez de la frase de la Fig. 18).

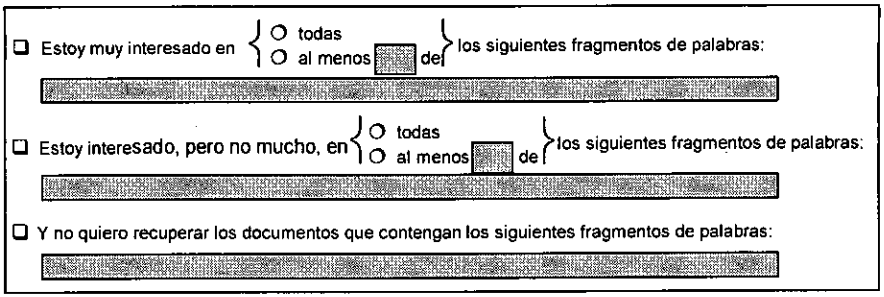


Fig. 21. Frase en LNA para otras técnicas de recuperación de textos

En resumen, la idea principal que debe extraerse de este y anteriores subapartados es que sea cual sea el tipo de dato de un atributo y sea cual sea la condición que queramos permitir expresar al usuario, podremos construir una frase en Lenguaje Natural Acotado que permita expresarlas de una forma intuitiva para el usuario (en lenguaje natural) y sin obligarlo a usar caracteres extraños con un significado especial en la Interfaz de Consulta.

3.4.2 Implementación y Esqueletos de frases

Las frases en Lenguaje Natural Acotado que hemos presentado en el apartado anterior para atributos de diferentes tipos de datos sirven, con sencillas modificaciones, para expresar restricciones no sobre los atributos que aparecen en los ejemplos, sino sobre cualquier otro atributo del mismo tipo de dato.

En este punto podemos introducir lo que hemos denominado Esqueleto de Frase. Un Esqueleto de Frase es una plantilla que aplicada a un atributo produce la frase en Lenguaje Natural Acotado útil para expresar restricciones sobre dicho atributo. Como ya hemos avanzado, un mismo Esqueleto de Frase, salvo pequeñas modificaciones, servirá para expresar restricciones sobre distintos atributos, si dichos atributos son del tipo para el que el Esqueleto de Frase está pensado.

Por tanto, para implementar la técnica del Lenguaje Natural Acotado en nuestro Sistema de Acceso Integrado hemos creado, como se verá más adelante, Esqueletos de Frase para cada grupo de atributos del mismo tipo de dato [72, 72].

Esqueletos de Frase para atributos de tipo cadenas de caracteres

Por lo tanto, para los atributos de tipo cadenas de caracteres cortas, utilizamos el Esqueleto de Frase de la Fig. 22. De esta manera, en el momento en el que el sistema tenga que presentar al usuario la frase en Lenguaje Natural Acotado

para el atributo "Nombre de autor", el sistema tomará el Esqueleto de Frase de la Fig. 22 y reemplazará la etiqueta <ATRIBUTO> por "Nombre de autor", y así generará la frase en LNA que presentamos en la Fig. 13 y que será la que verá el usuario.

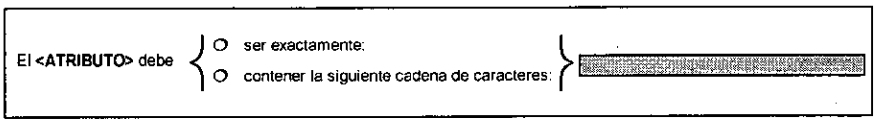


Fig. 22. Esqueleto de Frase para cadenas de caracteres cortas

El Esqueleto de Frase de la Fig. 22 es útil también para atributos como "Ciudad", "Nacionalidad" o "Dirección", que son también cadenas de caracteres cortas.

Para los atributos de tipo cadena de caracteres de mayor longitud, como por ejemplo "Título del Libro de Emblemas", es más apropiado el Esqueleto de Frase que se muestra en la Fig. 23.

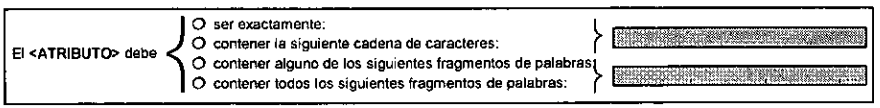


Fig. 23. Esqueleto de Frase para cadenas de caracteres largas

El Esqueleto de Frase de la Fig. 24, es útil para atributos multivaluados de tipo cadena de caracteres. Este esqueleto permite generar la frase en LNA útil para expresar restricciones sobre atributos como "Palabras clave" (ver Fig. 15).

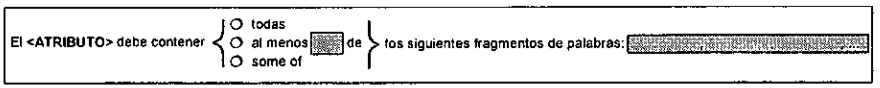


Fig. 24. Esqueleto de Frase cadenas de caracteres multivaluadas

Esqueletos de Frase para atributos de tipo fecha

De la misma forma, para los atributos de tipo fecha, podemos crear un Esqueleto de Frase que una vez instanciado con cada atributo fecha concreto ("Fecha de edición", "Fecha de publicación", "Fecha de nacimiento", etc.), dé lugar a las Frases en Lenguaje Natural Acotado útiles para que el usuario pueda expresar condiciones sobre dichos atributos.

El Esqueleto de Frase útil para expresar restricciones sobre atributos de tipo fecha podría ser el mostrado en la Fig. 25.

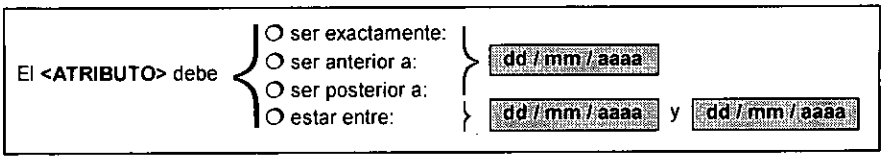


Fig. 25. Esqueleto de Frase para fechas

Esqueletos de Frase para atributos de tipo numérico

Asimismo, el Esqueleto de Frase útil para expresar condiciones sobre atributos de tipo numérico podría ser el presentado en la Fig. 26. El Esqueleto de Frase mostrado en la Fig. 25 será útil para atributos como "Número de páginas", "Año", etc.

Fig. 26. Esqueleto de Frase para numéricos

Esqueletos de Frase para atributos de tipo texto

El Esqueleto de Frase de la Fig. 27 aplicado sobre atributos de tipo texto como "Glosa", "Epigrama", etc. da lugar a las frases en Lenguaje Natural Acotado útiles para expresar restricciones sobre dichos atributos.

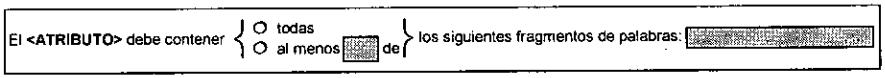


Fig. 27. Esqueleto de Frase para textos

En resumen, en nuestro Sistema de Acceso Integrado hemos implementado la técnica del Lenguaje Natural Acotado sin necesidad de crear una frase en Lenguaje Natural Acotado para cada atributo sobre el que queremos permitir al usuario expresar restricciones. Basta con crear una Esqueleto de Frase para todos los atributos de un mismo tipo de dato.

3.5 Aproximación Navegacional

Esta técnica está íntimamente ligada a las interfaces de usuario desarrolladas para los buscadores de páginas en Internet. En este aspecto debemos indicar

que existen amplios estudios en el campo de recuperación de información [9, 30] y también en el diseño de sistemas de búsqueda de información documental y Bibliotecas Virtuales que faciliten el acceso a grandes colecciones de documentos.

Existen estudios comparativos que establecen que los usuarios Web, en función de la temática, prefieren en la mayoría de las ocasiones hacer búsquedas sencillas, para navegar después sobre los resultados obtenidos, en lugar de realizar consultas complejas que les permitan afinar más sus restricciones de búsqueda. En [43] se presentan estudios realizados sobre buscadores populares, como Altavista [5], que indican que más del 72% de las consultas estaban formadas como máximo por 2 palabras, y que cerca del 80% de las consultas no incluían operadores, ni siquiera los booleanos `and` u `or`, o los signos + (que exigen que la palabra a la que precede esté en el documento) o – (que fuerza que la palabra no aparezca).

La Aproximación Navegacional es una técnica de diseño de Interfaces de Usuario que puede usarse tanto en Interfaces de Consulta como interfaces de presentación de respuestas.

Por ejemplo, en consulta, si un atributo sobre el que el usuario puede querer expresar restricciones tiene pocos valores posibles, es mejor usar la aproximación de presentar todos las alternativas en pantalla y dejarle elegir, que pedirle que restrinja la búsqueda por ese atributo en un formulario típico, escribiendo el valor que el atributo debe tener.

En la Fig. 28, se muestra un ejemplo de uso de la Aproximación Navegacional para una Interfaz de Consulta. Como se puede ver, para restringir el "Género Literario" de las obras buscadas sólo es necesario pinchar sobre el cajón que corresponda.

Esta Interfaz de Consulta usa la Metáfora Cognitiva del archivador combinada con la técnica de la Aproximación Navegacional y ha sido creada para la Biblioteca Virtual Galega [13].

Por otro lado, para la presentación de respuestas es mejor presentar los resultados bien clasificados por algún atributo de interés que obligar al usuario a restringir la búsqueda por dichos atributos.

En la Fig. 29 se muestra un ejemplo de uso de la Aproximación Navegacional en respuesta. El resultado de pinchar sobre el cajón del archivador correspondiente a "Títulos de Poesía" es un listado de dichas obras clasificadas por autor y época. Esta es una de las Interfaces de Respuesta que ha sido creada para la Biblioteca Virtual Galega [13].

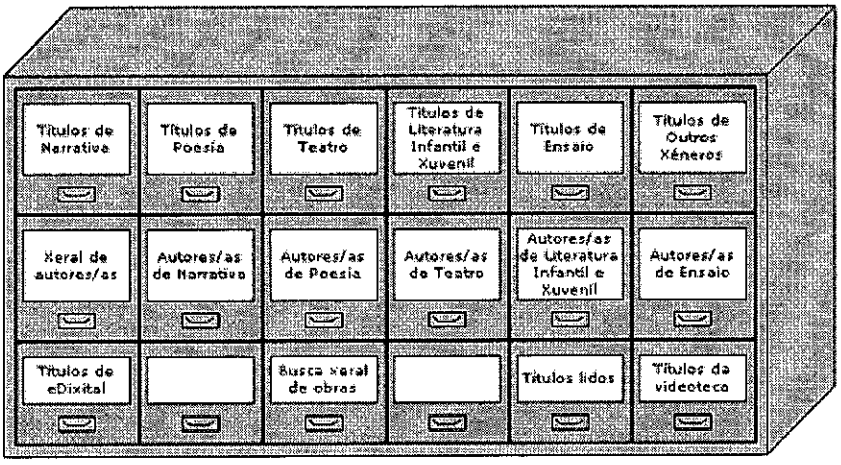


Fig. 28. Aproximación Navegacional en consulta

Listaxe de obras de poesía

Época	Autor(a)/Obra	Dispoñíbel en: Texto, Audio, Vídeo, Imaxe
Medieval	Joan de Cangas:	
	· Cantigas de amigo	
Medieval	Martin Codax:	
	· Cantigas de amigo	
Medieval	Meendinho:	
	· Cantiga de amigo	
Medieval	Trobadores:	
	· Cantigas de amigo	
	· Cantigas de amor	
	· Cantigas de escarnho e maldizer	
S. Escuros	Cornide, Xosé:	
	· Composicións galegas	
S. Escuros	Fruíme, Cura de:	
	· Composicións galegas	
S. Escuros	Padre Sarmiento:	
	· Coloquio de 24 gallegos rústicos	
S. XIX	Añón, Francisco:	
	· Poesías varias	

Fig. 29. Aproximación Navegacional en respuesta

La técnica de la Aproximación Navegacional es la técnica subyacente que hace tan útiles y atractivas las interfaces de los Sistemas de Información Geográfica (SIG). Evidentemente, un mapa sobre una pantalla es una Metáfora Cognitiva extremadamente útil. En vez de usar un formulario en donde el usuario pueda restringir el atributo "País" para buscar información sobre un país determinado, se le presenta un mapa y el usuario, al pinchar sobre el área de cierto país, encuentra de forma natural la información sobre dicho país.

Dado que en Biblioteca Virtual Gallega queríamos presentar información sobre diferentes elementos de interés para el turismo cultural de la región, decidimos desarrollar un SIG que permitiese usar y explotar las ventajas de la Aproximación Navegacional.

En la Fig. 30, se presenta la Interfaz de Consulta del Viaje Virtual por Galicia [80], que, cómo hemos dicho, es un SIG incluido en la Biblioteca Virtual Galega [13].

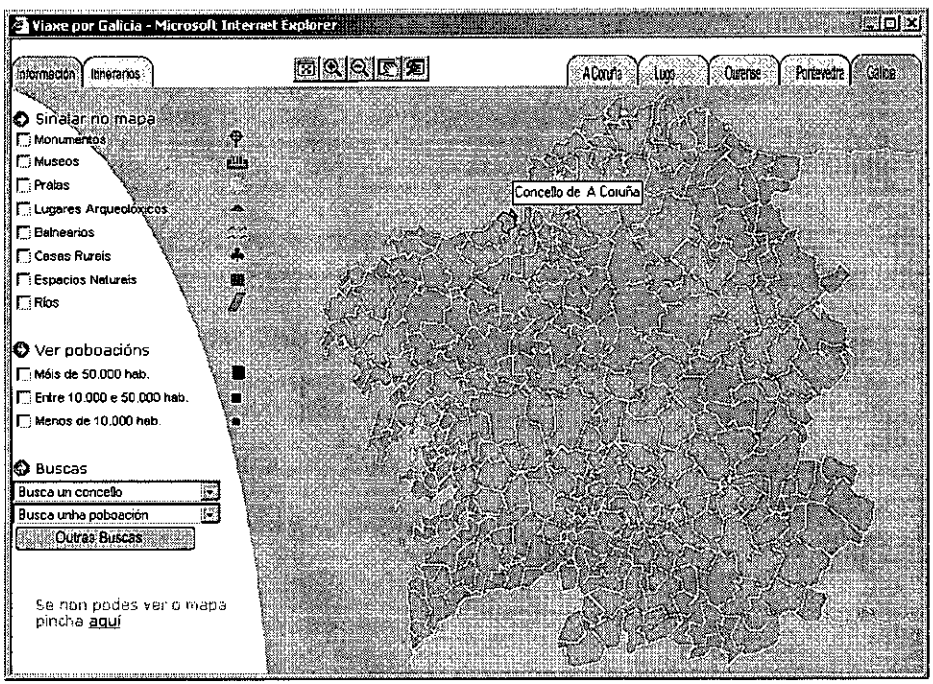


Fig. 30. Aproximación Navegacional en SIG

3.6 Validación de las técnicas de Diseño de Interfaces propuestas

Es bien sabido que en el campo de Human Computer Interaction (HCI) es preciso realizar una validación experimental de cualquier técnica que se proponga para garantizar que, efectivamente, mejora la facilidad del uso de la interfaz, con respecto a la interfaz que se construiría sin aplicar dicha técnica.

Sin embargo, en el marco de esta tesis, no se ha realizado tal validación experimental sistemática de las ideas que se han expuesto en esta sección, principalmente debido a que dicha investigación se apartaba del objetivo principal de la tesis, y a que, por otro lado, tenemos, como se verá a

continuación, numerosos datos empíricos (aunque no de procedencia experimental) que avalan la utilidad de las técnicas que se han descrito.

Los datos empíricos, a los que se ha he hecho referencia en el párrafo anterior, provienen, en unos casos, de informaciones directas extraídas de los usuarios de las interfaces que se construyeron, y que probaron a acceder a cada una de las Bibliotecas Digitales por separado cuando estas se crearon. Por otro lado, provienen de información sobre el impacto en el número de accesos recibidos que tuvo el cambio de interfaz.

Además, usamos también otros datos que proceden indirectamente del número de usuarios que han utilizado las Bibliotecas Digitales para las que se construyeron las diferentes interfaces.

Para exponer adecuadamente este conjunto complejo de datos, es necesario relatar el proceso de construcción de las mismas.

3.6.1 Descripción del primer prototipo

El primer prototipo fue creado por integrantes del Laboratorio de Bases de Datos, antes de que se iniciase esta tesis, para la base de datos de Libros de Emblemas. Este prototipo seguía una aproximación QBE básica, cuya idea principal es realizar una consulta a una base de datos mediante una posible respuesta a esa consulta sin tener que especificar un procedimiento para obtener la respuesta.

La interfaz de consulta se basaba en formularios compuestos por una serie de cuadros de texto en los que el usuario introducía las condiciones de búsqueda. Para realizar la consulta, el usuario debía rellenar sólo aquellos campos de búsqueda que deseaba restringir.

Se diseñaron varios formularios de consulta que agrupaban conjuntos de atributos relacionados temáticamente, cada uno de ellos adecuado para consultar la base de datos desde un punto de vista o interés diferente.

El primer prototipo de interfaz se muestra en la Fig. 31.

Este prototipo no fue aceptado ni usado, ni siquiera por el grupo de investigadores del Equipo de Humanidades con el que trabajábamos. Los problemas detectados se pueden resumir en dos:

- La posibilidad de usar conectores lógicos en un campo no quedaba indicado en ningún lado.
- La interfaz daba lugar a múltiples errores de escritura debido a la necesidad de teclear todos los valores.

Buscando resolver los problemas detectados en el prototipo anterior, se desarrollo una nueva versión de la interfaz que denominamos segundo prototipo y que describimos a continuación.

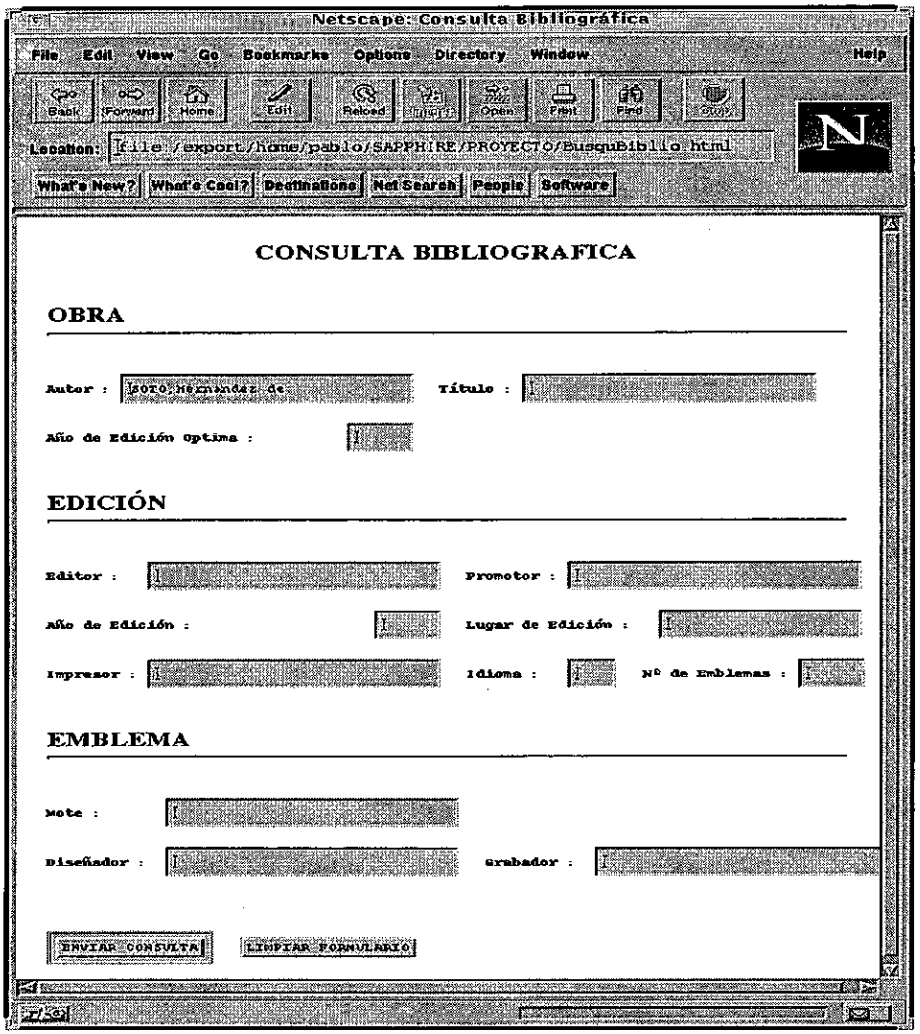


Fig. 31. Primer prototipo

3.6.2 Descripción del segundo prototipo

Esta interfaz presentaba la misma estructura que la anterior, pero se le añadieron una serie de mejoras que afectaban a cada atributo de búsqueda.

- *Uso de listas desplegables* (combo boxes): Cuando el número de valores distintos de un atributo era reducido, se presentaban dichos valores en una lista desplegable (cuadro combinado), para evitar la necesidad de teclear, evitando así errores.
- *Conectivas lógicas*: Al lado de cada atributo se añadían dos botones con las etiquetas "y" (and) y "o" (or), para indicar qué conector era posible usar para relacionar entre sí los valores que se daban a ese atributo.
- *Establecimiento de rangos*: Para establecer una restricción en forma de rango, se situó al lado del atributo un botón con la etiqueta "de..a". De esta forma, si el usuario quería establecer un rango pulsaba el botón, con lo que automáticamente se escribía "de" en el cuadro de texto. Después de escribir el límite inferior del rango, volvía a pulsar de nuevo el botón para introducir el límite superior. Esto era muy útil para atributos de tipo fecha y numéricos, y, sin embargo, fue poco usado.

El aspecto que presentaban los nuevos formularios de consulta era el de la Fig. 32.

Antes de que este segundo prototipo se llegase a desarrollar por completo fue presentado a nuestros expertos en Filología, quienes, a pesar de que esta nueva interfaz resultaba más amigable y fácil de utilizar que la anterior, rechazaron la propuesta porque todavía presentaba ciertos problemas:

1. Problemas derivados del uso de conectores lógicos:

- Los conectores and y or no les resultaban sencillos de usar, ya que son contraintuitivos y confusos para usuarios no informáticos. Para buscar obras de dos autores diferentes, se planteaban la consulta como "Obras de A y obras de B", con lo que en la interfaz también usaban el conector lógico "y", lo que provocaba que sólo apareciesen las obras escritas por los dos autores (es decir, ninguna en Literatura Emblemática).
- Aun en el caso de usar correctamente los conectores lógicos, seguían teniendo problemas cuando los usaban para unir palabras que debían aparecer en el texto de los emblemas para que fuesen recuperados.

  El problema es que usando el conector or entre las palabras de búsqueda se restringe muy poco la consulta. Para que un emblema satisfaga la consulta basta con que contenga una de las palabras especificadas, por lo que la respuesta está formada por demasiados emblemas que, realmente, se adecuan muy poco a la consulta. Por el contrario, el conector and la restringe demasiado, suelen existir muy pocos emblemas, o ninguno, que contengan todas las palabras especificadas.

Por ejemplo, si se está interesado en buscar emblemas que versen sobre la corrupción de la iglesia y su abuso de poder durante la inquisición, podrían usarse como apalabras de búsqueda las siguientes: "juicio, inquisición, tribunal, sentencia, hoguera, converso, corrupción, clérigo". Estas palabras conectadas por `or`, recuperarían un sin fin de emblemas y conectadas por `and` probablemente no recuperarían ninguno.

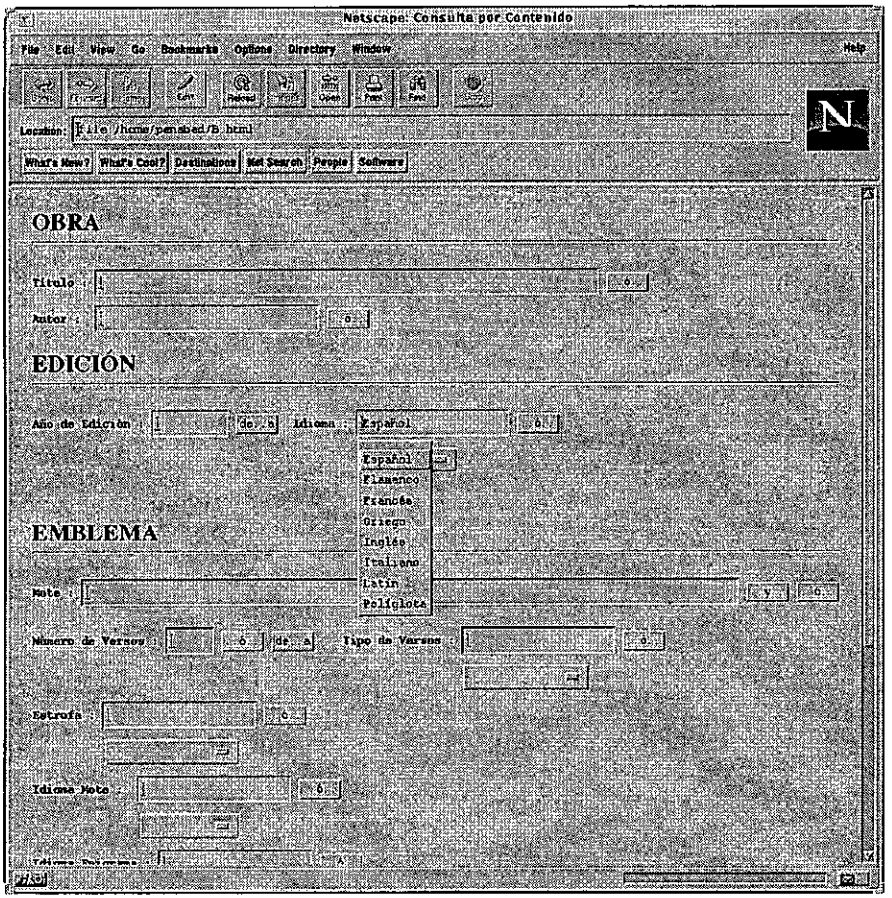


Fig. 32. Segundo prototipo

2. Problemas derivados de la aproximación booleana:

- Dada una consulta, un emblema se recupera o no, pero no se le puede asignar un valor que indique su grado de adecuación a la consulta, por lo que si se obtenían muchos resultados, el usuario tardaba mucho tiempo en buscar cuáles eran los emblemas que realmente le interesaban.

No se utiliza la idea de relevancia de una palabra en un emblema. Una palabra aparece, o no, en un documento, pero no se puede valorar lo importante que es, es decir, no se considera si aparece muchas veces en el emblema o solamente una vez.

Para resolver los problemas observados en el segundo prototipo, se desarrolló un nuevo modelo en el que ya se usó la técnica de Lenguaje Natural Acotado, lo que constituyó la base de esta primera interfaz exitosa después de los dos fracasados prototipos anteriores. Además, dicha interfaz explotaba algunas técnicas de Recuperación de Textos ya conocidas que hubo que adaptar a las necesidades concretas de nuestra base de datos y usuarios, huyendo así de la rudimentaria aproximación booleana pura.

3.6.3 Descripción del tercer prototipo

Este fue el primer trabajo en el que participó esta doctoranda. En este prototipo utilizamos por primera vez la técnica del Lenguaje Natural Acotado (LNA). Construimos una interfaz completamente basada en esta aproximación, que fue presentada como demo en EDBT'98 [14].

La interfaz está basada en una serie de pantallas, cada una de las cuales tenía una serie de frases en Lenguaje Natural Acotado que permitía expresar restricciones sobre el mismo aspecto de la base de datos. El usuario debía seleccionar en qué frases estaba interesado para expresar su consulta y rellenar los huecos. Finalmente, el conjunto de frases seleccionadas y rellenadas expresaban la consulta del usuario. Una de estas pantallas es la que se presenta en la Fig. 33.

Este prototipo generó una mayor satisfacción en nuestros investigadores. Las frases en lenguaje natural facilitaban la definición de consultas. Así que esta interfaz se publicó en Web y fue la primera interfaz pública de nuestra Biblioteca Digital de Libros de Emblemas. Así, esta fue la interfaz que se presentó en congresos de Literatura Emblemática para darle difusión a la Biblioteca Digital.

El número de accesos recibidos en esta interfaz fue relativamente escaso a pesar de que los investigadores en Literatura Emblemática estaban muy interesados en los fondos de nuestra biblioteca. Sólo algunos estudiosos de la Sociedad Española de Emblemática [58] llegaron a sacarle partido a toda su potencialidad.

Aunque el significado de las frases les resultaba perfectamente claro, la lógica de la agrupación de las frases en pantallas, y la navegación entre dichas pantallas, para construir las consultas completas o para restringir todavía más los resultados obtenidos en consultas ya formuladas, requería cierto

entrenamiento por parte de los usuarios. No estaba clara la lógica de enlace entre dichas pantallas.

Por ello, se decidió crear una nueva interfaz, aprovechando lo aprendido en el tercer prototipo pero con dos nuevos objetivos: hacerla más amigable, y próxima al usuario, y facilitar su utilización y comprensión, simplificándola para usuarios poco exigentes, pero dotándola, además, de capacidades de realización de consultas más complejas para usuarios más exigentes y especializados. Es decir, queríamos hacer una interfaz muy fácil para usuarios que buscaran una cierta "Obra" o un cierto "Autor", pero, al mismo tiempo, queríamos permitir que usuarios interesados en hacer búsquedas complejas (Ej. "Emblemas con pájaros en la imagen y que citen a Aristóteles") también pudieran realizar sus consultas.

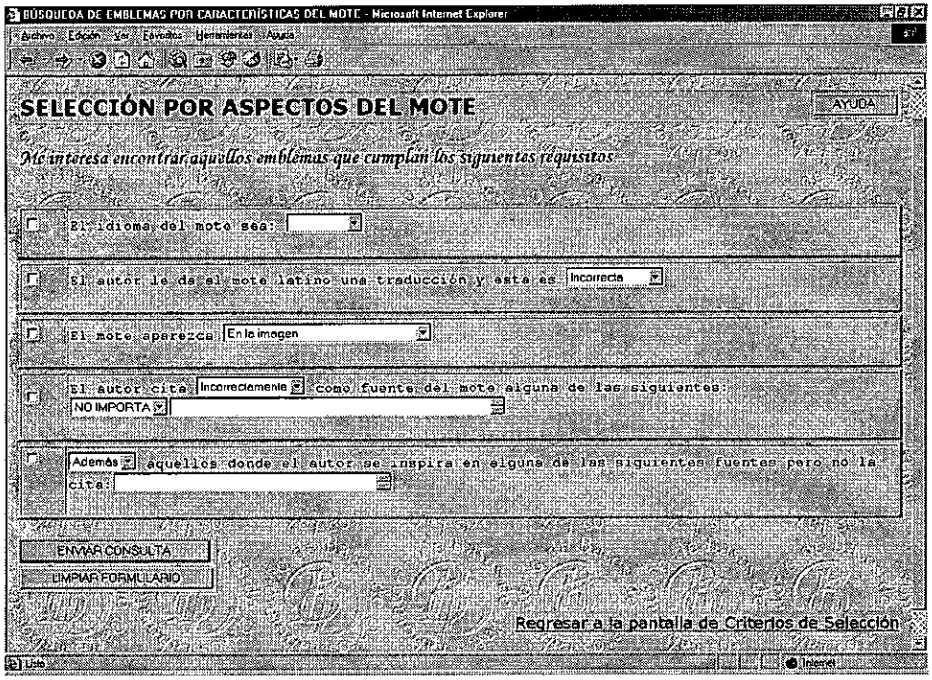


Fig. 33. Tercer prototipo

3.6.4 Descripción del cuarto prototipo

Este prototipo, la Biblioteca Virtual de Literatura Emblemática [11], fue realizado íntegramente por esta doctoranda y ha dado lugar a numerosas publicaciones [68, 69, 77, 83].

Al iniciar el diseño de este prototipo sabíamos que se necesitaba una interfaz muy fácil de usar y que permitiese expresar consultas muy simples. Sin embargo, no queríamos sacrificar por completo la capacidad de realizar consultas complejas, que de forma tan intuitiva permite expresar la técnica del Lenguaje Natural Acotado. Por ello, se siguió la estrategia de diseñar e implementar una Interfaz que se adaptase al nivel de experiencia de los usuarios, permitiendo dos niveles de complejidad en las consultas. De esta manera, se desarrollaron los dos tipos siguientes de interfaces:

- Interfaz para usuarios "comunes": Se aplicó la técnica de las Metáforas Cognitivas combinada con la Aproximación Navegacional. El resultado fue una interfaz intuitiva, fácil de usar y que permite realizar consultas simples.
- Interfaz para usuarios "expertos": Se desarrollo utilizando la técnica del Lenguaje Natural Acotado. A través de esta interfaz se puede sacar partido a la gran cantidad de información que está almacenada en la base de datos, ya que permite expresar consultas más complejas.

De aquí en adelante se describe esta interfaz y se verá claramente qué pantallas se crearon para permitir búsquedas muy simples, y cuales están diseñadas para permitir formular consultas complejas.

La página principal (Fig. 34) representa una biblioteca tanto en apariencia como en funcionamiento. La sección de búsquedas está accesible pinchando con el botón izquierdo del ratón en las estanterías que aparecen en la parte derecha de la página.

Pinchando en una de las estanterías el usuario puede acceder a la pantalla de la Fig. 35. En ella se presenta la "Portada" de un libro y se permite al usuario especificar el título o el autor del libro que desea consultar.

Además, (pinchando en el interrogante en la pantalla de la Fig. 35) se le da la opción de restringir los atributos más relevantes de los emblemas (mote, epigrama, imagen, palabras clave, etc.). La pantalla que se le presenta al usuario para que exprese este tipo de consultas está basada en una Metáfora Cognitiva y es muy fácil de usar como se puede ver en la Fig. 36. Se presenta con el aspecto de un "Libro vacío" en el que el usuario rellena las características que desea que restrinjan la búsqueda.

Desde esta pantalla todavía el usuario puede acceder a interfaces que le ofrecen la posibilidad de expresar consultas más complejas por el "contenido" o "tema" de los documentos.

Para diseñar este tipo de interfaces se ha usado, como ya hemos dicho, la técnica del Lenguaje Natural Acotado. En la Fig. 37, se muestra una de las pantallas en las que usamos esta técnica.

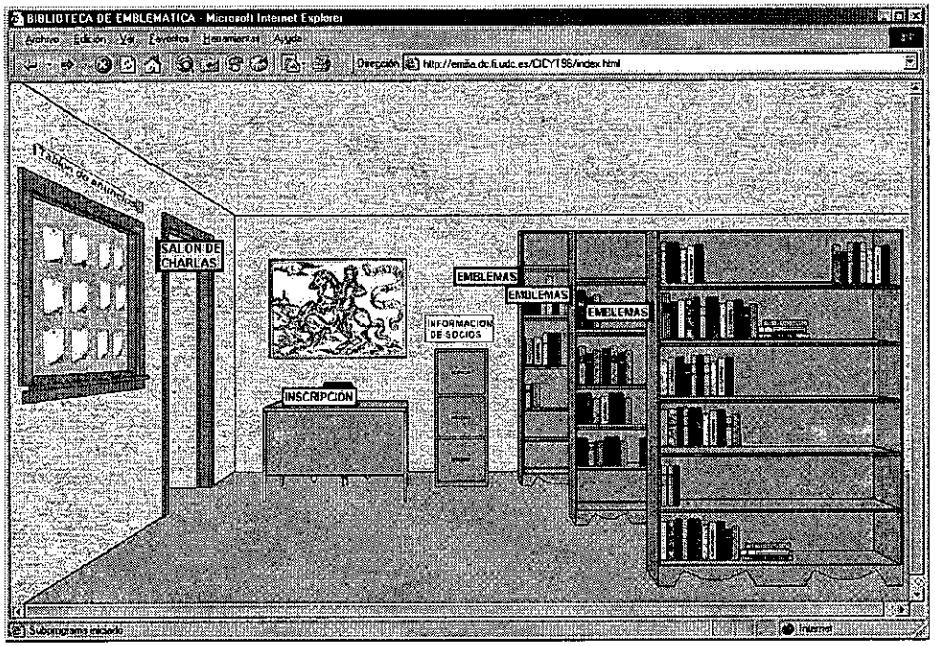


Fig. 34. Página principal de la Biblioteca Virtual de Emblemática

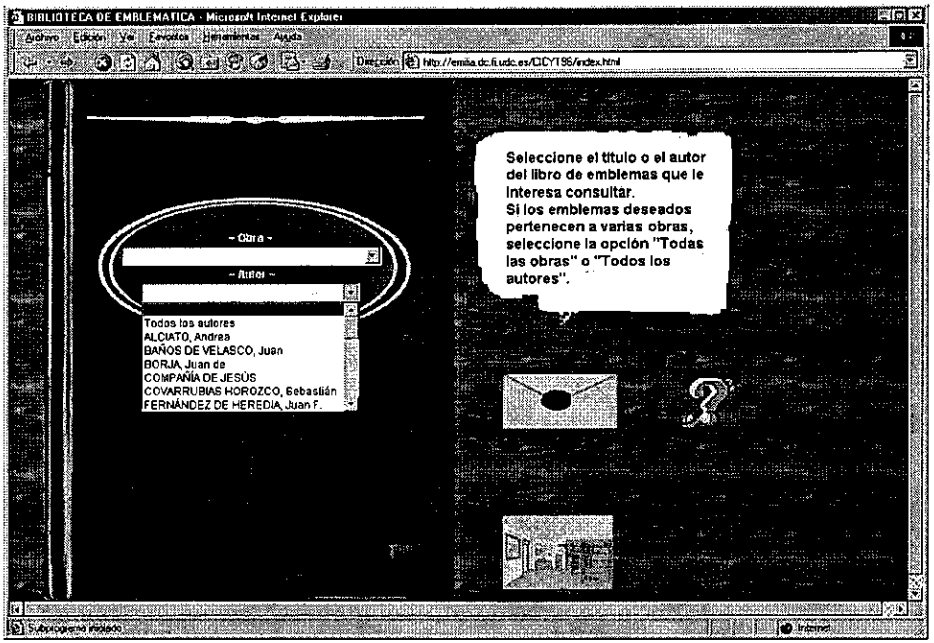


Fig. 35. Consulta por título de la obra y autor

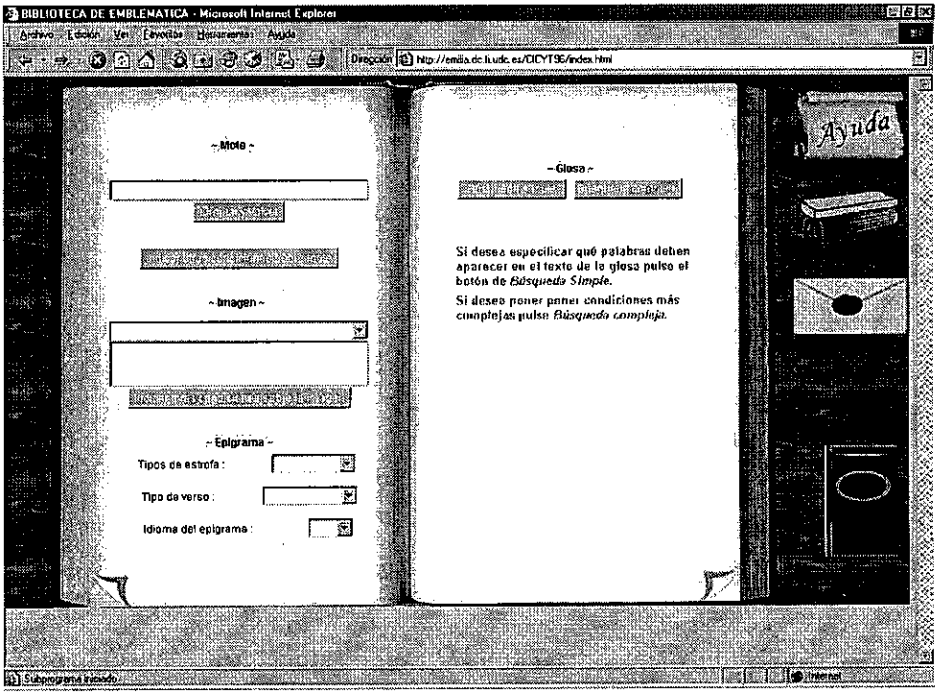


Fig. 36. Consulta por los principales datos de los emblemas

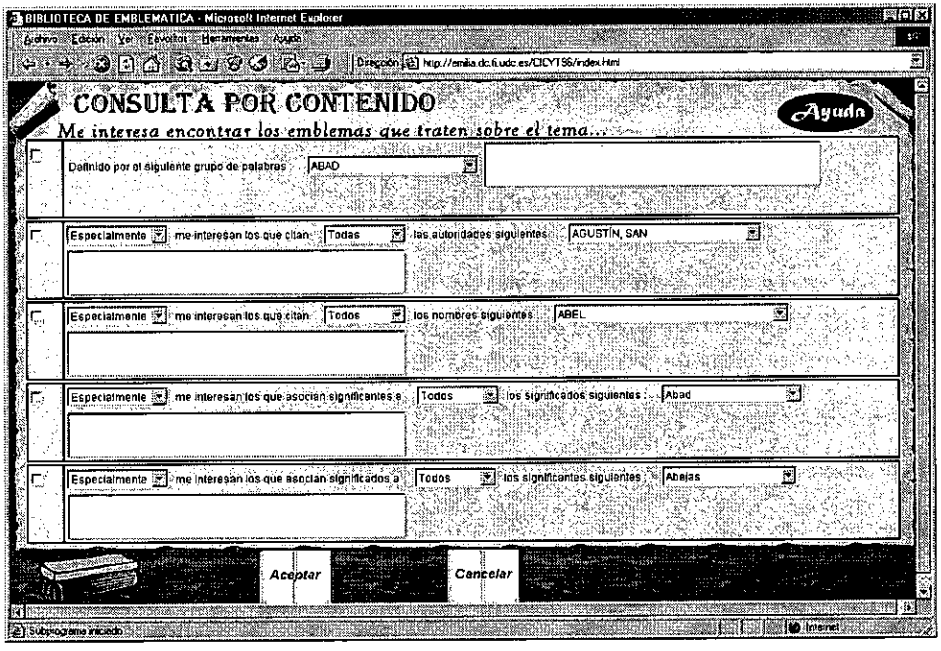


Fig. 37. Búsqueda por contenido

Para presentar al usuario los emblemas obtenidos en su consulta se usa una metáfora de un libro. De forma intuitiva, el usuario puede pasar las páginas del "Libro Virtual" para navegar por el conjunto de emblemas recuperados. Tiene la posibilidad de ver las páginas correspondientes a emblemas sobre un tema concreto o de un autor determinado, etc y, para cada emblema, visualizar la página original del libro digitalizada, si está disponible en la base de datos.

Una de las páginas de la Interfaz de Respuesta de la Biblioteca Virtual de Literatura Emblemática se muestra en la Fig. 38, que presenta las páginas virtuales asociadas a un emblema.

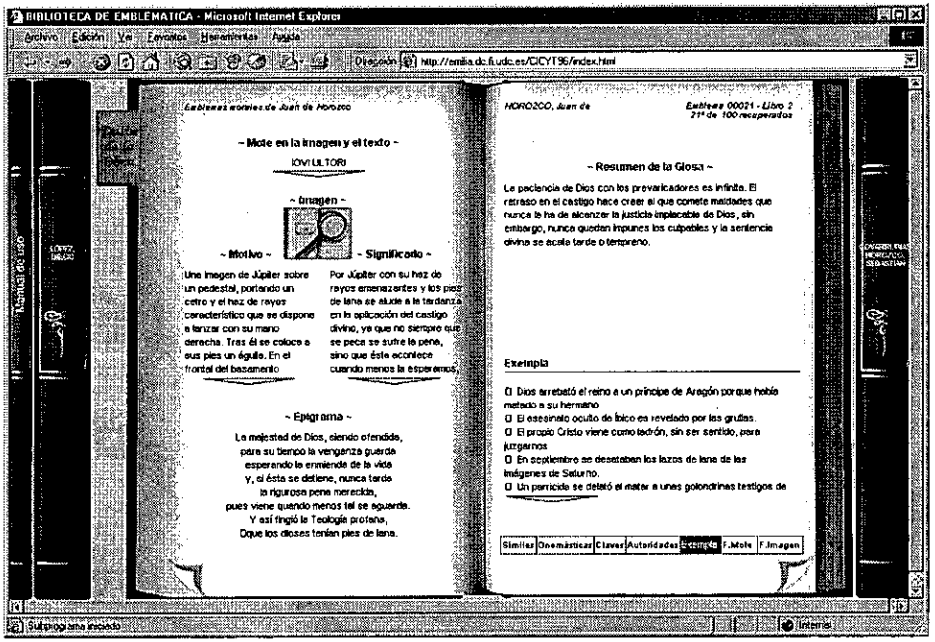


Fig. 38. Libro Virtual

Este último prototipo ha tenido mucho éxito tanto a nivel nacional como internacional. Cuando sustituimos la interfaz anterior, basada íntegramente en Lenguaje Natural Acotado, por esta en la que se combinan Metáforas Cognitivas, Aproximación Navegacional y Lenguaje Natural Acotado, no sólo nuestro grupo de filólogos se sintió más cómodo, sino que el número de accesos de investigadores empezó a subir hasta el punto de que en los tres años que han pasado desde su publicación hemos recibido más de 29.000 accesos. Esta cifra es muy alta considerando que el contexto en el que estamos es la Literatura Emblemática.

Aunque, si bien es cierto que cada día la Biblioteca Virtual de Literatura Emblemática es conocida por más investigadores (sorprendentemente

cualquier investigador en Literatura Emblemática conoce nuestra interfaz) lo que implica que el número de accesos sube progresivamente, sí hemos registrado un salto significativo en el número de accesos recibidos a partir del cambio de interfaz.

Actualmente, la Biblioteca de Literatura Emblemática es la primera entrada en Google [33], Yahoo [64] o Altavista [5] en el tema de Literatura Emblemática. En la Fig. 39 se muestran los resultados obtenidos en Google al buscar "Emblematic Literature".

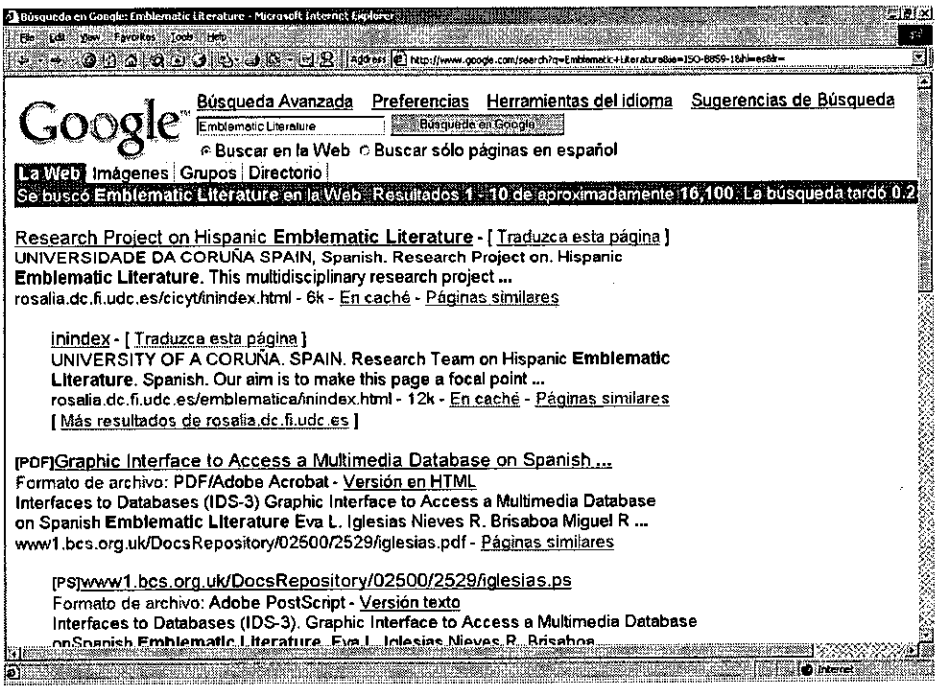


Fig. 39. Resultados en Google

Por otro lado, la actividad de los usuarios que acceden a la Biblioteca de Literatura Emblemática tiende a desenvolverse en las pantallas más sencillas, mientras que sólo en unos pocos casos se accede a las pantallas con frases en Lenguaje Natural Acotado para expresar consultas más complejas. De hecho, cerca del 90% de las consultas formuladas son expresadas a través de la Metáfora Cognitiva en la que aparece la Portada del Libro; cerca de un 10% se formulan usando la Metáfora del Libro Abierto; y poco más del 1% de las consultas son formuladas a través de las frases en Lenguaje Natural Acotado.

A la vista de estos datos, nos parece imprescindible el contar con diferentes secciones a través de las que se puedan expresar consultas de diferente nivel de complejidad.

3.6.5 Interfaz para la Biblioteca Virtual Gallega

La Biblioteca Virtual Gallega (BVG) [13] es una Biblioteca Digital de Literatura Gallega de todos los tiempos que ofrece versiones, texto, audio y vídeo de un buen número de obras literarias que cubren todos los géneros, desde la Literatura Infantil hasta el Teatro.

La biblioteca se ofrece como medio de comunicación entre los escritores y escritoras actuales y el público lector, ya que cada autor actual tiene una página Web en la que puede publicar noticias, opiniones, etc., además de actualizar su autobiografía o su catálogo de obras. A través de esta página Web, los autores pueden comunicarse abiertamente con su público, anunciándole conferencias, publicaciones, o simplemente manifestándole sus opiniones. Por otro lado, también a través de esta página, los lectores pueden enviar a los diferentes autores y autoras, mensajes y comentarios críticos sobre sus obras.

Esta biblioteca tiene una cuidada Interfaz de Usuario, desarrollada bajo la dirección de esta doctoranda, en la que se han puesto en práctica, sistemáticamente, todas las ideas sobre diseño de interfaces desarrolladas a lo largo del trabajo de investigación previo realizado para las Interfaces de Usuario de las Bibliotecas del Siglo de Oro.

Desde su inauguración, la Biblioteca Virtual Gallega ha recibido un sorprendentemente alto número de visitas, tanto de los autores, como del público, siendo en la actualidad la biblioteca virtual de referencia en la comunidad gallega. Hay que destacar, que para muchos de los autores y autoras actuales, la Biblioteca Virtual y la página Web, que en ella tienen, han constituido su primer contacto con Internet, y que éste les ha resultado extremadamente sencillo gracias a la amigabilidad de la Interfaz de Usuario que hemos desarrollado.

Los testimonios de numerosos autores y autoras, los mensajes de felicitación recibidos de la comunidad de escritores y lectores, y el alto y creciente número de visitas, además de las evaluaciones recibidas en los trabajos de publicación [73, 79, 82], nos permiten afirmar que la Interfaz de Usuario desarrollada cumple en alto grado con todos los requisitos de amigabilidad y facilidad de uso que se requieren en una Biblioteca Digital. Podemos concluir, por tanto, que, aunque de modo indirecto, estos datos permiten afirmar que las ideas desarrolladas en esta tesis sobre diseño de Interfaces de Usuario son, no sólo adecuadas, sino de gran interés y utilidad.

3.7 Resumen

En este capítulo se han descrito las tres técnicas de diseño de Interfaces de Usuario que han sido fruto de una parte del trabajo de investigación realizado en esta tesis, y se ha puesto de manifiesto la necesidad de Interfaces de Usuario que permitan expresar consultas de diferente nivel de complejidad para adecuarse a diferentes tipos de usuarios.

Además, hemos esquematizado y descrito el proceso de "ensayo y error" que hemos seguido en el diseño e implementación de los prototipos intermedios de Interfaces de Usuario hasta llegar a usar estas tres técnicas. Queda demostrada así su utilidad en el diseño de interfaces.

Hay que destacar, que cuando se realizó el diseño e implementación de la Biblioteca Virtual Galega [13], hicimos ya un uso exhaustivo de estas tres técnicas y el resultado ha sido que, desde el primer día, hemos tenido un alto número de visitas que han utilizado todas las secciones de la biblioteca. Esto, sin duda, implica que todas ellas son accesibles y fáciles de usar por cualquier usuario.

Capítulo 4

Estado del Arte en aproximaciones previas: Sistemas de Información Federados, Z39.50 y OAI-PMH

4.1 Introducción

Para realizar el Sistema de Acceso Integrado a las tres bases de datos documentales descritas empezamos por estudiar las aproximaciones que se han desarrollado para abordar este tipo de problema. Encontramos tres aproximaciones, muy diferentes entre sí, cada de las cuales trata de resolver la localización y el acceso a fuentes de datos heterogéneas, desde su propia perspectiva. Estas tres aproximaciones son: los Sistemas de Información Federados, el protocolo Z39.50 y la iniciativa de Open Archives (OAI), *Protocol for Metadata Harvesting* (OAI-PMH).

En este capítulo se presenta una breve descripción de cada uno de estos tres campos de investigación por separado y se termina el capítulo con una valoración de la adecuación de estas aproximaciones al problema que concierne a esta tesis.

4.2 Sistemas de Información Federados

En los 90 surgió la necesidad de combinar los datos almacenados en distintos sistemas de gestión de bases de datos. De hecho, aunque algunos de los primeros trabajos sobre estos *sistemas multi-base de datos* aparecieron en 1985, no fue hasta 1990 cuando se definió el término de *bases de datos federadas* para caracterizar a las técnicas utilizadas para proveer de un sistema integrado de acceso a un conjunto distribuido y heterogéneo de bases de datos autónomas. En esta época se definieron también otros conceptos de

importancia, como la distinción entre sistemas multi-base de datos y sistemas federados, y entre sistemas fuertemente acoplados y débilmente acoplados.

En los últimos años, han sido muchos los retos a los que ha tenido que enfrentarse el campo de investigación de Federación de Bases de datos. Por ejemplo, los problemas técnicos de la integración producidos por incompatibilidades de redes de datos han desaparecido, prácticamente, con la llegada de Internet, y la distribución física se ha convertido en algo manejable gracias a herramientas como CORBA y Java. Por otro lado, el número potencial de fuentes de datos se ha incrementado en gran medida, principalmente debido al uso del *World Wide Web* como sistema de publicación de datos. Esto último ha producido una gran heterogeneidad, que ha provocado cambios en distintas facetas, como puede ser la integración de esquemas o el acceso a la información a través de lenguajes de consulta. Aparte de esto, ha surgido un nuevo problema: la integración de fuentes Web se realiza comúnmente sin notificárselo a la fuente, y si la fuente cambia su esquema (o formato) no se lo puede comunicar al sistema federado, aun estando interesada en ello.

Aparte de todo esto, se han producido cambios en los paradigmas de desarrollo de software, lo que modifica los requisitos de las técnicas de integración de información. Las técnicas actuales se basan en el uso de componentes de integración llamados *mediadores,* que acceden a las fuentes o a otros mediadores bajo demanda. Las fuentes, a su vez, se encuentran encapsuladas por unos componentes llamados *wrappers*, que son capaces de presentar la información en un formato que el cliente (mediador) necesita. Estas ideas de sistemas basados en conjuntos de pequeños componentes interactivos semiautomáticos, se encuentran en muchos de los proyectos de investigación actuales.

La integración de los diferentes esquemas de las bases de datos sigue siendo un tema de investigación actual. En los últimos años, el uso de ontologías para federar bases de datos proporciona una interesante forma de conciliar los diferentes esquemas conceptuales de bases de datos a integrar. Así, existen muchos trabajos de investigación que enfatizan el uso de ontologías como una forma de integrar bases de datos independientes. Véase, por ejemplo, [19, 36].

Como efecto colateral se ha producido una redefinición de los conceptos comúnmente usados en estos contextos. 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. Otro de los cambios se produce en la clasificación de los distintos sistemas de integración de información. Veremos a continuación una posible clasificación de estos sistemas basada en los paradigmas actuales.

4.2.1 Tipos de Sistemas de Información Federados

En este apartado se presenta la clasificación realizada en [17] de los sistemas de información en tres tipos: sistemas de información débilmente acoplados, sistemas de bases de datos federadas y sistemas de informació n basados en mediadores. La Tabla 2 describe brevemente sus características.

Tabla 2. 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 de 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	Cualquiera
Métodos de acceso	Lenguaje de consulta	Lenguaje de consulta	Cualquiera
Restricciones de acceso	No	No	Sí
Acceso de escritura	Sí	Sí	No
Acoplamiento	Débil	Fuerte	Fuerte
Tipos de integración semántica	Colecciones	Colecciones y fusiones	Colecciones, fusiones y a veces abstracciones
Metadatos necesarios	Técnicos, infraestructuras	Lógicos, técnicos, semánticos	Lógicos, técnicos, semánticos
Bottom-Up Vs. Top-Down		Bottom-Up	Top-Down
Capacidad de evolución	Altas	Bajas	Altas

Resumiendo, la clasificación dada en [17] para los Sistema de Información Federados se presenta en la Tabla 3.

Tabla 3. Clasificación de los Sistemas de Información Federados

Sistemas de Información Federados	• Esquemas federados	• Con componentes que no son bases de datos o con lenguajes de consulta restringidos	**Sistemas de Información basados en Mediadores**
		• Compuesta sólo por bases de datos	**Sistemas de Bases de Datos Federadas**
	• Esquemas no federados	**Sistemas de Información débilmente acoplados**	

4.2.2 Sistemas de Información débilmente acoplados

Los Sistemas de Información débilmente acoplados no ofrecen un esquema federado, sino, solamente, un lenguaje de consulta para acceder a los componentes. Esto tiene la ventaja de que los componentes no pierden autonomía para participar en una federación. Por otro lado, no se ofrece transparencia de esquema, ni de localización: el usuario se ve obligado a seleccionar el componente involucrado y el campo en particular en el esquema de ese componente, en sus consultas.

Al proporcionar un lenguaje de consulta uniforme, el sistema federado soluciona los problemas técnicos y de lenguaje. Los conflictos lógicos tienen que ser resueltos por el usuario de los servicios en la capa de presentación. El usuario va a ser el responsable de toda la integración de los datos, con todos los aspectos de colecciones, fusiones y abstracciones.

La capa de federación es independiente del diseño lógico de los componentes. Como no existe un esquema global, los cambios de los componentes del esquema no afectan al sistema. Sin embargo, la falta de integración lógica lleva a diversas dependencias entre las aplicaciones y los sistemas componentes, con todos los efectos negativos conocidos de los sistemas en dos capas.

En la literatura, los sistemas de información débilmente acoplados son llamados sistemas multi-base de datos.

4.2.3 Sistemas de Bases de Datos Federadas

Los sistemas de bases de datos federadas proporcionan la funcionalidad clásica de los sistemas de bases de datos. Esto incluye acceso de lectura y escritura para la gestión de los datos. El término *bases de datos* indica la relación con sistemas clásicos de bases de datos: los componentes de los sistemas de bases de datos federadas son fuentes estructuradas, que son accedidas a través de lenguajes de consulta. Pueden existir diferencias entre los lenguajes de consulta de los distintos sistemas componentes, pero se asume siempre la existencia de acceso a través de un lenguaje de consulta.

Generalmente, se pierde cierta autonomía. Por ejemplo: notificación de cambios, acceso a metadatos de lógica del sistema o información de planificación para la gestión global de transacciones.

Los sistemas de bases de datos federadas, son sistemas de información fuertemente acoplados. Son construidos con técnicas *Bottom-Up* mediante la aplicación de alguna técnica de integración de esquemas. Como sistemas fuertemente acoplados, esos sistemas federados van a ofrecer transparencia completa de localización y de esquema para sus usuarios. Sin embargo, estos

sistemas suelen tener una arquitectura estática que impide la fácil evolución del sistema, debido a la dependencia de los procesos de integración de esquemas que no permiten la adición ni la substracción sencilla de componentes, ni tampoco una gestión sencilla de cambios en el sistema.

4.2.4 Sistemas de Información basados en Mediadores

Los Sistemas de Información basados en Mediadores son sistemas fuertemente acoplados, por lo tanto un esquema federado es usado para proveer acceso integrado a la información de los distintos componentes (heterogeneidad semántica). Una diferencia obvia con los Sistemas de Bases de datos Federadas, es el acceso de sólo lectura a los datos. A parte de eso, los sistemas basados en mediadores suelen ser construidos *Top-Down*, de acuerdo a las necesidades de información. En relación con esto está la visión de los mediadores como servicios construidos y ofrecidos a los clientes. Esto denota la importancia del requisito de flexibilidad respecto a la evolución del sistema. Como mínimo debe ser posible añadir o eliminar componentes de un modo sencillo, debido a que las fuentes en un sistema de información basado en mediadores mantienen, típicamente, autonomía completa de comunicaciones.

Una clasificación más detallada de los sistemas de este tipo revela ciertas diferencias. En la literatura, se consideran tanto componentes estructurados como componentes semi-estructurados o no estructurados. La heterogeneidad de los métodos de acceso es un tópico importante de investigación, por ejemplo refiriéndose a patrones de enlazado (presentes en la integración de fuentes Web) y las capacidades de consulta restringidas. Los sistemas basados en mediadores pueden considerar diversos mecanismos de integración como abstracción, agregación, o acercamientos de metainformación. Generalmente, un mediador no soluciona todos estos aspectos pero debería ocuparse, al menos, de alguno de ellos.

4.2.5 Proyectos relevantes

A continuación se describen seis de los Sistemas de Información Federados más destacados, incluyendo un breve resumen individual.

Garlic

Garlic [18] es un proyecto de IBM Research que trata sistemas de información multimedia de gran tamaño, considerando sistemas de componentes especializados para almacenar y buscar tipos de datos específicos, como los sistemas de gestión de imágenes. Los esquemas de exportación de los componentes están fuertemente integrados en un modelo de datos orientado a

objetos. Garlic no considera la heterogeneidad en los esquemas, pero las diferencias entre las capacidades de las consultas son manejadas por un potente optimizador de consultas. De hecho, incluso los cambios de estas capacidades no afectan al mediador. Garlic requiere wrappers bastante potentes, debido a que la ejecución de la consulta depende de una comunicación interactiva, entre el mediador y los wrappers, sobre las capacidades de los componentes. Una vez que los wrappers están implementados, la autonomía de los componentes es alta.

Information Manifold

Este prototipo [39], desarrollado por AT&T durante los años 95 y 96, integra fuentes de datos web estructuradas. Los temas principales del proyecto son la descripción de las fuentes y el procesamiento de la consulta. El esquema del mediador (el "*word model*") hace transparente la heterogeneidad de los componentes. Este esquema está diseñado de acuerdo con las necesidades de información de la parte de arriba del sistema (top-down). Cada concepto de un esquema componente está relacionado con el esquema del mediador usando un potente lenguaje declarativo, siguiendo el acercamiento *LaV* (*Lo Local como Vista*). A partir de una consulta dada, el sistema usa las descripciones para identificar las fuentes relevantes, ejecuta las subconsultas y recolecta los resultados. El resto de la computación o fusión de objetos que sea necesarios deberán ser aplicados por el usuario. Debido a la independencia entre mediador y esquemas componentes y la descripción explícita de su relación, los componentes mantienen su autonomía y el sistema puede evolucionar de un modo sencillo.

SIMS

SIMS ("*Search In Multiple Sources*") [6] usa una ontología para proporcionar un punto de acceso global a los componentes heterogéneos. Esta ontología usa una lógica descriptiva como modelo de datos. La expresividad de la lógica descriptiva permite integrar fuentes con modelos de datos muy distintos. El acercamiento basado en ontologías soluciona particularmente la heterogeneidad semántica. El procesado de consultas en SIMS considera fuentes de datos replicadas, de modo que las consultas pueden ser respondidas incluso aunque falte algún componente. SIMS, al igual que *Information Manifold,* permite la autonomía de los componentes y posee una alta capacidad de evolución. Esto lo consigue porque los componentes son integrados por conceptos relacionados en la ontología del mediador, de forma independiente a otros componentes y al esquema del mediador en sí mismo

(*Lo Local como Vista*). Las modificaciones o extensiones del sistema se pueden hacer de forma independiente.

TSIMMIS

TSIMMIS ("*The Stanford-IBM Manager of Multiple Information Sources*") [62] permite la integración de fuentes de datos heterogéneas. Puede estar compuesto de componentes estructurados o semi-estructurados. En contraste con los proyectos mencionados anteriormente, TSIMMIS no proporciona un esquema de mediador, sino que propaga todos los esquemas de los wrappers componentes hasta el usuario. La heterogeneidad de modelo de datos está resuelta mediante modelos de intercambio de objetos semi-estructurados ("*Object Exchange Model*" - *OEM*), un modelo simple con objetos y anidamiento de objetos (pero sin herencia ni clases). Para resolver conflictos semánticos entre componentes se propone un servicio de diccionario, pero no se implementa. Uno de los componentes del TSIMMIS permite la especificación de vistas de integración usando un lenguaje específico llamado *MLS* ("*Mediator Specification Languaje*"). También se encuentran disponibles mecanismos particulares para la fusión de objetos o abstracciones. TSIMMIS usa semánticas no estándar para los términos. En su lenguaje un mediador es simplemente una vista de integración. Otra de las características es que promueve el uso de wrappers gruesos, delegando parte de las tareas que son habitualmente consideradas como parte del mediador, en el wrapper, como por ejemplo la descomposición de consultas y la compensación de capacidades de consulta ausentes.

OBSERVER

OBSERVER [46] es un Sistema de Información Global que permite que el usuario pueda realizar consultas sin que tenga que ocuparse de la selección y localización de las fuentes que conviene consultar y al que se le ofrece una Interfaz de Consulta basada en Ontologías. Las ontologías que utiliza OBSERVER están definidas con lenguajes basados en lógica descriptiva (en particular CLASSIC) y las estructuras que relacionan las ontologías entre sí y con las fuentes de datos confieren la suficiente flexibilidad al sistema OBSERVER para adaptarse al contexto cambiante de la Web.

Una persona que desee realizar una consulta tiene a su disposición una colección de ontologías de entre las que selecciona una para utilizar su vocabulario en la formulación de una pregunta. El sistema OBSERVER se encarga de todo el proceso que va desde analizar la pregunta, enviarla a las distintas fuentes asociadas a la ontología elegida y combinar

convenientemente las respuestas que obtiene para, finalmente, mostrar el resultado en el formato solicitado.

El sistema responde a la pregunta accediendo sólo a los datos de los depósitos relacionados directamente con la ontología seleccionada. Si el usuario desea más resultados relevantes de la misma pregunta, puede solicitarlo y el sistema procederá a traducir la pregunta original al vocabulario de otra ontología relacionada y repetirá desde ese punto los mismos pasos realizados con la primera ontología. Para poder realizar esta traducción, OBSERVER mantiene una colección de vinculaciones semánticas entre pares de ontologías que recogen las relaciones en términos de cada ontología.

Sistema Cooperativo para Integración de Fuentes Heterogéneas de Información

Se trata de un Sistema Federado de Bases de Datos [55] que presenta la característica distintiva e innovadora de que el acceso integrado lo hace en tiempo real, es decir, que la consulta la procesa accediendo directamente a las bases de datos preexistentes, que interoperan formando un Sistema Cooperativo, y también consultando un almacén de datos (Data Warehouse) [3], en el que periódicamente se vuelcan y consolidan los datos de esas bases de datos.

Además, han desarrollado el modelo BLOOM [1, 2], que utilizan como modelo canónico de datos del sistema.

Algunos otros aspectos en los que trabajan son: Control de acceso cooperativo, Gestión de transacciones cooperativas y Evolución de esquemas.

4.3 Z39.50

Detallamos a continuación, con cierto detalle, este protocolo por ser el primero y haberse convertido en el estándar más extendido de integración de Bibliotecas Digitales.

Su nombre "oficial" es: "Information Retrieval (Z39.50); Application Service Definition and Protocol Specification, ANSI/NISO Z39.50-1995", también llamado The ANSI/NISO Z39.50 Search and Retrieval Protocol, correspondiente al estándar ISO23950. Sin embargo, comúnmente se conoce por "Z39.50".

Z39.50 proviene de un comité formado por la National Information Standards Organization (NISO) y la American National Standards Institute (ANSI), relacionado con bibliotecas, publicaciones y servicios de información

que fue llamado Z39. Por otra parte, todos los estándares NISO están numerados secuencialmente y éste es el estándar número 50 desarrollado por NISO.

Z39.50 tiene como objetivo permitir la recuperación de información en fuentes remotas y heterogéneas. Otros objetivos que actualmente pueden ser atribuidos al estándar, tales como la estandarización semántica de la información dentro de comunidades de usuarios, han sido realmente inducidos por la utilización que de sus diversas implementaciones han hecho estos usuarios.

El estándar define un conjunto de servicios y especifica un protocolo orientado a la búsqueda y recuperación de información en recursos heterogéneos y remotos. El protocolo especifica un conjunto de mensajes (APDU) y reglas de intercambio que permiten a un cliente (llamado *origen* en el estándar), mediante una consulta, solicitar recursos que se encuentran en un servidor (llamado *objetivo* en el estándar) y recuperar registros identificados a través de la búsqueda. Una sesión de intercambio de mensajes entre el *origen* y el *objetivo* es llamada *Z-Asociación*, que es siempre iniciada por el *origen*, aunque puede ser terminada por el *origen* o el *objetivo*. La arquitectura propuesta por Z39.50 se muestra en la Fig. 40.

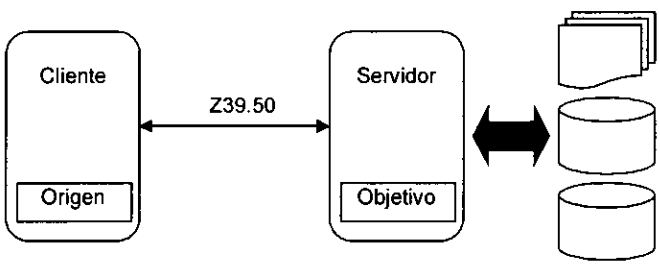


Fig. 40. Arquitectura de un sistema basado en Z39.50

Servicios

La versión 3 de Z39.50 define los siguientes 13 servicios: *Init*, *Search*, *Present*, *Segment*, *Access Control*, *Delete Result Set*, *Resource Control*, *Resource Report*, *Trigger Resource Control*, *Sort*, *Scan*, *Extended Service*, *Close*. Estos servicios son agrupados lógicamente en 11 facilidades que son *Initialization*, *Search*, *Retrieval*, *Result Set Delete*, *Browse*, *Sort*, *Access Control*, *Accounting/Resource Control*, *Explain*, *Extended Services* y *Termination*.

Los servicios Z39.50 se realizan por el intercambio de mensajes entre el origen y el objetivo. Un mensaje es una solicitud (request) o una respuesta (response). Los servicios se clasifican en servicios confirmados, no confirmados y condicionalmente confirmados. Un servicio confirmado exige que toda solicitud tenga una respuesta. Los servicios *Init*, *Search*, *Present*, *Access Control*, *Resource Report*, *Delete Result Set*, *Sort*, *Scan*, *Extended Service*, y *Close* son servicios confirmados . Los servicios no confirmados envían una solicitud y no tienen respuesta. *TriggerResourceControl* y *Segment* son los servicios no confirmados. Los servicios condicionalmente confirmados son aquellos que dada una solicitud pueden o no recibir una respuesta. *ResourceControl* es el único servicio condicionalmente confirmado.

Operaciones

El estándar define 8 tipos de operaciones: *Init, Search, Present, Delete, Scan, Sort, Resource-report* y *Extended-service.* Una operación se inicia con una solicitud de servicio y termina con la respuesta a la misma. Sólo el *origen* puede iniciar operaciones. Desde el punto de vista del *objetivo* una operación comienza cuando recibe una solicitud de servicio y termina cuando envía la respuesta al *origen*.

No todas las solicitudes de servicio inician una operación. Por ejemplo, una solicitud del servicio *ResourceControl* no inicia operación alguna.

Protocolo

Z39.50 especifica un protocolo de red orientado a sesión definido dentro de la capa de aplicación del modelo OSI. El estándar establece el formato de los mensajes y el procedimiento a seguir para el intercambio de mensajes entre el *origen* y el *objetivo*. En el estándar los mensajes se denominan *Aplication Protocol Data Unit* (APDU). El formato de los mensajes se especifica mediante el uso del estándar ISO 8824, *Abstract Syntax Notation One* (ASN.1).

El procedimiento a seguir para el intercambio de mensajes entre el *origen* y el *objetivo* se norma a través de las tablas de estado. En las tablas de estado se definen explícitamente cuáles son los estados posibles en que pueden encontrarse el *origen* y el *objetivo* durante su interacción.

4.3.1 Interoperabilidad de Z39.50

Z39.50 puede ser implementado en cualquier plataforma, ya que no norma detalles de implementación. Esto significa que puede ser soportado en diferentes sistemas computacionales con diferentes sistemas operativos, hardware, motores de búsqueda, sistemas de administración de bases de datos, u otros recursos, de ahí su gran interoperabilidad.

El objetivo fundamental del estándar es lograr la interoperabilidad en el proceso de búsqueda y recuperación de información. Para alcanzar este objetivo cuenta con tres mecanismos: la negociación durante el servicio *Init*, la facilidad *Explain* y los perfiles (profiles).

Negociación durante el servicio Init

El servicio *Init* permite al *origen* negociar con el *objetivo* las condiciones bajo los cuales se llevará a cabo la búsqueda y recuperación de la información proponiendo una negociación. Esta negociación permite al *origen* proponer parámetros de funcionamiento al *objetivo* y permite al *objetivo* a su vez proponer valores alternativos para estos parámetros. Tanto el *origen* como el *objetivo* pueden, en dependencia de los parámetros negociados, aceptar o rechazar el establecimiento de su interacción. Como ejemplo de parámetros negociables tenemos las versiones (Version) y servicios (Options) Z39.50 soportados y el tamaño en bytes de los mensajes a intercambiar (Preferred-message-size).

La Facilidad Explain

La facilidad *Explain* permite a un *origen* obtener detalles sobre la implementación de un *objetivo,* las bases de datos disponibles para la búsqueda, sintaxis en la cual se envían los resultados y otros elementos informativos imprescindibles para llevar a cabo conjuntamente la búsqueda y recuperación de información. Para lograr esto *Explain* hace uso de los servicios de las facilidades *Search* y *Retrieval.* Un *objetivo* que soporta la facilidad *Explain* permite el acceso a una base de datos llamada *IR-Explain-1*, referida como base de datos Explain (Explain database), en la cual se almacena esta información. El contenido de esta base de datos se establece por Z39.50 mediante categorías de información predefinidas. El estándar especifica el formato en que deben ser trasmitidos los récord de la bases de datos *Explain* hacia el *origen* (Explain Record Syntax).

Profiles

Los profiles son conjuntos de funcionalidades y elementos que especifican las distintas comunidades de usuarios para aumentar la interoperabilidad de la búsqueda informativa dentro de la misma. Un profile puede incluir entre otros: un conjunto de atributos y combinaciones de atributos para la búsqueda, esquemas y sintaxis de recuperación, y servicios obligatorios a soportar. Hoy en día los profiles han sido la vía más exitosa para lograr la interoperabilidad en este ámbito. *BATH* [63] es un profile definido por la comunidad de los profesionales de la información para el intercambio de información dentro de la misma. No existen en Z39.50 especificaciones sobre los profiles, estos son más bien una consecuencia de su utilización.

El modelo abstracto de datos

Para manipular los diferentes formatos y estructuras presentes en recursos heterogéneos y acceder a estos, Z39.50 usa un modelo abstracto de representación de los recursos que le permite manipularlos de una forma homogénea. El término base de datos, como es usado en Z39.50, se refiere a un conjunto de Registros Abstractos, donde un registro es simplemente una colección de información relacionada, una vista virtual del recurso, que permite ocultar la forma en la cual este recurso está almacenado físicamente. La Fig. 41 ilustra este concepto.

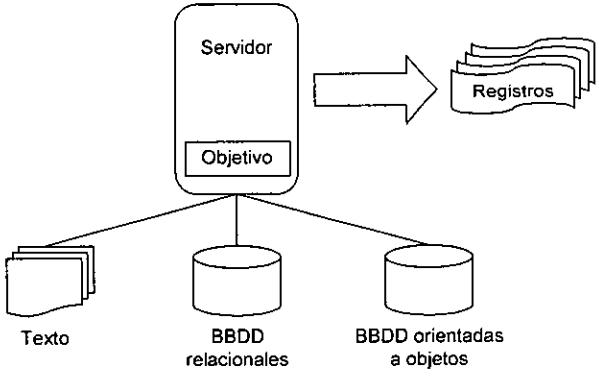


Fig. 41. Modelo abstracto de bases de datos

El estándar ofrece dos mecanismos para la búsqueda y recuperación de información:

- Los *puntos de acceso* para la formulación de la solicitud: Un *punto de acceso* es una llave única o no que puede ser especificada, sola o en combinación con otros puntos de acceso, en una solicitud de recursos. Los

puntos de accesos son publicados como *conjunto de atributos* (Attributes Set) estandarizados y son de dominio específico. Los conjuntos de atributos forman parte de los perfiles.

- Los *puntos de recuperación* para la recuperación de los resultados: Los *puntos de recuperación* son elementos de los Registros Abstractos o virtuales. Los puntos de recuperación son publicados como esquemas (schemas) estandarizados y son devueltos físicamente como resultado de una solicitud en un formato especificado denominado sintaxis de récord (Record Syntax).

Puntos de Acceso

Los puntos de acceso son de vital importancia para llevar a cabo la formulación de una solicitud y mantener la interoperabilidad. Necesitamos referirnos de forma única a los atributos de los recursos que queremos buscar y recuperar.

Ilustremos mediante un ejemplo los puntos de accesos. Una base de datos A coloca la información del título en el campo "Tit"; la base de datos B la coloca en el campo "Titulo". Estos dos campos denotan con nombres diferentes, la misma propiedad - el titulo de los documentos. Un atributo, por ejemplo *Title*, perteneciente a un conjunto de atributos, resultado del acuerdo de una comunidad de usuarios, permitirá hacer referencia de manera abstracta a dicha propiedad. Esta solicitud abstracta será convertida posteriormente en una solicitud específica para la base de datos A en la cual *Title* es sustituido por "Tit" y otra para la base B donde *Title* es sustituido por "Titulo".

Bib-1 [12] es el conjunto de atributos más utilizado en los sistemas de búsqueda y recuperación de información basados en Z39.50. BIB-1 es resultante del acuerdo de una comunidad de usuarios y modela los campos de los recursos bibliográficos. Otros conjuntos de atributos lo son *Government Information Locator Service* (GILS) [32], y *Scientific and Technical Attribute Set* (STAS) [56]. La norma especifica un conjunto de atributos para la base de datos Explain, llamado Exp-1.

En el estándar se definen seis tipos de solicitudes (query):

- *Tipo-0:* de uso privado entre un *origen* y un *objetivo* puestos de acuerdo sobre el formato de la solicitud.
- *Tipo-1:* las solicitudes son expresadas por términos de búsqueda individuales, cada uno de los cuales tiene una lista de atributos. Los términos pueden estar enlazados por operadores booleanos y tanto los términos como los operadores son expresados en Notación Reversa Polaca (RPN). Este es un tipo obligatorio.

- *Tipo-2:* especificado por la ISO 8777 – (Commands for Interactive Text Searching).
- *Tipo-100:* especificado por la ANSI Z39.58 – (Common Command Language for Online Interactive Information Retrieval).
- *Tipo-101:* extensión de la solicitud de tipo-1 para la solicitud con uso de proximidad.
- *Tipo-102:* extensión de la solicitudes de tipo-1 para las solicitud con uso de pesos (ranked list querying).

La estructura de la solicitud de tipo-1 se especifica en el estándar mediante la notación ASN.1. Cada uno de los términos de búsqueda tiene asociado una lista de atributos que describe el contexto y la estructura en la cual el término de búsqueda se está aplicando.

Puntos de recuperación

Los puntos de recuperación son el mecanismo encargado de lograr la abstracción de los recursos manejados por el *objetivo*. Para lograr tal objetivo Z39.50 hace uso de los *esquemas* (schemas). Un *esquema* especifica cuales son los elementos de cada registro o récord del conjunto resultado que formarán parte de un Registro Abstracto. Los esquemas son información compartida por el *origen* y el *objetivo*, y su especificación puede ser de común acuerdo dentro de una comunidad de usuarios.

Esquema

A cada recurso se le asignan uno o varios esquemas, estos esquemas se encuentran almacenados en la base de datos Explain. Todo recurso tiene asociado un esquema por defecto. El *origen* le informa al *objetivo*, en una solicitud de presentación de registro (Service Present), de cuál es el esquema en el que desea recibir el conjunto resultado. En caso de que el *origen* no lo especifique, los registros se enviarán usando el esquema por defecto. Cuando el *objetivo* aplica un esquema al conjunto resultado, se obtiene un Registro Abstracto por cada registro, en el cual sólo aparecen los elementos especificados por el esquema.

La componente fundamental de un esquema es la Estructura de Registro Abstracta (ARS) que especifica cuáles son los elementos que serán incluidos en el Registro Abstracto. Esto es posible porque cada campo contiene un identificador llamado *tag* que lo identifica inequívocamente, y por el que puede ser identificado en una ARS.

Ejemplifiquemos la estructura de una ARS concreta de forma simplificada. Supongamos que una base de datos contiene los campos: "titulo", "autor", "año", "ISBN" y "editor", y la siguiente correspondencia entre el campo y un *tag* numérico asignado a cada uno de ellos como se muestra a continuación:

Campo	Tag
Titulo	1
Autor	2
Año	3
ISBN	4
Editor	5

y se tiene la ARS que sigue:

Tag	Obligatorio	Repetido	Definición
1	Sí	No	Título del libro
2	Sí	Sí	Autor o autores del libro
5	Sí	No	Pequeña descripción del libro

El primer elemento (Tag) de la ARS identifica el valor de tag asignado al campo que se desea incluir en el Registro Abstracto, el segundo elemento (Obligatorio) nos indica si la presencia del campo es obligatoria o no, el tercer elemento (Repetido) indica que el campo puede encontrarse repetidamente y, por último, el cuarto elemento (Definición) ofrece una breve descripción del tag.

Es posible definir un *tag* en términos de un par ordenado *(x,y)* - donde *x* representa el tipo de tag, e *y* el número de tag - con el propósito de agrupar los tag en conjuntos, los cuales son llamados en el estándar conjunto de tags (Tag Set). El estándar define y registra dos conjuntos de tag, *TagSet-M* y *TagSet-G* a los cuales le asigna 1 y 2 respectivamente como tipo de *tag* y deja abierto a definir por el usuario los *tags* a partir del 3. El conjunto *TagSet-M* incluye una serie de *tags* para describir meta información asociada a un récord. Por ejemplo, el *tag* (1,16) pertenece a este conjunto e indica la fecha de última actualización de un registro. El conjunto *TagSet-G* incluye tag genéricos que describen los campos. Por ejemplo, el *tag* genérico (2,1) describe "autor" y el *tag* (2,2) describe el "titulo".

Una vez que se tiene un conjunto resultado abstracto, se realiza la selección de los elementos a enviar al *origen* como resultado de una solicitud de presentación de resultados. Para esto el *origen* especifica cuáles son los elementos a seleccionar a través de los *tags* o de un Nombre de Conjunto de Elementos (*ESN*). Este último es la vía más usada. El *ESN* es un nombre conocido por el *origen* y soportado por el *objetivo*, y almacenado en la base de datos Explain. Supongamos que tenemos un *ESN* nombrado "AutorTitulo" que el *objetivo* define como la "selección del campo autor y el campo título de un conjunto de Registros Abstractos", el *origen* podrá solicitar este *ESN* para recuperar registros que contienen dichos campos. La selección de los campos

convierte un Registro Abstracto en otro Registro Abstracto, obteniéndose un conjunto resultado que sólo contiene los campos seleccionados.

Sintaxis de Registro

El proceso explicado anteriormente culmina con la codificación de cada Registro Abstracto, mediante la Sintaxis de Registro (Record Syntax), para su envió al *origen*. El proceso de conversión de los resultados a la Sintaxis de Registro se muestra en la Fig. 42.

Existe varias Sintaxis de Registro, definidas por las distintas comunidades de usuarios. En el marco de la comunidad de los profesionales de la información se encuentra definida la sintaxis MARC [44] para el intercambio de registros bibliográficos. La norma especifica una Sintaxis de Registro para la base de datos Explain llamada Explain-RS.

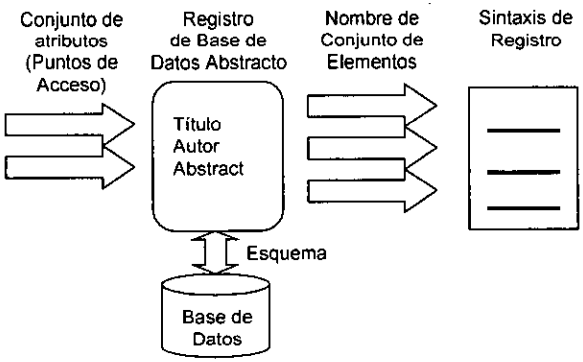


Fig. 42. Conversión de los resultados a las Sintaxis de Registro

4.3.2 Una sesión Z39.50

A continuación describimos el proceso de búsqueda y recuperación de información en un sistema basado en Z39.50 con el fin de ilustrar una sesión Z39.50 (Z-Asociación). Sin perdida de generalidad y con propósitos de simplificación, se consideran sólo los servicios básicos *Init*, *Search*, *Present* y *Close*. En la Fig. 43 se muestra gráficamente dicha sesión.

Inicialización (*Init*)

Por medio de una pagina Web o cualquier otro usuario que permita la comunicación con un *origen* Z39.50, se formula una solicitud. La primera operación que se realiza es establecer la conexión con el *objetivo* que permite

el acceso a la base de datos en cuestión. Para esto el *origen* envía una APDU *InitRequest* al *objetivo* en la cual están los parámetros necesarios para la negociación. Con esta APDU se inicia una Z-Asociación. El *objetivo* recibe la *InitRequest* enviada, la procesa y decide si acepta o no las condiciones propuestas.

El *objetivo* envía como respuesta al *origen una* APDU *InitResponse*, reescribiendo los parámetros enviados por el *origen*. El *origen* recibe la *InitResponse* y la procesa. Si el *objetivo* no acepta, termina la Z-Asociación. En caso contrario, el *origen* puede a su vez aceptar o no las condiciones del *objetivo*. Si no las acepta, le enviará una APDU *Close*, para terminar la Z-Asociacion. Si las acepta, pasará a una nueva fase de la interacción. Todos los mensajes intercambiados posteriormente se enmarcarán en la Z-Asociación que inició el *origen*.

Búsqueda (*Search*)

El *origen* envía una APDU *SearchRequest* que contiene, como parámetros, la solicitud formulada por el usuario. Como parte de esta APDU, también son enviados otros datos tales como la o las bases de datos que se quiere interrogar y la Sintaxis de Record en la cual se desea recibir el resultado de la búsqueda.

Esta APDU llega al *objetivo*, que la procesa y, mediante mecanismos establecidos al efecto, se realiza la búsqueda y recuperación del conjunto resultado correspondiente a la solicitud formulada por el usuario. El *objetivo* informa al *origen* de los resultados obtenidos mediante una APDU *SearchResponse*.

Presentación (*Present*)

El *origen* envía una APDU *PresentRequest* solicitado la presentación del conjunto resultado obtenido para mostrárselo al usuario. El *objetivo* a su vez enviará al *origen,* mediante una o más APDU *PresentResponse*, el conjunto resultado.

Terminación (*Close*)

Tanto el *origen* como el *objetivo* pueden terminar la Z-Asociación enviando una APDU *Close*, la cual puede ser respondida con otra APDU *Close* , que confirma la terminación de la Z-Asociación.

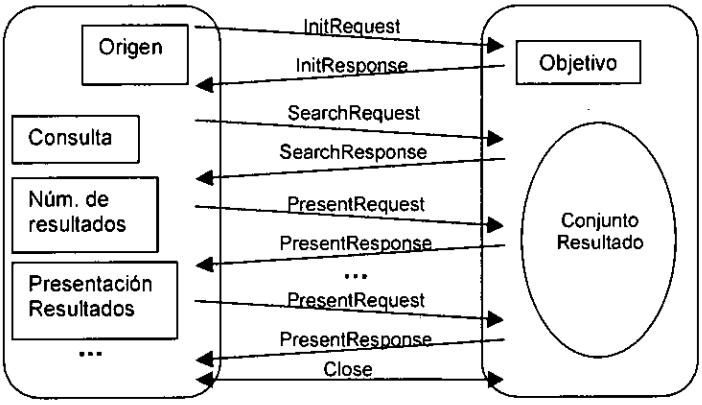


Fig. 43. Sesión Z39.50

4.3.3 Problemas de Z39.50

Son evidentes los puntos fuertes de Z39.50: puntos de acceso abstractos, capacidad para representar búsquedas complejas, capacidad para representar datos complejos, registros con elementos identificables, etc. Sin embargo, Z39.50 presenta ciertos inconvenientes, que ya fueron identificados por ZIG (Z39.50 Implementers' Group) en el informe de su reunión en el 2000 [54]. La mayoría de estos inconvenientes son debidos a que el contexto en el que apareció Z39.50 es muy diferente del contexto actual. Algunos de ellos son los siguientes:

- La documentación de Z39.50 es difícil de entender. El núcleo del protocolo no está explicado claramente.
- Z39.50 es difícil de implementar y carece de herramientas que simplifiquen la implementación.
- Las implementaciones que se han hecho de Z39.50 son diferentes, lo cual lleva a problemas de interoperabilidad.
- Algunas de las funciones de Z39.50 no son necesarias hoy en día y no hacen más que complicar el estándar.
- Z39.50 no está pensado para la Web (no tiene en cuenta la Web).
- Hoy en día, se tiende a no instalar más que un simple servidor Web.

4.4 Open Archives Iniciative

El OAI-PMH es un protocolo que proporciona interoperabilidad no inmediata (es decir, no es un protocolo para búsquedas online) entre fuentes que pueden ser documentos electrónicos, Bibliotecas Digitales o cualquier servidor que quiera exponer (hacer visibles) los metadatos de los documentos que tiene almacenados para un sistema que quiera recolectarlos.

El *Protocol for Metadata Harvesting* (PMH) de la *Open Archives Iniciative* (OAI) [49] proporciona un entorno de interoperabilidad independiente de la aplicación basado en la recolección de metadatos (metadata harvesting).

Los participantes de OAI descartan opciones como el protocolo de recuperación de información Z39.50 a favor de una solución más simple y menos costosa en términos de recursos computacionales [54]. Es decir, el *Open Archives Iniciative* ha apostado por la recolección de metadatos, en vez de por la integración del acceso real a los datos, y, así, ha definido el "protocolo para recolección de metadatos".

Esta apuesta por la recolección de los metadatos (datos sobre los documentos, como "autor", "fecha", "edición", etc.) para hacerlos accesibles al usuario, en vez de hacerle accesible los documentos completos, se justifica por las dificultades de realizar integraciones completas de Bibliotecas Digitales debido a la heterogeneidad de los documentos que contienen y a la lentitud de la red. Así Fox, justifica las ventajas del AOI-PMH en estos términos:

> *"... En la búsqueda simultánea[6], la necesidad de información del usuario, expresada en la forma de una pregunta, es enviada por el sistema de búsqueda simultánea a todos los sitios que puedan contestar la pregunta. Una vez que los sitios han completado la búsqueda y generado resultados, el usuario puede ver cada sitio que tenga algún contenido relevante (ver [51]), o puede hacer una fusión de los resultados generando una sola lista fusionada. Aunque la búsqueda simultánea produce resultados en el momento, tal tendencia no es de alta prioridad en el mundo de las tesis electrónicas. Al mismo tiempo, la búsqueda simultánea se puede dificultar por el "time out" de los sistemas locales y por la administración de los respaldos, si algunos sitios remotos están caídos o son lentos en enviar las respuestas. Aún, en el mejor de los casos, la búsqueda simultánea suele ser lenta (debido a la red) y tiene problemas para manejar una*

[6] Término utilizado por Fox para referirse al acceso integrado a un conjunto de fuentes.

gran diversidad de representaciones de datos en sitios remotos, lo que repercute, en algunos casos, en una baja calidad de información. Aunque a pesar de todo esto en ocasiones es imprescindible un servicio de búsqueda simultánea, el Open Archives Iniciative ha apostado por la recolección de metadatos. ..." (Fox, [29])

El OAI-PMH define el intercambio de solicitudes y de metadatos entre el servidor de documentos digitales y un programa recolector externo. En OAI existen las instituciones llamadas Proveedores de Datos (Data Providers), que son bancos 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 en OAI 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 y el programa recolector (harvester) del Proveedor de Servicios, para la transferencia de metadatos es unidireccional. El Proveedor de Servicios hace solicitudes al Proveedor de Datos, el cual responde enviando metadatos. Las solicitudes del Proveedor de Servicios son realizadas a través del protocolo http, usando comandos codificados a través de los métodos GET o POST. Las solicitudes son respondidas por el Proveedor de Datos a través del envío de los metadatos de los documentos almacenados, codificados en XML. El OAI-PMH establece que el Proveedor de Datos debe publicar sus metadatos, como mínimo, en formato Dublin Core [21] no calificado. Por su parte, el Proveedor de Datos puede ofrecer otros formatos de metadatos más complejos, como por ejemplo MARC [44].

Los metadatos de un Proveedor de Datos están clasificados por registros. Existe un registro por cada documento almacenado en el Proveedor de Datos, de manera que los metadatos que envía un Proveedor de Datos tienen un identificador único, formado por el identificador del Proveedor de Datos más el identificador del registro. Cada registro contiene también una marca temporal (datestamp) que indica la fecha de creación o de última actualización del documento asociado a ese registro. Esta marca temporal es la clave que permite la recolección automática de los metadatos del Proveedor de Datos a partir de una fecha determinada, permitiendo, por tanto, la sincronización entre los registros del Proveedor de Datos y de un Proveedor de Servicios que lleva a cabo un servicio de acceso simultáneo a metadatos de documentos almacenados en diversos Proveedores de Datos.

El protocolo prevé seis tipos de peticiones que un programa robot de un Proveedor de Servicios puede enviar a un Proveedor de Datos para recolectar metadatos de documentos allí almacenados. En la Fig. 44 se muestra la arquitectura general de PMH.

Los seis tipos de peticiones de las que hablábamos son las siguientes:

- *Identify:* Obtiene datos administrativos sobre el Proveedor de Datos, su política de publicación de documentos, su alcance, etc.
- *ListSets:* Lista de clasificaciones (Sets) a través de las que están organizados los documentos en el Proveedor de Datos.
- *ListMetadataFormats:* Lista de formatos de metadatos a través de los que los metadatos de los documentos almacenados en el proveedor de datos pueden ser presentados.
- *ListIdentifiers:* Lista los identificadores de registros almacenados en el Proveedor de Datos, pudiendo opcionalmente limitar estos registros según una fécha o una clasificación (Set).
- *ListRecords:* Lista los metadatos de los registros almacenados en el Proveedor de Datos según un formato de metadatos especificado. Se listan todos los registros, todos los que pertenezcan a un "set" o todos a partir de una fecha.
- *GetRecord:* Obtiene los metadatos de los registros almacenados según un formato de metadatos, dado un identificador de registro.

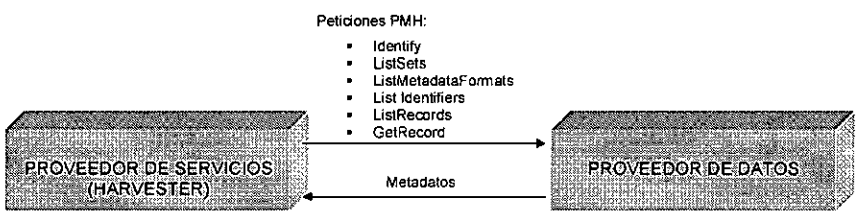


Fig. 44. Arquitectura de PMH

Uno de los informes de la *Reunión Internacional de Especialistas en Información Científica Digital* celebrada en São Paulo durante el 2002 [45] se ponía de manifiesto que:

- en Internet existe una gran cantidad de información sin organizar,
- la información disponible es sobre una infinidad de temas y que está disponible en diferentes idiomas,

- los mecanismos de búsqueda generales indexan la Web periódicamente, de forma automática, ciegamente y sin comprender el tema de una página, simplemente extrayendo palabras del texto HTML de la página, y
- estos mismos mecanismos de búsqueda sólo buscan en páginas HTML estáticas, sin considerar la gran cantidad de información almacenada en bases de datos disponibles a través de Internet. Es decir, existe una gran parte de la Web, la llamada Web oculta ("Deep Web"), que queda inaccesible.

Por todo esto, el OAI-PMH mejorará sensiblemente la eficiencia en la búsqueda de fuentes de información relevantes. Sin embargo, una vez que las fuentes de información hayan sido encontradas, es necesario poder acceder a la información que contienen.

4.5 Adecuación de las tres aproximaciones expuestas a la Integración de Bibliotecas Digitales

Para valorar la adecuación de estas tres aproximaciones es preciso empezar por detallar cuáles eran los requisitos del sistema de integración que queríamos desarrollar para integrar el acceso a nuestras tres Bibliotecas Digitales del Siglo de Oro, y que consideramos imprescindibles en cualquier Sistema de Acceso Integrado a Bibliotecas Digitales.

Ya en la introducción se detallaron los tres requisitos fundamentales que perseguíamos:

- **El sistema debe escalable**
- **El sistema debe acomodar los cambios fácilmente**
- **La Interfaz de Usuario debe ser intuitiva, fácil de usar, potente y flexible**

Además de estos tres requisitos fundamentales, hay otros aspectos a considerar, menos relevantes para la integración de nuestras tres Bibliotecas Digitales concretas pero muy importantes para la integración de Bibliotecas Digitales, en general:

- **Conservación de capacidades para Recuperación de Textos:** Un Sistema de Acceso Integrado a Bibliotecas Digitales debe permitir explotar las capacidades de Recuperación de Textos de cada una de las Bibliotecas Digitales componentes.
- **Identificación de las fuentes de datos:** Si se quiere que un Sistema de Acceso Integrado tenga éxito y que diferentes Bibliotecas Digitales deseen

integrarse en él, es fundamental no restarle protagonismo a las Bibliotecas Digitales que participan, de modo que un usuario siempre sea consciente de a qué Biblioteca pertenecen los documentos que recupera con el Sistema de Acceso Integrado.

- **Conservación de duplicados:** En bases de datos convencionales estructuradas, si una entidad tiene sus datos registrados en dos bases de datos, para el usuario sólo es relevante recuperar una instancia de dichos datos. Sin embargo, en Bibliotecas Digitales, una misma obra, puede estar en dos bibliotecas distintas, pero para el usuario es relevante recuperar las dos por diferentes motivos: (1) Si se trata de dos ejemplares diferentes que pueden tener diferentes estados de conservación, exlibris, y, por supuesto, diferentes signaturas para identificarlos dentro de cada biblioteca; (2) Aún en el caso de que se tratase del mismo ejemplar físico recogido en dos Bibliotecas Digitales diferentes, probablemente el conjunto de datos estructurados y no estructurados (número de páginas digitalizadas, o cantidad y calidad del texto transcrito) varían de una Biblioteca Digital a otra.

Para la integración de nuestras tres Bibliotecas Digitales del Siglo de Oro, estos tres últimos requisitos no eran fundamentales ya que, primero, nuestras bases de datos no tienen grandes capacidades de Recuperación de Textos que haya que conservar. Por otro lado, las tres Bibliotecas Digitales fueron desarrolladas por el Laboratorio de Bases de datos, así como el Sistema de Acceso Integrado, por lo que su uso no nos resta protagonismo. Finalmente, las tres colecciones de documentos son diferentes entre sí, por lo que no existen duplicados. Sin embargo, estos tres requisitos los hemos tenido en cuenta en el diseño de la arquitectura porque son clave para la integración futura de cualquier conjunto de Bibliotecas Digitales.

A continuación se valora cada una de las aproximaciones según los seis requisitos planteados.

- **Escalabilidad**

Este requisito se cumple en las tres aproximaciones. Tanto los Sistemas de Información Federados que hemos estudiado, como los sistemas que usen los protocolos Z39.50 como OAI-PMH, están diseñados para funcionar de forma eficiente independientemente del número de fuentes de datos (bien estructuradas o bien semiestructuradas, según el caso) que integren.

- **Facilidad para adaptarse a cambios**

De las tres aproximaciones estudiadas, los Sistemas de Información Federados, más concretamente, los Sistemas débilmente acoplados y los Sistemas de Información basados en Mediadores, son los que más facilidad

tienen para adaptarse a los cambios que se produzcan en las fuentes de datos que integran. Tanto los Sistemas de Bases de Datos Federadas como los sistemas construidos sobre los protocolos Z39.50 y OAI-PMH son malas aproximaciones en contextos en los que haya que enfrentarse a una continua evolución en las fuentes de datos. Aunque los protocolos están diseñados para reflejar inmediatamente los cambios que se produzcan, los sistemas construidos sobre ellos tendrán que cambiar su código si quieren explotar dichos cambios.

- **Interfaz de Usuario amigable**

La amigabilidad de la Interfaz de usuario, tan crucial en el caso de acceso a Bibliotecas Digitales, es un aspecto olvidado en las tres aproximaciones que hemos revisado. Recuérdese que el diseño de Interfaces de Usuario es una de las líneas de trabajo de esta tesis, precisamente porque consideramos que la facilidad de uso de la interfaz de una Biblioteca Digital accesible vía Web condiciona el éxito de dicha Biblioteca. Por eso, nos pareció imprescindible que la Arquitectura de Acceso Integrado satisficiera el requisito de proporcionar una Interfaz de Usuario amigable. Precisamente, es la falta de atención a este requisito lo que ha hecho que las tres aproximaciones, que constituyen el estado del arte en integración de fuentes de datos, no nos hayan resultado satisfactorias.

Los protocolos Z39.50 y OAI-PMH no especifican nada sobre la Interfaz de Usuario que será parte del sistema software desarrollado sobre dichos protocolos.

Asimismo, en los Sistemas de Información Federados, simplemente se considera que existirá un lenguaje de consulta al Sistema Federado, exactamente igual que existe para las bases de datos individuales, (por ejemplo SQL) y será necesario construir una Interfaz de Usuario que recoja las peticiones de los usuarios y las exprese en dicho lenguaje de consulta. Por tanto, de nuevo la Interfaz de Usuario es un aspecto externo a la arquitectura de integración.

- **Conservación de Capacidades de Recuperación de Textos**

En los Sistemas de Información Federados estudiados, se conservan las capacidades de Recuperación de Textos de las fuentes sólo en aquellos casos en los que el lenguaje de consulta definido en el sistema sea lo suficientemente expresivo como para mantenerlas.

Por otro lado, los sistemas construidos sobre Z39.50 podrán atender consultas realizadas sobre los conjuntos de atributos definidos en los Puntos de Acceso, pero no contemplan las búsquedas por contenido.

De nuevo, el protocolo OAI-PMH se utiliza para recolectar metadatos, no datos. Por lo tanto, no se contemplan las consultas por el contenido de los documentos de los Proveedores de Datos, sino únicamente por los metadatos de esos documentos.

– **Identificación de las fuentes de datos**

Este requisito es tratado de forma diferente en los Sistemas de Información Federados. Mientras que los Sistemas de Información débilmente aclopados sí identifican la fuente de la que proceden las respuestas, sólo algunos de los Sistemas de Bases de Datos Federadas y algunos de los Sistemas de Información basados en Mediadores lo hacen.

Los sistemas construidos sobre Z39.50, permiten seleccionar el conjunto de bases de datos que consultar, pero no permiten en tiempo de respuesta identificar a qué bases de datos pertenecen las respuestas que se reciben.

Lógicamente, los sistemas basados en el protocolo OAI-PMH sí identifican la fuentes que tienen documentos que corresponden a ciertos metadatos.

– **Conservación de duplicados**

OAI-PMH no devuelve datos, sino metadatos. De los Sistemas de Información Federados, los únicos que no eliminan duplicados son los Sistemas débilmente acoplados. Asimismo, los sistemas construidos sobre el protocolo Z39.50 también realizan un preprocesado de las respuestas obtenidas para eliminar los registros duplicados obtenidos de distintas bases de datos.

Debido a que ninguna de las tres aproximaciones se adaptaba a los requisitos que exigíamos, decidimos diseñar nuestra propia arquitectura en la que estos requisitos se cumpliesen completamente.

4.6 Resumen

En este capítulo se han descrito los Sistemas de Información Federados más destacados de los últimos años. Además, se han descrito los dos protocolos más relevantes para integración de información de Bibliotecas Digitales, el protocolo Z39.50 y el OAI-PMH.

Finalmente, hemos hecho un breve análisis detallando qué características de estos sistemas y protocolos no se adecuan para solucionar los problemas que nos hemos propuesto resolver en este trabajo de tesis.

Capítulo 5

Arquitectura del Sistema de Acceso Integrado a tres Bases de Datos Documentales y Árboles de Conceptos

5.1 Introducción

En este capítulo se describe de forma general la arquitectura, que hemos construido, para el Sistema de Acceso Integrado a nuestras tres bases de datos documentales. La arquitectura se describe desde el punto de vista de las capas por las que está compuesta y de los módulos funcionales de cada una de las capas.

La arquitectura que proponemos tiene como guía central unos ficheros XML que almacenan información clave que rige el funcionamiento de todos los módulos del sistema. Dichos ficheros XML se denominan Árboles de Conceptos y Árboles de Correspondencias. La arquitectura tiene cuatro capas que se comunican entre sí a través de tres lenguajes de intercambio de datos que están definidos también en XML.

En este capítulo se describen también los Árboles de Conceptos y los Árboles de Correspondencias en los que se basa todo el funcionamiento de la arquitectura que hemos diseñado. En la descripción nos preocuparemos de indicar qué información tienen asociada los conceptos de ambos esquemas que hace que no sea necesario reescribir el código de los módulos de software cada vez que se produzcan cambios en las bases de datos o en las necesidades de información de los usuarios.

En los capítulos 6 y 7 se completa la descripción de la arquitectura del sistema, describiendo más detalladamente la implementación de los diferentes módulos de la misma, explicando mediante ejemplos el proceso de consulta

completo desde que es expresada por el usuario, guiado por el sistema, hasta que se presentan las respuestas en la Interfaz de Respuesta.

5.2 Arquitectura General

La arquitectura está formada por las cuatro capas, las cuales se presentan en la Fig. 45 y se describen a continuación.

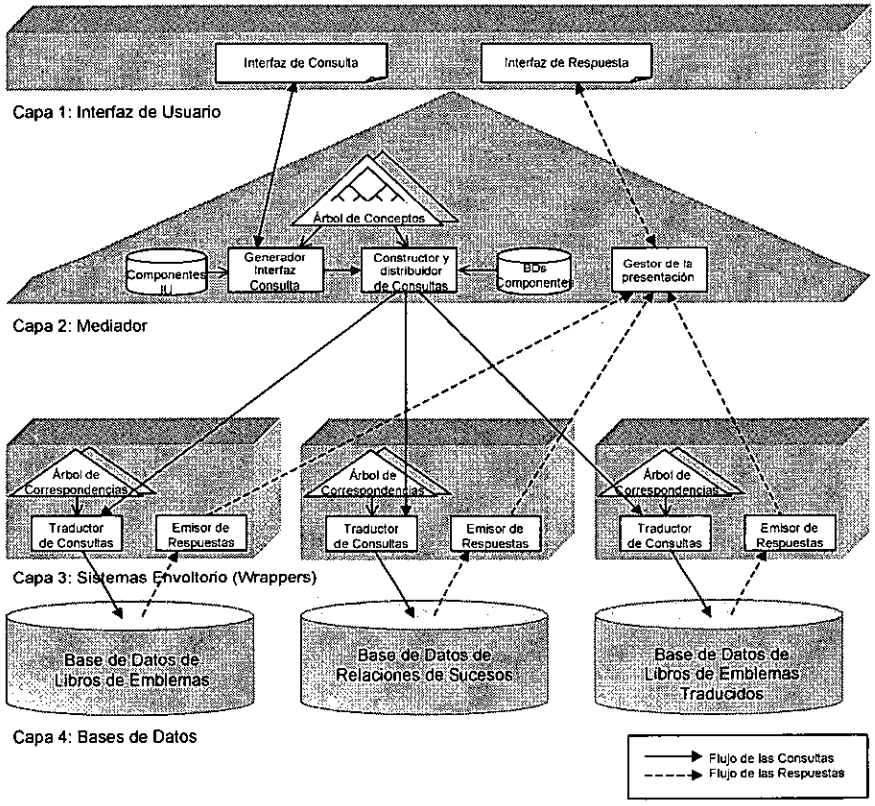


Fig. 45. Sistema de Acceso Integrado a bases de datos documentales

- **Capa 1: Interfaz de Usuario:** El sistema la genera automáticamente cada vez que se accede al sistema.
- **Capa 2: Mediador (Mediator):** Es el núcleo del sistema. Es lo que en bases de datos federadas corresponde al llamado Sistema Federado.
- **Capa 3: Sistemas Envoltorio (Wrappers):** Existe un Sistema Envoltorio por cada base de datos componente. Estos sistemas actúan de puente entre las bases de datos y el Mediador. Su función es "envolver" a la base de

datos haciendo transparentes para el Mediador los detalles de implementación de dicha base de datos.

- **Capa 4: Bases de Datos:** las fuentes que se integran en nuestro sistema son las tres bases de datos documentales, heterogéneas y autónomas descritas. Sin embargo, como se verá en el Capítulo 8, esta arquitectura puede extenderse fácilmente para que pueda integrar el acceso a fuentes de datos semi–estructuradas o no estructuradas.

5.2.1 Capa 1: Interfaz de Usuario

Se trata de una capa conceptual, ya que no existen módulos de software propiamente dichos en la Interfaz de Usuario. Es el Mediador el que la genera cada vez que un usuario se conecta al sistema.

El diseño de la interfaz se basa en una utilización combinada de las tres técnicas de diseño de Interfaces de Usuario descritas en el Capítulo 3. Conseguimos así que el Mediador sea capaz de generar una interfaz realmente fácil de usar, amigable y flexible de manera que el usuario pueda aprovechar las ventajas que ofrece el Sistema de Acceso Integrado que hemos diseñado y construido. Además, la Interfaz de Usuario presenta una parte sencilla que le permite hacer consultas muy simples a usuarios "comunes", y otra parte que, sin incorporar demasiada dificultad, permite realizar consultas mucho más complejas a usuarios "expertos".

Las principales características de la Interfaz de Usuario que es capaz de generar automáticamente nuestro Sistema de Acceso Integrado son las siguientes:

- Proporciona los medios para expresar consultas, no sólo sobre los clásicos datos estructurados, sino que también facilita las consultas sobre el contenido de los documentos. Pensamos que no podía ser de otra manera ya que las tres bases de datos a las que el usuario va a acceder son documentales – almacenando, como ya hemos visto, tanto textos como datos estructurados sobre los documentos (además de tipos de datos multimedia, como imágenes, pero sobre los que no se efectuarán consultas).
- Proporciona los mecanismos necesarios para acceder a las bases de datos tanto a través de consultas globales a todo el conjunto de bases de datos, como consultas específicas a bases de datos de un dominio específico. Este es también un requisito imprescindible debido a que las bases de datos son heterogéneas, en cuanto a los textos y datos que almacenan.
- Es extremadamente intuitiva y fácil de usar. Requisito fundamental debido a que los usuarios del sistema van a ser usuarios web, en su mayoría, no

informáticos, y, los usuarios web no podrán recibir entrenamiento alguno antes de acceder por primera vez al sistema.

- Como ya hemos dicho, la Interfaz de Usuario da la posibilidad de expresar consultas sencillas, a usuarios generales, y consultas más complejas y sofisticadas a usuarios expertos.

5.2.2 Capa 2: Mediador

Esta capa engloba los módulos (y los Almacenes de Datos necesarios para soportar el funcionamiento de dichos módulos) encargados de hacer de nexo entre el usuario y las bases de datos componentes, haciendo transparente para el usuario las diferencias entre las mismas y su dispersión. Como ya sabemos, esta capa también incluye los Árboles de Conceptos, en los cuales aparecen todos los conceptos que existen en las bases de datos y que son relevantes para realizar consultas.

A continuación se describen los elementos (módulos y almacenes) de esta capa:

- Generador de la Interfaz de Consulta: se encarga de generar de forma automática la interfaz Web de consulta, cada vez que un usuario se conecta al sistema;
- Constructor y Distribuidor de Consultas: recibe las restricciones que expresa el usuario y construye el documento XML de la consulta (XML que sigue el DTD del Lenguaje de Consulta del sistema, el cual será descrito en el Capítulo 6). Además, decide qué bases de datos están implicadas en la consulta y la redirige a dichas bases de datos;
- Gestor de la Presentación: construye la interfaz Web que presenta un resumen de las respuestas obtenidas de cada base de datos y gestiona la navegación a través de los datos y documentos, accediendo a la Interfaz de Respuesta implementada para cada base de datos.
- Componentes IU: es almacén en el que están las Frases en Lenguaje Natural Acotado y las Metáforas Cognitivas necesarias para generar la Interfaz deConsulta.
- BDs Componentes: almacena datos (nombre, descripción, parámetros de acceso, etc.) sobre las bases de datos integradas en el sistema.

En los capítulos 6 y 7 se describe la funcionalidad de los módulos de esta capa en detalle y se presentan algunos ejemplos de funcionamiento, respectivamente.

5.2.3 Capa 3: Sistemas Envoltorio

Existe un Sistema Envoltorio por cada base de datos componente. Su función es ocultar al Mediador las diferencias entre las bases de datos. Este sistema gestiona todas las peticiones que el Mediador hace a las bases de datos. Es decir, traduce todas las peticiones que le envía el Mediador al lenguaje propio de la base de datos a la que está asociado, para que puedan ser ejecutadas. Los Sistemas Envoltorio:

- Reciben las consultas del módulo Constructor y Distribuidor de Consultas en formato XML (Lenguaje de Consulta).
- Envían un resumen de las respuestas obtenidas por la consulta al Gestor de la Presentación.

Cada Sistema Envoltorio incluye un Árbol de Correspondencias que contiene la información necesaria para llevar a cabo la traducción de las consultas (expresadas en el Lenguaje de Consulta) al lenguaje propio de la base de datos que tiene asociada.

Los módulos que se encargan de implementar estas funcionalidades son el Traductor de Consultas y el Gestor de Respuestas, y se describen en el Capítulos 6.

5.2.4 Capa 4: Bases de Datos

Las bases de datos forman la cuarta capa de la arquitectura. Las tres bases de datos a las que actualmente se proporciona acceso integrado tienen las siguientes características:

- Heterogéneas, en cuanto a las máquinas que las albergan, al sistema operativo de dichas máquinas y a su estructura. Aunque nuestras bases de datos van a estar soportadas todas ellas por Oracle 9i, los SGBD que las soportasen podrían ser también completamente diferentes entre sí;
- Autónomas, es decir, totalmente independientes en funcionamiento. Esto implica, por un lado, que no es posible cambiar el esquema de ninguna de las bases de datos para adaptarlo al esquema global del Sistema de Acceso Integrado. Por otro lado, es posible que el esquema de las bases de las bases de datos cambie para satisfacer las necesidades de las aplicaciones propietarias. Las tres bases de datos tienen su propia Interfaz de Usuario vía Web. El hecho de que formen parte de nuestro Sistema de Acceso Integrado, no significa que éstas no puedan contestar consultas provenientes de su propia interfaz.
- Documentales, almacenan datos estructurados, textos y páginas digitalizadas (imágenes), por lo que una buena explotación de las bases de

datos documentales debe permitir la aplicación de técnicas de recuperación de textos [7, 8, 9].

5.3 Árboles de Conceptos y Árboles de Correspondencias

De la descripción (general) que hemos hecho de la arquitectura se deduce que los Árboles de Conceptos y los Árboles de Correspondencias son estructuras que juegan un importante papel en el sistema.

Los Árboles de Conceptos almacenan los conceptos presentes en las bases de datos componentes, aunque únicamente aquellos que son interesantes para realizar consultas. Lógicamente, los Árboles de Conceptos se sitúan en la capa del Mediador, como puede verse en la Fig. 46.

Para poder ejecutar en las bases de datos las consultas formuladas en términos de los conceptos del Árbol de Conceptos, existe para cada base de datos un Árbol de Correspondencias. Los Árboles de Correspondencias están situados en los Sistemas Envoltorio, como puede verse en la Fig. 46.

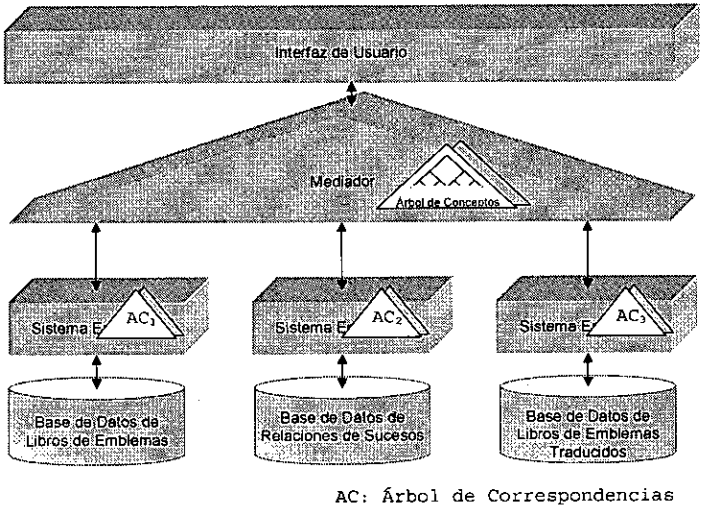


Fig. 46. Ubicación de los Árboles de Conceptos y los Árboles de Correspondencias

Sin embargo, en nuestro sistema utilizamos los Árboles de Conceptos para algo más que para representar el esquema global del sistema. Hemos incluido en los Árboles de Conceptos información útil para generar la Interfaz de Consulta, cada vez que un usuario se conecte, y para saber qué bases de datos están implicadas en cada consulta para poder así enviar las consultas

únicamente a esas bases de datos, en vez de a todas las que formen parte del Sistema de Acceso Integrado.

Por otro lado, los Árboles de Correspondencias son estructuras que almacenan, para cada uno de los conceptos del Árbol de Conceptos, su correspondiente (tabla, atributo, etc.) en la base de datos a la que están asociados. Esta información servirá para realizar una traducción eficiente de la consulta en XML al lenguaje de la base de datos.

Tanto los Árboles de Conceptos como los Árboles de Correspondencias de nuestro sistema están definidos en XML [62]. XML permite definir estándares para formatos de datos y metadatos y es un lenguaje universal para estructurar documentos y datos. Hemos escogido XML porque este lenguaje permite definir formatos para intercambio de datos y metadatos, legibles por humanos y fácilmente parseables por máquinas.

A continuación se describen las características comunes de los Árboles de Conceptos y los Árboles de Correspondencias y, en los siguientes apartados, la información que almacenan y el propósito de cada uno de ellos.

La especificación formal de los Árboles de Conceptos y los Árboles de Correspondencias, y sus DTDs se describen en los apéndices I y II, respectivamente. En estos apéndices aparecen también los documentos XML de los Árboles de Conceptos y de los Árboles de Correspondencias de nuestro sistema en su versión actual.

5.3.1 Descripción Árboles

Tanto los Árboles de Conceptos como los Árboles de Correspondencias tienen características en común que son las que vamos a describir en este apartado, refiriéndonos a ambos como Árboles.

Un Árbol en nuestro sistema es un conjunto de conceptos y relaciones (entre ellos) que describen la información almacenada en las tres bases de datos componentes, que son relevantes para realizar consultas sobre ellas.

Los Árboles son estructuras, en donde los nodos son conceptos. Los conceptos tienen atributos, es decir, propiedades atómicas de dichos conceptos. Tanto los conceptos como sus atributos, son nociones usadas en el mundo real, en los dominios de las bases de datos. Por ejemplo, un posible concepto de un Árbol podría ser "Autor". Entre los posibles atributos de este concepto podrían estar: "Nombre", "Lugar de Nacimiento" o "Año de Nacimiento".

En Fig. 47 aparece la representación gráfica de un concepto con sus atributos. De aquí en adelante, hablaremos de "términos" cuando nos estemos refiriendo tanto a conceptos como a atributos de un Árbol.

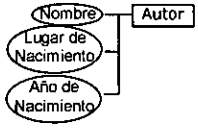


Fig. 47. Un concepto y sus atributos

Las ramas de un Árbol representan las relaciones entre los conceptos. En nuestro dominio de interés (bases de datos documentales) hemos considerado relevantes dos tipos de relaciones: Relaciones de Generalización / Especialización y Relaciones de Descripción.

Relación de Descripción

Un concepto se describe, no sólo mediante sus atributos, sino también a través de otros conceptos. Esta Relación de Descripción entre conceptos se representa con la etiqueta "has".

Por ejemplo, una obra no sólo se describe por sus atributos, tales como "Título", sino también por su(s) edición(es). Pero "Edición" es, a su vez, un concepto que tiene atributos, como "Año", "Editor" o "Impresor". Por lo tanto, el concepto "Obra" y el concepto "Edición" están relacionados por una Relación de Descripción. En la Fig. 48 se muestra la representación gráfica de este tipo de relación.

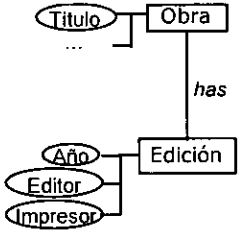


Fig. 48. Relación de Descripción

Relación de Generalización / Especialización

Una relación de Generalización / Especialización se define entre un concepto, que llamamos general, y un conjunto de uno o más conceptos, que llamamos especializados.

Decimos que existe una relación de Generalización / Especialización entre un concepto general y un conjunto de conceptos especializados cuando estos últimos se pueden definir a partir de una propiedad distintiva del concepto general. O, lo que es equivalente, cuando el concepto general se genera para englobar las propiedades comunes de un conjunto de conceptos (especializados).

Hay dos razones principales para incluir relaciones de Generalización / Especialización en un Árbol. La primera es que pueden existir ciertos atributos aplicables únicamente a ciertos tipos de instancias del concepto. La segunda razón es que cada especialización del concepto puede establecer una Relación de Descripción con conceptos distintos.

Por ejemplo, el concepto "Edición", según el valor de su atributo "Soporte" se especializaría en "Edición en papel" y en "Edición digital". La "Edición en papel" tendría atributos como "ISBN" y la Relación de Descripción con el concepto "Ejemplar" y la "Edición digital" tendría atributos como "Formato" y "URL".

Las relaciones de Generalización / Especialización se representan, como en el ejemplo de la Fig. 49, etiquetadas con la cadena "is a".

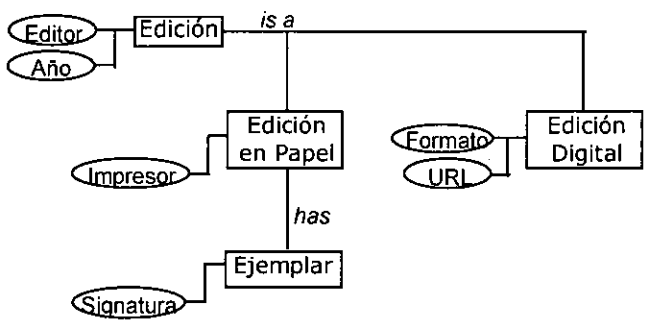


Fig. 49. Relación de Generalización / Especialización

Las relaciones de Generalización / Especialización de nuestra sistema son:

- Disjuntas: una instancia del concepto general sólo se especializa en uno de los conceptos especializados.
- Totales o completas: todas las instancias del concepto general se especializan en alguno de los conceptos especializados.

Por ejemplo, una obra en nuestro sistema es forzosamente, o bien, una "Relación de Sucesos", o un "Libro de Emblemas" o un "Libro de Emblemas Traducido", y solamente de uno de estos tipos.

Los conceptos se organizan en el árbol de modo que pueden dar lugar a subárboles con conceptos que requieran diferentes grados de conocimiento por parte de los usuarios. En concreto, en los Árboles usados en nuestro Sistema de Acceso Integrado a las Bibliotecas Digitales del Siglo de Oro se distinguen dos partes que denominamos Parte General y Parte Experta.

- **Parte General:**

  La Parte General es un subárbol en el que los conceptos son generales, y comprensibles para cualquier usuario, aunque no esté especializado en ningún dominio concreto. Por ejemplo, no necesitamos ser especialistas en Literatura Emblemática para saber lo que es un "Libro", un "Título" o un "Autor". La Parte General la representamos en el subárbol izquierdo de los Árboles, como se puede ver en la Fig. 52.

- **Parte Experta:**

  En esta parte están englobados todos los conceptos propios de dominios específicos y especializados. Para entender los conceptos de esta parte de un Árbol, es necesario tener cierto nivel de conocimiento sobre el dominio al que pertenezcan. En la Fig. 51 se muestra el Árbol de Conceptos de nuestro sistema donde se puede ver la Parte Experta formada por tres subárboles. Esta Parte Experta contiene conceptos especializados como "Mote" o "Epigrama" que son sólo familiares a especialistas en Literatura Emblemática.

El que los conceptos estén organizados de esta forma permitirá que en la Interfaz de Consulta se puedan expresar consultas sencillas y generales (sobre conceptos de la Parte General) y consultas más específicas y complejas (sobre la Parte Experta). De esta forma, la interfaz se adapta al nivel de conocimiento que los usuarios tengan sobre los dominios a los que pertenezcan las bases de datos componentes.

Hay que recordar que los Árboles sólo contienen aquellos conceptos y atributos de las bases de datos que son útiles para las búsquedas. Por ejemplo, en la Biblioteca Digital de Relaciones de Sucesos, la tabla Biblioteca, tiene atributos como "Página Web, "Teléfono" y "Dirección". Sin embargo, en el Árbol de Conceptos (Fig. 51), dichos atributos no aparecen asociados al concepto "Biblioteca" por no ser relevantes para las búsquedas.

Por otro lado, los Árboles pueden presentar redundancia. En la Fig. 51, se puede ver el concepto "Edición" repetido en los subárboles de "Libros de Emblemas" y "Relaciones de Sucesos". Mantenemos esta redundancia en los Árboles, porque las desventajas que siempre provoca el tener redundancia son compensadas por la simplicidad de los algoritmos que usan los Árboles para

generar la Interfaz de Consulta y para analizar y redirigir las consultas, como veremos en los Capítulos 6 y 7.

5.4 Árboles de Conceptos

Como ya hemos dicho, los Árboles de Conceptos están situados en el Mediador. Cada Árbol tiene una raíz que corresponde a un concepto sobre el que el usuario está buscando información.

Por ejemplo, los dos Árboles de Conceptos de un Sistema de Acceso Integrado podrían ser los presentados en la Fig. 50. Quizá pudieran generarse automáticamente a partir de una ontología que represente la totalidad del conocimiento.

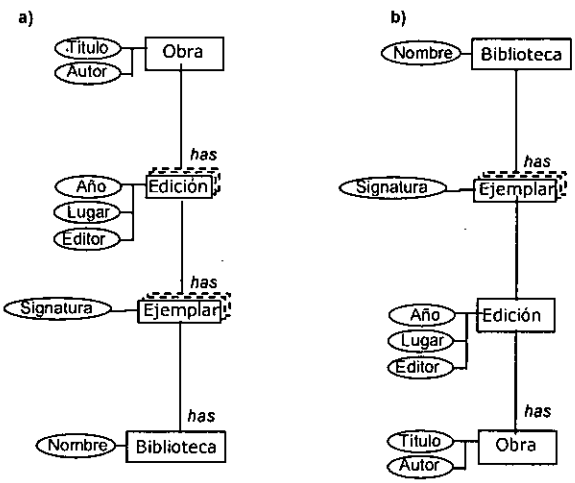


Fig. 50. Dos Árboles de Conceptos

La interfaz de este Sistema de Acceso Integrado empieza por preguntar al usuario cuál de los conceptos que están en la raíz de los Árboles de Conceptos que tiene, representa la entidad sobre la que está buscando información. Una vez elegido el Árbol de Conceptos, sigue el proceso que se verá en el Capítulo 6.

En un Árbol de Conceptos, las relaciones, los conceptos y los atributos tienen asociada información relativa, tanto a las bases de datos en las que existen, como cierta información que permite construir la Interfaz de Consulta para expresar condiciones sobre dichos conceptos y atributos.

5.4.1 Información para construir la Interfaz de Consulta

Una consulta del usuario puede definirse como una descripción de las propiedades que deben tener los objetos que se obtengan en la respuesta. Un usuario puede describir los objetos que está buscando describiendo las características de los mismos. Dichas características o propiedades que el usuario puede "describir" no son más que los atributos asociados a dicho objeto en el Árbol de Conceptos del sistema. Por lo tanto, podemos decir que las consultas permitidas por el sistema son conjuntos de condiciones sobre los atributos del Árbol de Conceptos.

El sistema, por tanto, ofrece un entorno (Interfaz de Consulta) en el que el usuario puede, por un lado, seleccionar los atributos del Árbol de Conceptos sobre los que tiene interés en expresar restricciones, es decir, puede recorrer el Árbol de Conceptos y seleccionar aquellos atributos que le interesan; y, por otro lado, en el que puede, finalmente, expresar las restricciones que desee sobre los atributos que ha seleccionado. Además, cuando los valores que puede tomar un atributo son un conjunto pequeño, dichos valores se almacenan en el Árbol de Conceptos asociados a dicho atributo, facilitando así que el usuario sólo escoja valores válidos para restringir dichos atributos.

En nuestro sistema el módulo encargado de presentar la Interfaz de Consulta, no está programado "ad hoc" para un Árbol de Conceptos concreto. Por el contrario, se trata de un módulo capaz de leer cualquier Árbol de Conceptos (fichero XML adaptado al DTD que hemos definido y presentamos en el Anexo I). Después de leer el Árbol de Conceptos, este módulo es capaz de generar, de forma automática, la Interfaz de Consulta adecuada para dicho Árbol. Esto se ha conseguido asociando a los atributos del Árbol de Conceptos información que permite al sistema construir automáticamente la Interfaz de Consulta desde la que el usuario podrá expresar condiciones sobre dichos atributos.

La técnica del Lenguaje Natural Acotado y las Metáforas Cognitivas (incluyendo cuando es útil la técnica de la Aproximación Navegacional) son las técnicas que usamos en nuestra arquitectura para construir la Interfaz de Consulta. En el Capítulo 3, vimos con detalle en qué consisten las tres técnicas que se usan para construir la Interfaz de Consulta. En este apartado, únicamente interesa establecer que todos los atributos del Árbol de Conceptos, y, opcionalmente, los conceptos y las Relaciones de Generalización / Especialización, tienen asociado el identificador de la frase en Lenguaje Natural Acotado o el identificador de la Metáfora Cognitiva que permitirá construir la interfaz para expresar condiciones sobre dicho atributo.

Más adelante, en el Capítulo 6, se explica con más detalle cómo el módulo Generador de la Interfaz de consulta usa esta información.

5.4.2 Información relativa a las bases de datos

Los Árboles de Conceptos contiene los conceptos, útiles para consulta, que existen en las bases de datos. Sin embargo, puede ocurrir que algunos de los conceptos y/o atributos del Árbol de Conceptos no existan en alguna de las bases de datos componentes. Por lo tanto, a efectos de saber a qué bases de datos redirigir las consultas expresadas sobre un cierto conjunto de conceptos y atributos, es necesario saber qué atributos existen en cada base de datos componente.

En nuestros Árboles de Conceptos, cada atributo tiene asociada una lista de identificadores. Se trata de los identificadores de las bases de datos en las que de dicho atributo existe. De esta manera, una consulta expresada sobre un conjunto de atributos será redirigida únicamente a las bases de datos cuyos identificadores estén asociados a dichos atributos. Es decir, a aquellas bases de datos que realmente pueden contestar la consulta (aunque sea parcialmente).

En el capítulo 6 se explica detalladamente el protocolo que se sigue para la distribución de las consultas del usuario a las bases de datos componentes. En este apartado, simplemente queremos establecer que los atributos de los Árboles de Conceptos tienen asociada una lista con los identificadores de las bases de datos en las que existen.

5.4.3 Árbol de Conceptos del Sistema de Acceso Integrado a las tres bases de datos del Siglo de Oro

Nuestro sistema tiene un único Árbol de Conceptos por estar orientado únicamente a la búsqueda de documentos. Quizás en el futuro se pueda crear otro Árbol de Conceptos para buscar bibliotecas que tengan ciertos fondos.

El Árbol de Conceptos de nuestro sistema está formado por un único concepto en la Parte General y tres subárboles en la Parte Experta.

Aunque las Relaciones de Sucesos y los Libros de Emblemas Hispánicos y Libros de Emblemas Traducidos son, como ya se ha visto en el capítulo 2, documentos de los siglos XVI-XVIII que reflejan la cultura de aquella época, estos documentos son muy distintos entre sí. De hecho, el único concepto común, y útil para la realización de las búsquedas, que se extrae de estos tres dominios es el concepto "Obra" por lo que la Parte General del Árbol de Conceptos está formada únicamente por dicho concepto con sus atributos. La Parte General del Árbol de Conceptos se muestra en la Fig. 52 y en el subárbol izquierdo de la Fig. 51, que muestra el Árbol de Conceptos completo de nuestro Sistema de Acceso Integrado.

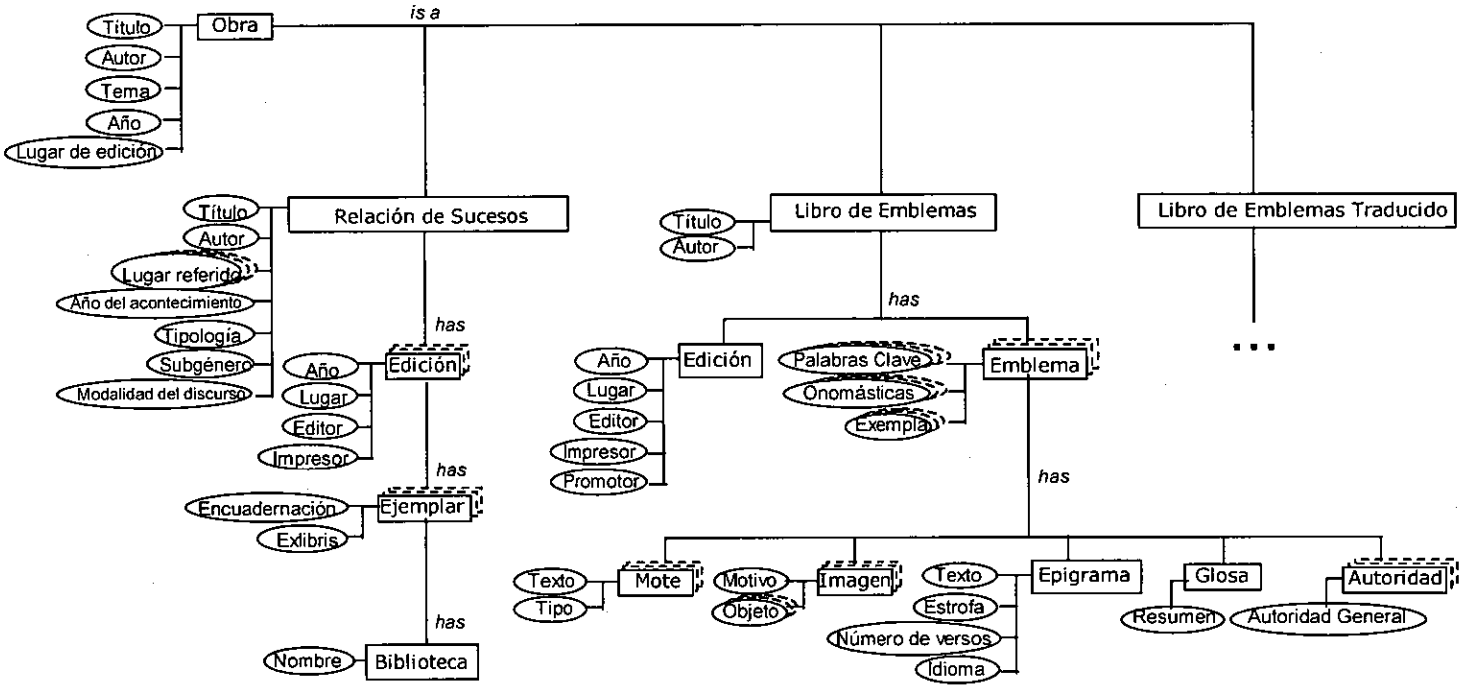


Fig. 51. Árbol de Conceptos del Sistema de Acceso Integrado

En cuanto a la Parte Experta del Árbol de Conceptos, está formada por tres subárboles: El subárbol para Libros de Emblemas, el de Libros de Emblemas Traducidos y el subárbol para Relaciones de Sucesos, teniendo asociada una base de datos a cada dominio. La Parte Experta del Árbol de Conceptos puede verse en la Fig. 51.

Para una mayor claridad y comprensión de los Árboles de Conceptos hemos omitido la información, que tienen asociada los términos, relativa a la generación de la Interfaz de Consulta y al envío de las consultas a las bases de datos implicadas. Sin embargo, como ejemplo, en la Fig. 52 se puede ver dicha información para el concepto "Obra" de la Parte General de nuestro Árbol de Conceptos y para uno de sus atributos "Lugar de Edición".

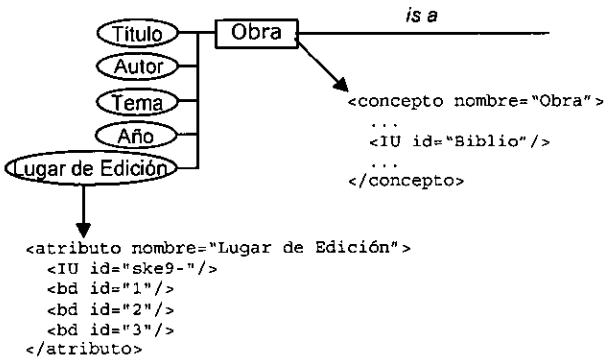


Fig. 52. Parte General del Árbol de Conceptos del Sistema

El concepto "Obra" tiene asociada la Metáfora Cognitiva `"Biblio"`, la cual se presenta en la Fig. 53. Con esta Metáfora, que, como se ve, permite restringir el atributo "Tipo de Obra", se generará la interfaz Web en la que el usuario podrá elegir en qué tipo de obras está interesado sin más que pinchar en la estantería que corresponda.

El atributo "Lugar de Edición" lleva asociado el Identificador del Esqueleto de Frase `ske9` que se muestra en la Fig. 54. Este Esqueleto servirá para construir la frase en Lenguaje Natural Acotado que se presentará en la Interfaz de Consulta para permitir expresar restricciones sobre este atributo.

Por otro lado, dicho atributo "Lugar de Edición" existe en nuestras tres bases de datos. Los identificadores `1`, `2` y `3` (Fig. 52) se refieren a cada una de las tres bases de datos integradas y sirven para buscar en el almacén de bases de datos del Mediador (BDs Componentes), todos aquellos parámetros necesarios para acceder al Sistema Envoltorio de cada una de ellas, y enviarle la consulta del usuario en XML. Se explica más adelante, en el Capítulo 6, el

funcionamiento del módulo Constructor y Distribuidor de Consultas, que es el encargado de realizar esta tarea.

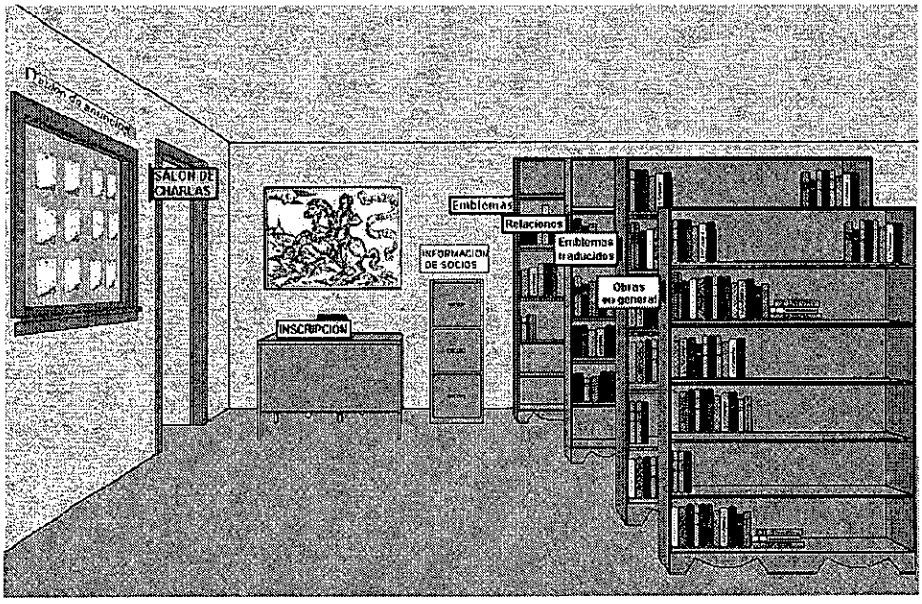


Fig. 53. Metáfora Cognitiva y Aproximación Navegacional

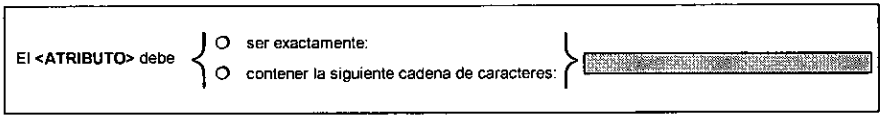


Fig. 54. Esqueleto asociado al atributo "Lugar de Edición"

5.5 Árboles de Correspondencias (Mappings)

Los Árboles de Correspondencias tienen una estructura idéntica a la de los Árboles de Conceptos. Es una estructura en forma de árbol en la que los nodos son conceptos, con propiedades que llamamos atributos, y las ramas relaciones. Existen los mismos tipos de relaciones y la semántica de los conceptos y los atributos es la misma. Sin embargo, el propósito de los Árboles de Conceptos y los Árboles de Correspondencias son muy distintos. Los Árboles de Correspondencias servirán, como hemos dicho, para traducir las consultas que plantean los usuarios sobre los atributos del Árbol de Conceptos (y que el sistema almacena en XML) al lenguaje de la base de datos a la que está asociado dicho Árbol de Correspondencias.

En los casos, como el nuestro, en los que las bases de datos componentes sean bases de datos relacionales, el lenguaje al que se deberá traducir la consulta del usuario es SQL.

Durante la traducción, para cada concepto que aparezca en el documento XML, se lee del Árbol de Correspondencias la Información de Correspondencia que tiene asociado dicho concepto y se añade a la sentencia SQL final. Asimismo, para cada atributo que exista en el documento XML de la consulta, se lee la Información de Correspondencia de dicho atributo y, antes de completar la sentencia SQL final con la nueva condición, se sustituye las etiquetas entre # por los valores concretos que haya establecido el usuario y que están recogidos en el documento XML. En nuestro sistema contemplamos también los casos en los que los valores de algún atributo, no estén almacenados tal cual en la base de datos, sino que se almacene su código correspondiente. En este caso, la Información de Correspondencia del atributo también incluye la tabla de traducción de cada posible valor al código que se almacena en la base de datos.

5.5.1 Árboles de Correspondencias del Sistema de Acceso Integrado a las bases de datos del Siglo de Oro

En nuestro sistema existen tres Árboles de Correspondencias uno para cada una de las bases de datos que están integradas en el mismo.

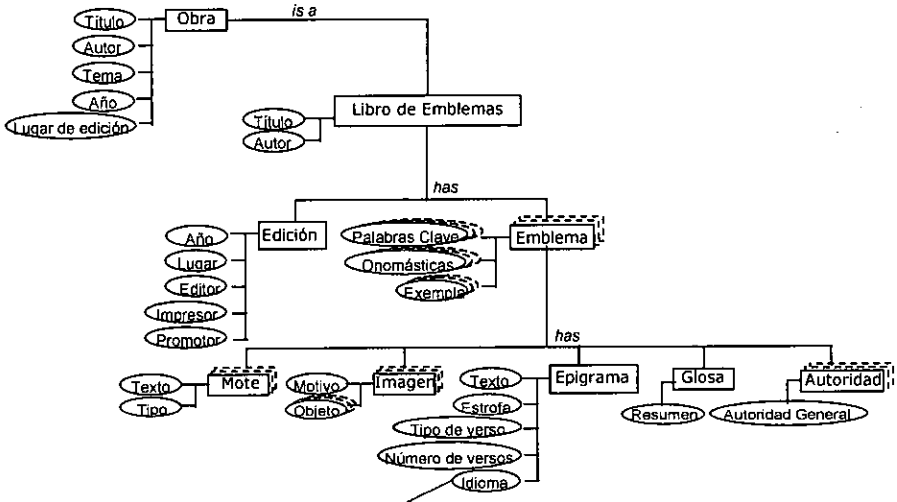
En el Apéndice III se presentan los documentos XML que almacenan los Árboles de Correspondencias de nuestras bases de datos. A continuación se explican dichos Árboles de Correspondencias.

Base de Datos Libros de Emblemas

El Árbol de Correspondencias asociado a la base de datos de Libros de Emblemas se presenta en la Fig. 55. En esta figura no aparece reflejada toda la Información de Correspondencias, únicamente se muestra la correspondiente al atributo "Idioma" del concepto "Epigrama".

Durante la traducción de una condición expresada sobre el atributo "Idioma" se incorporará a la cláusula `where` de la consulta `SQL` final lo que en la Fig. 55 aparece entre las etiquetas `<fijo>`. Previamente, se sustituiría el valor concreto que tenga ese atributo en la consulta en XML por el código que se almacena realmente en la base de datos y se sustituiría por el parámetro `#valor#`.

En el Capítulo 6 se explica en detalle el algoritmo de la traducción de las consultas de XML a SQL y en el Capítulo 7 se presentan un ejemplo de traducción.




```
<correspondencia>
   <where>
      <fijo> em.idioma_epi = '#valor#' </fijo>
   </where>
</correspondencia>
<valor>Español</valor><traduccion>Esp</traduccion>
<valor>Flamenco</valor><traduccion>Fla</traduccion>
<valor>Francés</valor><traduccion>Fra</traduccion>
<valor>Griego</valor><traduccion>Gri</traduccion>
<valor>Inglés</valor><traduccion>Ing</traduccion>
<valor>Italiano</valor><traduccion>Ita</traduccion>
<valor>Latín</valor><traduccion>Lat</traduccion>
<valor>Políglota</valor><traduccion>Pol</traduccion>
```

Fig. 55. Árbol de Correspondencias para Libros de Emblemas

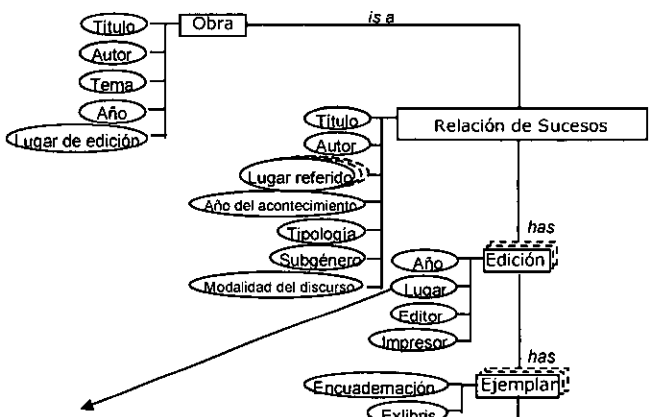
Base de Datos de Libros de Emblemas Traducidos

En principio, la estructura del Árbol de Correspondencias asociado a la base de datos Libros de Emblemas Traducidos, será parecida al de la base de datos Libros de Emblemas, pero aún esperamos peticiones de cambios en el modelo de datos, por parte de los filólogos.

Base de Datos de Relaciones de Sucesos

En la Fig. 56 se presenta el Árbol de Correspondencias de la base de datos de Relaciones de Sucesos. Igual que en el caso anterior, sólo presentamos la

Información de Correspondencia para el atributo "Lugar". En este caso, no es necesario traducir los valores que introduzca el usuario en su consulta.




```
<atributo nombre="Lugar">
  <correspondencia>
    <where>
      <fijo> ed.lugar like '%#valor#%' </fijo>
    </where>
  </correspondencia>
 </atributo>
```

Fig. 56. Árbol de Correspondencias para Relaciones de Sucesos

5.6 Lenguajes de Comunicación entre Capas

Como ya se ha comentado, se han definido dos lenguajes para establecer la comunicación entre las capas de la arquitectura. Estos lenguajes son:

- *Lenguaje de Consulta:* define el formato de las consultas del usuario.
- *Lenguaje de Respuestas*: proporciona un formato para enviar el número de respuestas obtenidas en una consulta.

Estos dos lenguajes de intercambio de datos hacen independiente el Mediador, no sólo de las diferencias entre los lenguajes de las bases de datos sino también entre los tipos de datos de los documentos almacenados en la base de datos.

Al igual que los Árboles de Conceptos y Los Árboles de Correspondencias, estos lenguajes se definen en XML. En el capítulo 6 se define el Lenguaje de Consultas, y en el Apéndice IV se presenta su DTD.

5.7 Resumen

En este capítulo hemos introducido la arquitectura de nuestro sistema, explicando de forma general su funcionamiento, e introduciendo los dos capítulos (6 y 7) en los que se explican en profundidad los principales módulos de la misma.

Hemos descrito también las estructuras en las que se basa nuestra nuestro Sistema de Acceso Integrado, los Árboles de Conceptos y los Árboles de Correspondencias y hemos introducido los dos lenguajes de intercambio de datos que se necesitan para comunicar las diferentes capas de la arquitectura.

Capítulo 6

Implementación del sistema

6.1 Introducción

Una vez descrito el funcionamiento general de la arquitectura y explicados en detalle los Árboles de Conceptos y los Árboles de Correspondencias, abordamos en este capítulo la descripción detallada de cada uno de los módulos de la arquitectura. Se explicará el funcionamiento de cada módulo por orden de intervención en el proceso de consulta.

En este capítulo se describe también el Lenguaje de Consulta que se ha definido para comunicar el Mediador con los Sistemas Envoltorio de las bases de datos.

6.2 Generador de la Interfaz de Consulta

Este módulo genera dinámicamente la Interfaz de Consulta del sistema guiado por el Árbol de Conceptos, y usando la información asociada a los conceptos y atributos. Además, este módulo recoge las restricciones del usuario.

Para la interfaz se usan, como hemos visto en el capítulo 3, tres técnicas de diseño de Interfaces de Usuario: la técnica del Lenguaje Natural Acotado, la técnica de las Analogías o las Metáforas Cognitivas y la Aproximación Navegacional. Por otro lado, las consultas que se pueden plantear al sistema son condiciones sobre los atributos de los conceptos de los Árboles de Conceptos. Por lo tanto, el Generador de la Interfaz de Consulta genera la Interfaz de Consulta guiado por los Árboles de Conceptos y usando la información disponible en los almacenes de Esqueletos de Frases y Metáforas Cognitivas (Componentes IU). Recordemos que cada atributo de los Árboles de Conceptos tiene asociado el identificador de un elemento de este almacén.

En la Fig. 57 se presenta la parte de la arquitectura encargada de generar la Interfaz de Consulta y recoger las restricciones formuladas en la interfaz.

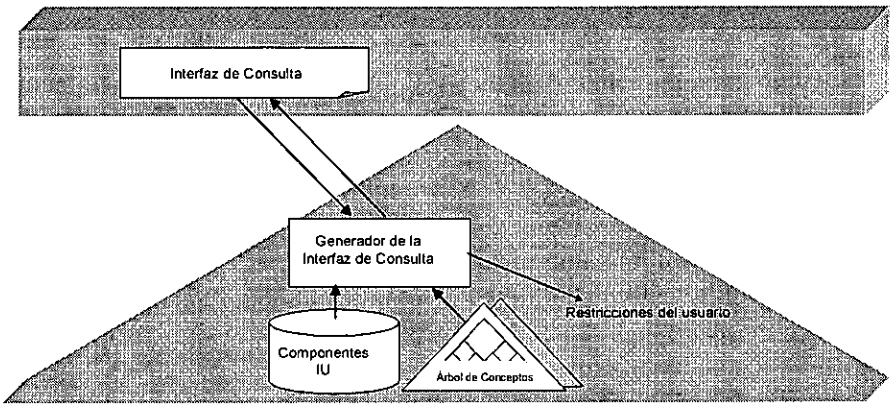


Fig. 57. Módulos implicados en la Generación de la Interfaz de Consulta

El Generador de la Interfaz de Consulta proporciona los mecanismos necesarios para que el usuario pueda:

- Recorrer el Árbol de Conceptos seleccionando los atributos sobre los que quiere expresar condiciones.
- Expresar, finalmente, las restricciones sobre los atributos seleccionados. Además, el Generador de la Interfaz de Consulta le pasa estas restricciones al Constructor y Distribuidor de Consultas, que será el que realmente cree el documento en XML que representará la consulta del usuario

A continuación veremos cómo se implementan en el sistema las funcionalidades anteriores.

Expresar condiciones sobre los atributos

Todos los atributos de los Árboles de Conceptos tienen asociada una frase en Lenguaje Natural Acotado o una Metáfora Cognitiva para facilitar al usuario expresar restricciones sobre dicho atributo. En el Capítulo 3 mostramos cómo siempre es posible construir una frase en Lenguaje Natural Acotado para facilitar expresar restricciones sobre cualquier atributo.

La Frase o la Metáfora Cognitiva se presentará en la Interfaz de Consulta en el momento en que el usuario quiera expresar alguna condición sobre dicho atributo.

Recorrer el Árbol de Conceptos

Además de los Esqueletos de Frase o Metáforas Cognitivas necesarios para permitir al usuario expresar restricciones sobre los atributos del Árbol de Conceptos, se necesitan otros dos Esqueletos de Frase más para permitir que el usuario recorra el Árbol de Conceptos seleccionando aquellos atributos sobre los que desea expresar restricciones, y construya así la consulta completa. Estos dos Esqueletos dan lugar a dos frases que llamamos *Frase de Especialización*, que se usa con las Relaciones de Generalización / Especialización y la *Frase de Descripción*. Como en el caso de los Esqueletos de Frase para expresar restricciones sobre los atributos, estas frases pueden ser sustituidas por Metáforas Cognitivas.

– Esqueleto de Frase de Especialización

Este Esqueleto de Frase, representado en la Fig. 58, se usa con los conceptos que tienen relaciones de tipo *Generalización/Especialización* con otros conceptos. Con esta frase el usuario podrá decidir si está interesado en el concepto general o en alguna de sus especializaciones.

Por ejemplo, el concepto "Edición" del Árbol de la Fig. 49, es el concepto general en una relación de Generalización/Especialización. La frase que se le presentaría al usuario para el concepto "Edición" sería la presentada en la Fig. 59. Completando esta frase el usuario podrá especificar si está interesado en "Ediciones de cualquier tipo", en "Ediciones digitales" o en "Ediciones en papel".

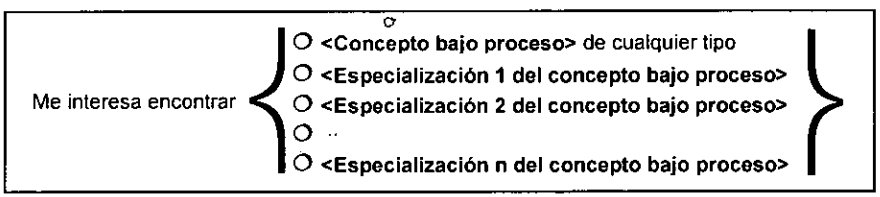


Fig. 58. Esqueleto Frase de Especialización

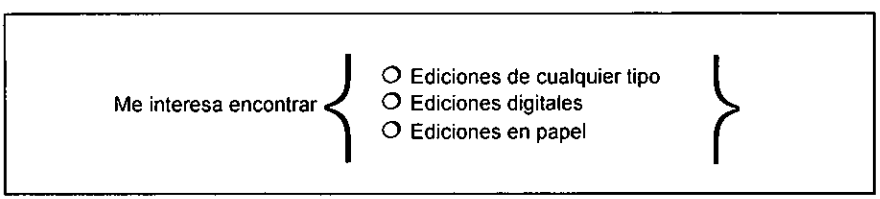


Fig. 59. Ejemplo de Frase de Especialización

De todas formas, aunque siempre se puede generar una Frase de Especialización, es posible también asociar una Metáfora Cognitiva que

permita elegir entre el concepto general y alguna de las especializaciones. Por ejemplo, se podría presentar una estantería con libros, que representase las ediciones en papel, y una pantalla de ordenador, para representar las ediciones electrónicas con URL, y permitir que se marcasen las dos imágenes o sólo una de ellas. Estas imágenes sustituirían a la frase en Lenguaje Natural Acotado de la Fig. 59.

– **Esqueleto de Frase de Descripción**

Se usa para permitir que el usuario elija qué atributos y qué conceptos, de aquellos que describen un concepto, le interesan para expresar restricciones.

Este esqueleto se presenta en la Fig. 60. La lista comienza con los atributos del concepto que se está procesando y va seguida por la lista de conceptos relacionados con el concepto actual a través de una Relación de Descripción. El usuario puede elegir tantos atributos o conceptos como desee.

Por ejemplo para el concepto "Obra" del Árbol de Conceptos de nuestro sistema (Fig. 51), la Frase de Descripción correspondiente sería la que se presenta en la Fig. 61.

Fig. 60. Esqueleto de Frase de Descripción

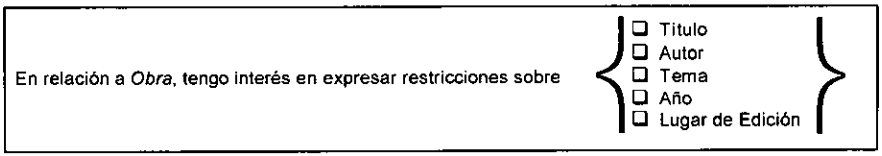


Fig. 61. Ejemplo de Frase de Descripción

6.2.1 Algoritmo del Generador de la Interfaz de Consulta

Aunque es el usuario el que expresa la consulta, el sistema le ayuda en esta tarea poniendo a su disposición las pantallas apropiadas, creadas como hemos visto a partir de las Metáforas Cognitivas y los Esqueletos de Frase asociados a los atributos y conceptos del Árbol de Conceptos del sistema. Una vez establecido cómo son dichas pantallas que el sistema le presentará al usuario para que pueda expresar restricciones sobre los atributos del Árbol de Conceptos y para que pueda también navegar a través de él seleccionando los

atributos que está interesado en restringir, vamos a describir el proceso global de consulta.

Como hemos visto, en la Fig. 57 se muestran los módulos de la arquitectura implicados en la generación de la Interfaz de Consulta. El principal módulo encargado de la gestión de la Interfaz de Consulta es el *Generador de la Interfaz de Consulta*. La ejecución de este módulo depende completamente de la información almacenada en el Árbol de Conceptos. Básicamente, su funcionamiento consiste en leer el Árbol comenzando por el concepto raíz, y presentarle al usuario la interfaz (que en unos casos contiene una Metáfora Cognitiva y en otros casos una frase en Lenguaje Natural Acotado) que está almacenada en el Almacén Componentes IU y que este módulo puede localizar a través del identificador asociado a cada atributo, a los conceptos y relaciones de Especialización / Generalización del Árbol de Conceptos.

El algoritmo que sigue el módulo Generador de la Interfaz de Consulta se muestra en la Fig. 62. El `Algoritmo de la Interfaz de Consulta (AIC)` guía al usuario a través de los conceptos y atributos del Árbol de Conceptos para que pueda formular su consulta completa.

La Interfaz de Consulta presenta el Árbol de Conceptos. En este árbol, se marcarán los conceptos y atributos sobre los que se ha expresado alguna condición. Además, para que el usuario pueda ver la consulta que va construyendo, una ventana diferente le mostrará las frases en LNA seleccionadas y completadas.

En la Fig. 63, se muestra un ejemplo de pantalla de consulta en una de las fases iniciales de diseño. En esta pantalla se pueden distinguir las tres áreas de las que hablábamos en el párrafo anterior.

El primer parámetro de entrada del Algoritmo de la Interfaz de Consulta es el concepto raíz del Árbol de Conceptos. El funcionamiento del algoritmo es el siguiente:

En primer lugar comprueba si el `concepto-actual` tiene una relación de Generalización/Especialización. En este caso, el algoritmo instancia el Esqueleto de Frase de Especialización con el `concepto-actual` y sus especializaciones, o presenta la Metáfora Cognitiva que tenga asociada dicha relación, captura la opción que elige el usuario y se llama a sí mismo tomando como parámetro de entrada el concepto seleccionado por el usuario.

Si el concepto elegido no tiene especialización (o ya ha sido tratada), el algoritmo instancia el Esqueleto de Frase de Descripción con la lista de atributos del `concepto-actual` y con la lista de conceptos con los que mantiene una relación de Descripción. Con esta frase, o con la Metáfora

Cognitiva que tenga asociada el concepto, el usuario puede seleccionar tantos atributos/conceptos como necesite. Con los atributos/conceptos seleccionados se crean dos listas. Una de ellas, `ListaAtributos`, contiene los atributos, del `concepto-actual`, que ha seleccionado el usuario. La otra, `ListaConceptos`, contiene, de los conceptos relacionados con el `concepto-actual` a través una Relación de Descripción, aquellos que ha seleccionado el usuario.

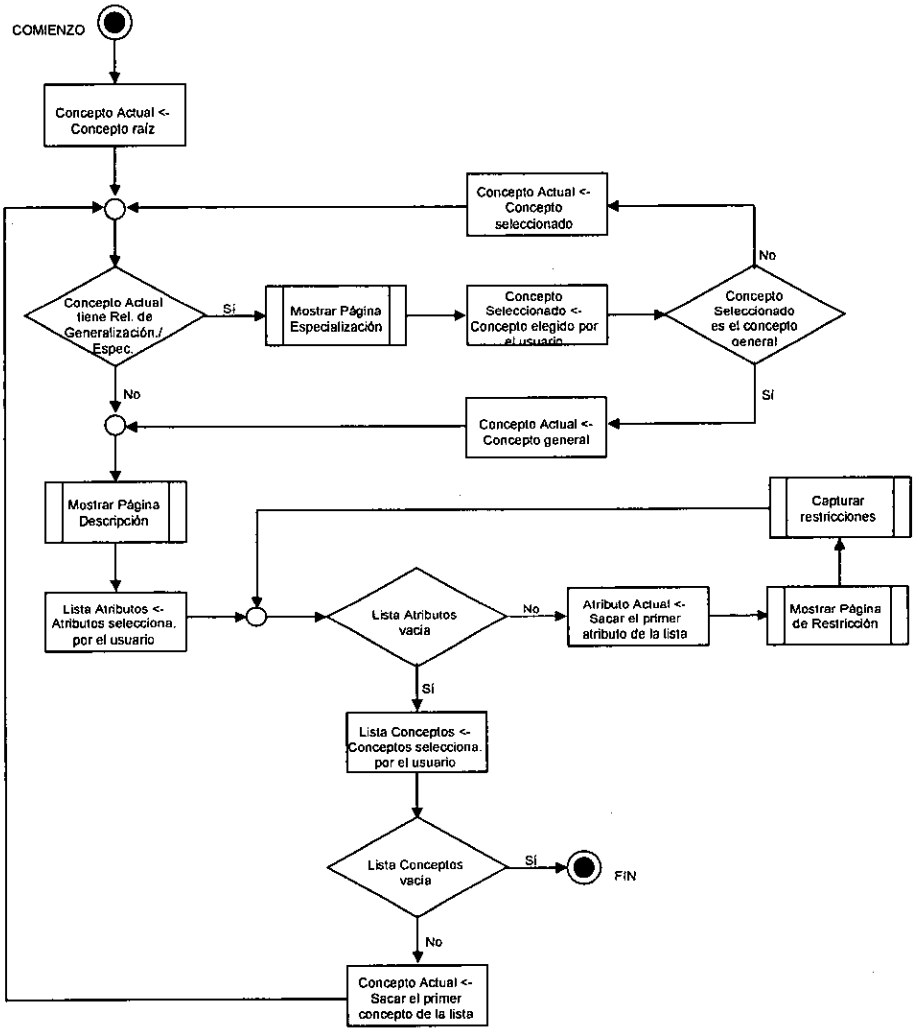


Fig. 62. Diagrama de flujo para la construcción de la Interfaz de Consulta

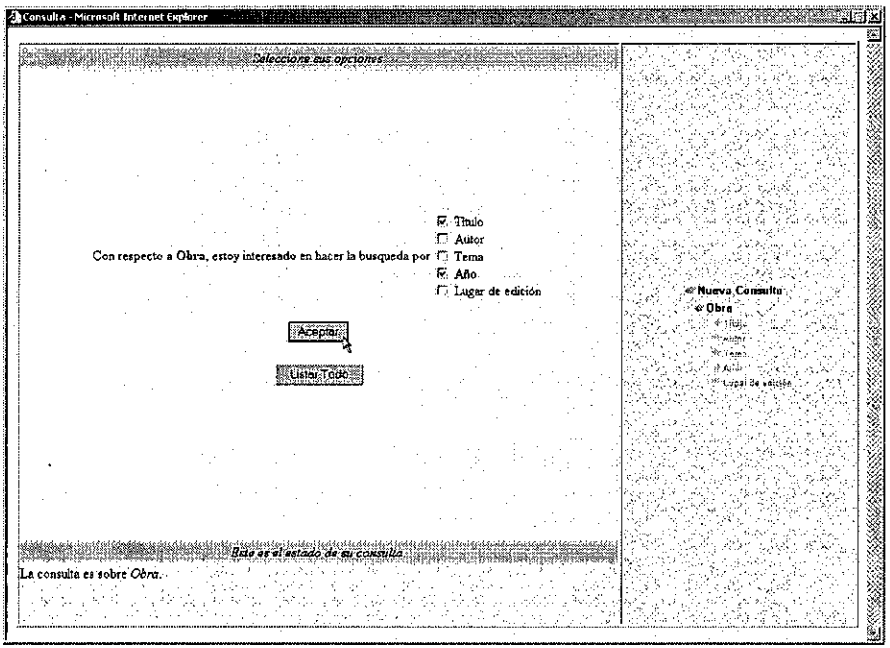


Fig. 63. Ejemplo de pantalla de consulta

Por ejemplo, si el `concepto-actual` fuese "Emblema", del Árbol de Conceptos de la Fig. 51, y este concepto no tuviese asociada una Metáfora Cognitiva, entonces se usaría por defecto el Esqueleto de Frase de Descripción que daría lugar a la siguiente frase en Lenguaje Natural Acotado:

```
"Con respecto al concepto Emblema, estoy
interesado/a en expresar restricciones sobre:
  -Palabras Clave
  -Onomástica
  -Exempla
  -Mote
  -Epigrama
  -Imagen
  -Glosa
  -Autoridades."
```

Si suponemos que el usuario selecciona "Palabras Clave", "Exempla", "Mote" y "Glosa", entonces el estado de las listas sería el siguiente: la `ListaAtributos` contendría "Palabras Clave" y "Exempla", y la `ListaConceptos` contendría "Mote" y "Glosa".

Hay que destacar que, en este paso, el usuario dice en qué características del `concepto-actual` está interesado. Esto significa que el usuario quiere

restringir esas características para especificar cómo tiene que ser el `concepto-actual`. En el ejemplo anterior, el usuario está diciendo que las "Palabras Clave", el "Exempla", el "Mote" y la "Glosa" del emblema son las características que quiere restringir para describir los emblemas que está buscando.

Estas dos listas se usarán para continuar con el proceso. El algoritmo usará los conceptos y atributos de ambas listas para continuar su ejecución.

Para cada atributo de la `ListaAtributos`, se le presenta al usuario la frase en Lenguaje Natural Acotado o la Metáfora Cognitiva que tiene asociado dicho atributo en el Árbol de Conceptos, para que el usuario exprese restricciones sobre él. Recuérdese que, en ocasiones, los atributos del Árbol de Conceptos tienen asociados los valores posibles para cada uno de ellos (cuando estos constituyen un conjunto pequeño). Así, las frases en Lenguaje Natural Acotado que piden restricciones sobre este tipo de atributos, tienen una lista desplegable que rellenan, en tiempo de ejecución, con el conjunto de valores asociados al atributo en cuestión. Así, si el conjunto de valores de un cierto atributo se modifica en las bases de datos integradas, bastará modificar el documento XML que representa el Árbol de Conceptos, añadiendo o borrando los valores asociados al atributo en cuestión.

Las restricciones expresadas por el usuario son capturadas y enviadas al módulo Constructor y Distribuidor de Consultas, el cual se encarga de crear el documento en XML que representará la consulta completa del usuario.

Para cada concepto de la `ListaConceptos`, el algoritmo se llamará a sí mismo usando el concepto como parámetro de entrada. El proceso se termina cuando ambas listas `ListaAtributos` y `ListaConceptos` se vacían.

Por tanto, el Generador de la Interfaz de Consulta no es más que un motor que recorre el Árbol de Conceptos siguiendo las indicaciones del usuario, y mostrando en cada momento del proceso de consulta, la interfaz (que contiene una frase en Lenguaje Natural Acotado o una Metáfora Cognitiva) que corresponda. Dichos componentes los obtiene del almacén Componentes IU siguiendo el identificador que los atributos y conceptos tienen asociado en el Árbol de Conceptos.

Hay que destacar que, como consecuencia de esta arquitectura, los cambios que se realicen en el Árbol de Conceptos debidos a la incorporación de nuevos términos (conceptos y/o atributos), modificación o eliminación de términos existentes, no fuerzan a una recodificación del Generador de la Interfaz de Consulta. Esta característica de este módulo representa un punto a favor tanto de la escalabilidad del sistema global, como de su facilidad para acomodar cambios.

6.3 Representación, Construcción y Distribución de Consultas

Una vez formulada la consulta en términos de las Frases en Lenguaje Natural Acotado y las Metáforas Cognitivas, y capturadas las restricciones por el propio módulo Generador de la Interfaz de Consulta, es el momento en el que el sistema ha de construir la consulta en el Lenguaje de Consulta de la arquitectura (lenguaje que hemos definido en XML) para después redirigirla a los Sistemas Envoltorio asociados a las bases de datos implicadas en la misma.

En este apartado se describe el Lenguaje de Consulta de nuestro sistema, dejando su especificación formal para el apéndice IV, en el que se presenta su DTD y varios ejemplos de uso. Además, en este apartado se describe el funcionamiento del Constructor y Distribuidor de Consultas del sistema.

6.3.1 Lenguaje de Consulta

Puede definirse una consulta del usuario como una descripción de aquello (conceptos) que está buscando. Un usuario puede describir los conceptos que está buscando describiendo las características de dichos conceptos. Las características o propiedades de los conceptos que el usuario puede "describir" no son más que los atributos de los conceptos del Árbol de Conceptos del sistema. Por lo tanto, podemos decir que las consultas permitidas por el sistema son conjuntos de condiciones sobre los atributos del Árbol de Conceptos.

El tipo de condiciones que se permiten expresar sobre los atributos del Árbol de conceptos se formaliza a través del DTD Lenguaje de Consultas de la arquitectura. Se trata de un lenguaje XML que hemos definido para comunicar el Mediador con los Sistemas Envoltorio. En el Apéndice IV se presenta el DTD del Lenguaje de Consulta y en el Capítulo 7 aparecen varias consultas y su representación en el Lenguaje de Consulta.

Hay que destacar que todas las condiciones están conectadas implícitamente por el conector lógico and. Por lo tanto, no se almacenan los conectores, sólo las condiciones.

En el documento XML de una consulta se mantiene la posición relativa que los conceptos y atributos tienen dentro del Árbol de Conceptos. No podía ser de otra manera porque la posición de un término en el Árbol de Conceptos forma parte de la información necesaria para identificar a dicho término.

En el árbol de la consulta sólo aparecen aquellos términos del Árbol de Conceptos sobre los que el usuario ha expresado alguna condición. Por lo tanto, una consulta en el Lenguaje de Consulta es un árbol, que es un

subconjunto del Árbol de Conceptos donde cada atributo tiene asociada una condición.

6.3.2 El módulo Constructor y Distribuidor de Consultas

El Constructor de Consultas actúa recorriendo la estructura en la que están recogidas las restricciones de la consulta del usuario y generando, a partir de ellas, la consulta en el formato dado por el Lenguaje de Consultas del sistema. A medida que el Constructor de Consultas genera el documento XML, el Distribuidor de Consultas obtiene del Árbol de Conceptos, para cada atributo implicado en la consulta, la lista de bases de datos que tienen asociada. De esta forma, una vez finalizado el proceso de construcción del documento XML, el Distribuidor de Consultas ya sabe a qué bases de datos redirigirlo.

En la Fig. 64 se presenta el módulo Constructor y Distribuidor de Consultas enmarcado en la capa 2 de la arquitectura (Mediador).

Cada Componente IU, ya sea frase en Lenguaje Natural Acotado (LNA) o Metáfora Cognitiva, lleva embebido el fragmento de la consulta en XML al que da lugar. Por otro lado, cuando el usuario rellena los huecos de una frase en LNA o expresa alguna condición en una Metáfora Cognitiva, el propio módulo Generador de la Interfaz de Consulta captura los valores concretos que introduce el usuario. De esta manera, el Constructor de Consultas sólo necesita recorrer la estructura en la que están almacenadas las restricciones que han sido expresadas en la Interfaz de Consulta para poder crear el documento XML que representa la consulta final.

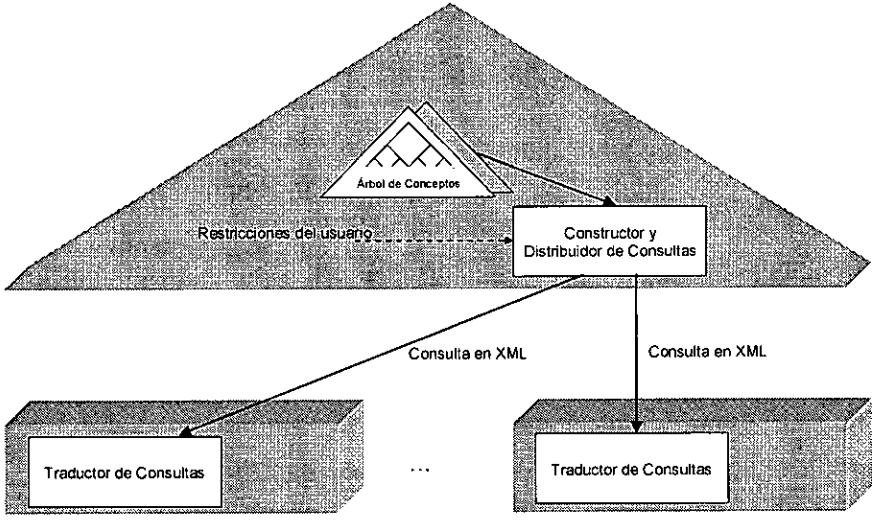


Fig. 64. Módulos implicados en el envío de las consultas del usuario a las bases de datos

Una vez que el usuario completa la consulta y el Constructor de Consultas la formatea según el Lenguaje de Consultas, el siguiente paso es redirigirla a las bases de datos para ser ejecutada.

El Distribuidor de Consultas redirige las consultas en XML a los Sistemas Envoltorio de las bases de datos implicadas en dicha consulta. Allí, es el módulo Traductor de Consultas, el que recibe las consultas en XML.

Cada atributo del Árbol de Conceptos, como hemos visto en la sección 6.2, tiene asociada la lista de bases de datos en la que existe. El Distribuidor de Consultas lee los Ids de las bases de datos que almacenan información sobre los atributos implicados en la consulta. La lista de bases de datos a las que se les van a enviar una consulta determinada es la unión de las listas de bases de datos asociadas a los atributos que forman parte de dicha consulta. Por lo tanto, existe la posibilidad de que algunas de las bases de datos que reciben la consulta no almacenen datos sobre algunos de los atributos sobre los que se ha especificado la consulta, y, por lo tanto, sólo puedan ejecutar parte de la consulta. De esto será informado el usuario en la Interfaz de Respuesta.

Por ejemplo, supongamos una consulta en la que se expresen condiciones sobre los atributos "Año de Edición" y "Lugar de Edición". Supongamos que el atributo "Año de Edición" tiene asociada la siguiente lista de bases de datos `[BD1, BD2]` y el atributo "Lugar de Edición" tiene asociada la lista `[BD1, BD3]`. La consulta será enviada por el Distribuidor de Consultas a las bases de datos `[BD1, BD2, BD3]` aunque sólo `BD1` almacena información sobre los dos atributos a la vez.

Además de enviar el documento XML de la consulta a los Sistemas Envoltorio apropiados, el Distribuidor de Consultas, envía al Gestor de la Presentación, los Ids de las bases de datos a las que ha enviado la consulta. Se pretende así que el Gestor de la Presentación construya una primera página de respuesta que liste las bases de datos a las que se ha enviado la consulta, junto con la consulta que se podrá ejecutar en cada una de ellas. Esta será la primera página de respuesta que se irá completando con un resumen de los resultados obtenidos en cada base de datos, a medida que el Mediador, y más específicamente, el Gestor de la Presentación, reciba las respuestas de los Sistemas Envoltorio.

Alternativa para la distribución de las consultas

Una posible alternativa al primer algoritmo de distribución de consultas que hemos propuesto (y que ha sido implementado) es el que planteamos en este apartado.

Básicamente, la alternativa consiste en enviar las consultas únicamente a aquellas bases de datos que estén totalmente implicadas en ellas. Es decir, enviar una consulta a aquellas bases de datos que almacenen información sobre todos y cada uno de los atributos implicados en dicha consulta.

La ventaja de este algoritmo es precisamente que sólo van a recibir las consultas aquellas bases de datos que puedan ejecutarlas completamente, por lo que las respuestas van a ser más precisas. El inconveniente obvio de este algoritmo es el tiempo que se consume en calcular a qué bases de datos enviar la consulta.

Sin embargo, puede ocurrir que alguna de las condiciones, aunque importante porque realmente ha sido expresada, no sea lo bastante restrictiva como para hacer que los documentos que no la cumplan no sean recuperados. Si se considera esta posibilidad, la opción que hemos elegido de enviar la consulta a todas las bases de datos que puedan ejecutar una parte de la consulta, y mostrar al usuario cuál ha sido la consulta ejecutada en cada base de datos, es la mejor opción.

6.4 Traducción de las Consultas

Una vez que la consulta es recibida en los Sistemas Envoltorio, más concretamente por el módulo Traductor de Consultas, comienza el proceso de traducción. Con la consulta en XML, por un lado, y el Árbol de Correspondencias que cada base de datos tiene asociado, por otro, los Traductores de Consultas, traducen las consultas, que llegan en el Lenguaje XML de Consulta, al lenguaje de consulta de la base de datos asociada (SQL en este caso) para que sean ejecutadas después en dichas bases de datos.

En este capítulo se describe el proceso de traducción de la consulta en XML al lenguaje de las bases de datos.

6.4.1 Algoritmo

Los Traductores de Consultas de los Sistemas Envoltorio (capa 3) reciben la consulta del usuario expresada en XML. Su tarea es traducir dicha consulta al lenguaje de la base de datos a la que están asociados. En nuestro sistema las tres bases de datos son relacionales, por lo que todas ellas usan SQL. En la Fig. 65 se muestran los módulos de la arquitectura implicados en la traducción de las consultas.

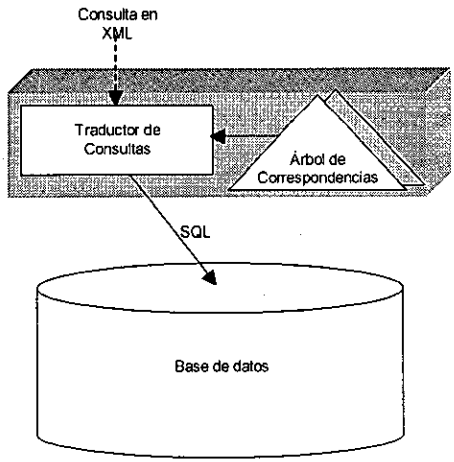


Fig. 65. Módulos implicados en la traducción de las consultas

Para llevar a cabo esta tarea, los Traductores de Consultas se guían por la consulta del usuario almacenada en el formato XML y obtienen la información necesaria para ejecutar la traducción del Árbol de Correspondencias que tienen asociado. Básicamente, el proceso consiste en, dada una consulta sobre un concepto (compuesta por un conjunto de condiciones sobre sus atributos) en XML, leer la Información de Correspondencia que dicho concepto, así como los atributos sobre los que existen condiciones, tienen asociada en el Árbol de Correspondencias y completar la sentencia SQL que finalmente será ejecutada en la base de datos.

El proceso de traducción está esquematizado en el diagrama de la Fig. 66.

La primera llamada del `Algoritmo-Traductor-Consulta` se hace pasándole como entrada el elemento `consulta` raíz del documento XML (documento que almacena la consulta). El Traductor de Consultas lee del Árbol de Correspondencias la información asociada al concepto que se especifica en el nombre de dicho elemento consulta.

La Información de Correspondencia de un término está clasificada en tres bloques identificados por tres etiquetas: `select`, `from` y `where`. Bajo cada una de estas etiquetas está el fragmento que hay que añadir a la sentencia SQL que se está creando, en la cláusula select, from o where, respectivamente.

Básicamente, el proceso de traducción consiste en ir añadiendo a las cláusulas select, from y where de la consulta lo que se indique en el Árbol de Correspondencias (ver un ejemplo en Fig. 55) para cada uno de los términos que participen en la consulta. En el Capítulo 7 se presenta un ejemplo detallado de este procedimiento.

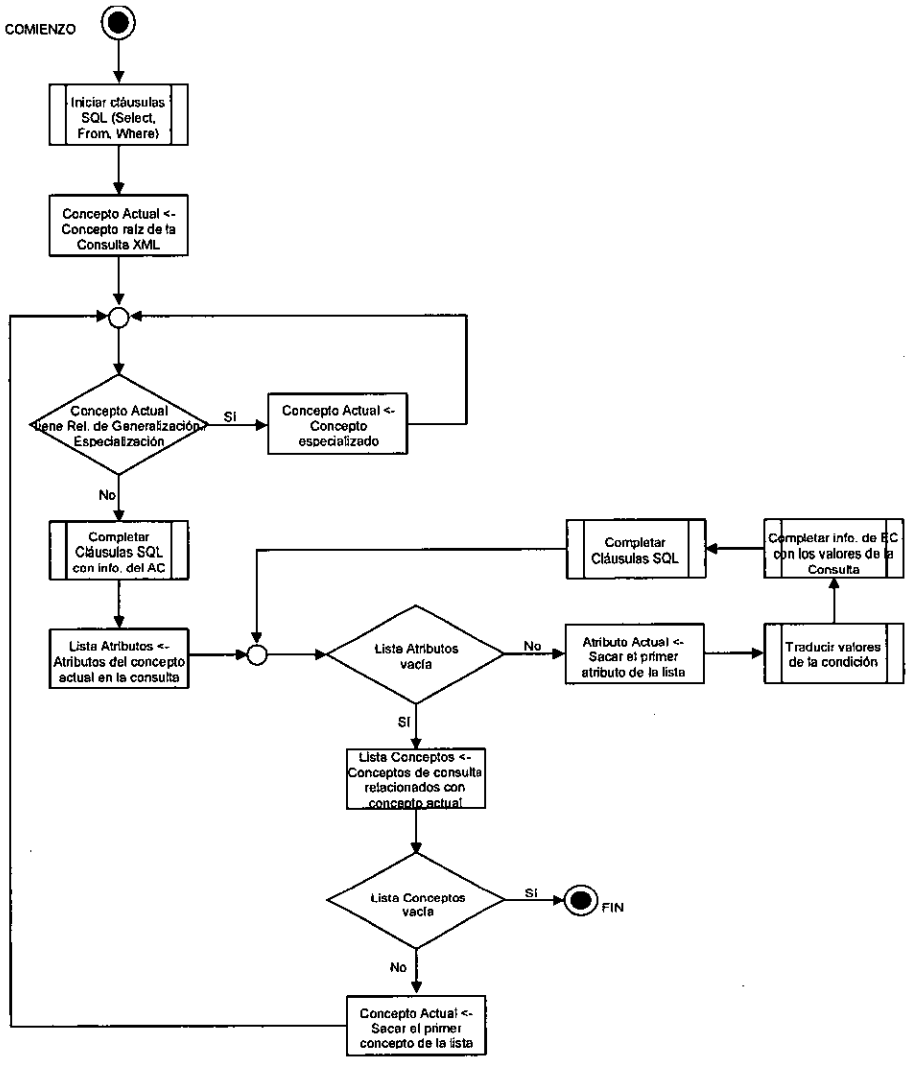


Fig. 66. Diagrama de flujo del algoritmo traductor de consultas

6.5 Gestor de la Presentación

Como ya hemos comentado, la primera página de respuesta del sistema, se genera inmediatamente después de que el módulo Constructor y Distribuidor de Consultas envíe la consulta a las bases de datos que están implicadas en ella y que, por tanto, podrán contestarle aunque sea de forma parcial.

El Constructor y Distribuidor de Consultas envía, en ese momento, al Gestor de la Presentación, la información necesaria para que este último

pueda construir una primera página de respuesta en la que esté reflejada la lista de bases de datos a las que se está interrogando, especificando para cada una de ellas:

- Su nombre o título,
- Una pequeña descripción de los datos que almacena,
- Una dirección, teléfono o email de contacto,
- La fecha de la última actualización de los datos que almacena,
- URL principal de la base de datos, y, por último,
- la consulta que se va a ejecutar en cada base de datos (por si alguna de las restricciones del usuario se expresó sobre conceptos que no existían en dicha base de datos).

A medida que se reciban las respuestas de las bases de datos, esta página se irá actualizando. Para cada Biblioteca Digital, se especificará el número de resultados obtenidos. Una vez que se obtenga este resultado, al usuario se le permite acceder una a una a cada base de datos para visualizar las respuestas obtenidas con su consulta a través de la propia interfaz de cada Biblioteca Digital.

Así como existe la necesidad de realizar las consultas de forma integrada sobre las tres bases de datos, en la fase de respuesta se ha tomado la decisión de visualizar los resultados a través de las propias interfaces de las Bibliotecas Digitales. En nuestro caso, las principales razones que nos han llevado a tomar esta decisión son las siguientes:

- Las tres Bibliotecas Digitales contarán con Interfaces de Respuesta bien adaptadas a sus datos y bien diseñadas, como ya ocurre con la Biblioteca de Literatura Emblemática, por lo que cualquiera de dichas interfaces van a ser siempre más intuitivas y amigables que cualquier interfaz que podamos construir para visualizar de forma integrada los resultados de las tres bases de datos.
- Las bases de datos son muy heterogéneas por lo que no existirán, en ningún caso, duplicados en los resultados obtenidos por una consulta.

Existe una razón más, que no se presenta en nuestro sistema, pero que sí es muy común que se dé en un Sistema de Acceso Integrado a bases de datos y es que, los administradores de las bases de datos no siempre estarán dispuestos a que la información de su base de datos se muestre a través de una interfaz que no sea la propia de la base de datos. Es más, incluso existirán casos en los que ni siquiera será posible hacer el enlace directamente con la Interfaz de Respuesta de la base de datos y será necesario que el usuario repita la consulta en la propia Web de la base de datos para acceder a las respuestas

obtenidas. En estos casos, nuestro sistema serviría únicamente para localizar fuentes interesantes para el tema sobre el que está buscando información el usuario.

6.6 Resumen

En este capítulo hemos descrito en detalle los módulos del sistema, así como el Lenguaje de Consulta que comunica el Mediador con los Sistemas Envoltorio.

Capítulo 7

Funcionamiento

7.1 Introducción

En este capítulo se presenta el funcionamiento del Sistema de Acceso Integrado a través de un ejemplo.

Vamos a suponer que queremos formular al sistema una cierta consulta y describiremos como es el proceso de consulta completo, desde que se expresa dicha consulta en la Interfaz, hasta que el usuario recibe el resumen de los resultados obtenidos por dicha consulta en cada base de datos.

7.2 Generador de la Interfaz de Consulta

Supongamos que un usuario esté interesado en expresar una consulta, y veamos cómo el sistema usa el algoritmo explicado en el capítulo anterior para ayudarlo en la tarea de expresar dicha consulta.

Cuando el usuario accede al sistema, el Generador de la Interfaz de Consulta comienza leyendo la raíz del Árbol de Conceptos. Si observamos el Árbol de Conceptos de nuestro sistema (Fig. 51), el concepto raíz es "Obra". Como este concepto tiene una relación de Generalización / Especialización, el usuario debe decidir si está interesado en el concepto general o en alguna de sus especializaciones. Es decir, para nuestro Árbol de Conceptos, el usuario tiene que elegir el tipo de obras en el que está interesado: "Relaciones de Sucesos", "Libros de Emblemas", "Libros de emblemas traducidos" u "Obras de cualquier tipo".

Para saber qué interfaz Web mostrar en este punto, el Generador de la Interfaz de Consulta comprueba, si en el Árbol de Conceptos la Relación de Generalización/Especialización del concepto "Obra" tiene asociado el

identificador de alguna Metáfora Cognitiva o Frase en Lenguaje Natural Acotado.

En el caso de nuestro Árbol de Conceptos, sí existe el identificador de una Metáfora Cognitiva (ver documento XML del Árbol de Conceptos en el Apéndice I) por lo que la interfaz web que se genera para expresar esta parte de la consulta completa se muestra en la Fig. 67. Hay que destacar que en el diseño de esta interfaz se ha usado también la técnica de la Aproximación Navegacional de manera que, el usuario puede elegir, pinchando en la estantería apropiada, el tipo de obras que desea recuperar.

En caso de que no hubiese ninguna Metáfora Cognitiva asociada, automáticamente el Generador de la Interfaz de Consulta generaría la Frase de Especialización (ver Fig. 58) correspondiente para esa Relación.

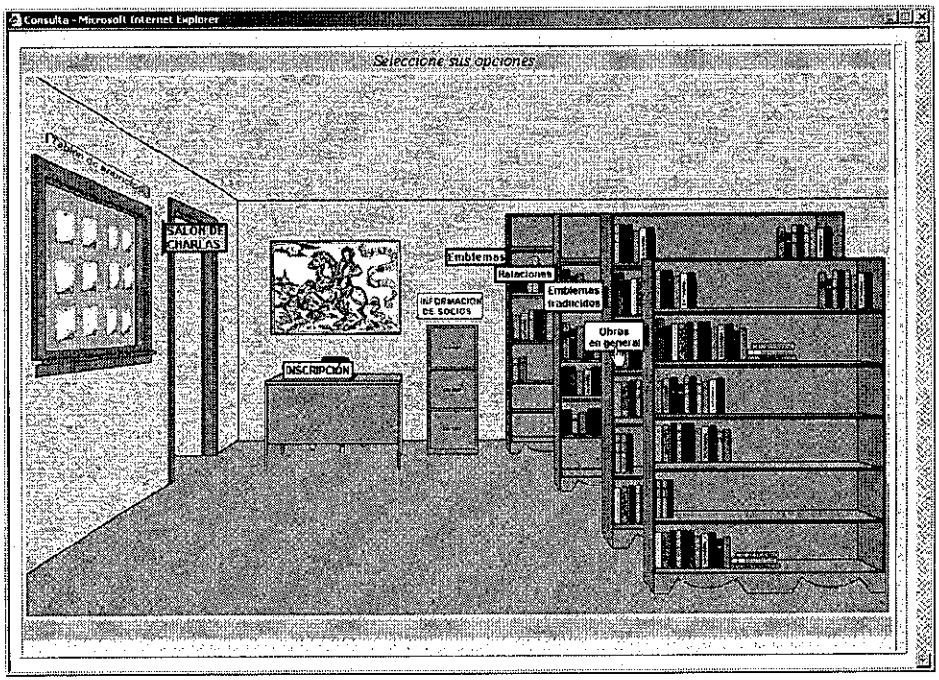


Fig. 67. Interfaz Web basada en la Metáfora Cognitiva de Biblioteca

Supongamos que el usuario pincha en la estantería etiquetada como "Obras en general". Después de este paso, en el área de la pantalla dedicada al estado actual de la consulta del usuario, pondría *"Tengo interés en encontrar obras de cualquier tipo"*.

Como se ha elegido la estantería "Obras de cualquier tipo", el Generador de la Interfaz de Consulta sabe que el usuario ha elegido bajar por la Parte

General del Árbol de Conceptos (Fig. 51) y no por ninguna de sus especializaciones (Parte Experta).

El siguiente paso que ejecuta el algoritmo es la utilización de la Frase de Descripción que permite que el usuario decida qué características del concepto "Obra" quiere restringir para describir las obras que está buscando, es decir, le presentará al usuario una interfaz que le permita seleccionar qué atributos del concepto "Obra" y qué conceptos, de los relacionados con él a través de una Relación de Descripción ("has"), quiere restringir.

De nuevo la interfaz que va a mostrar será la que se le indique en el Árbol de Conceptos. Si observamos nuestro Árbol de Conceptos (ver documento XML del Árbol de Conceptos en el Apéndice I), el concepto "Obra" no tiene asociado ningún identificador de frase en Lenguaje Natural Acotado ni Metáfora Cognitiva. Por lo tanto, el Generador de la Interfaz de Consulta mostrará una Frase de Descripción que construirá a partir del Esqueleto de Frase de Descripción estándar (Fig. 60). La frase en Lenguaje Natural Acotado instanciada a partir de dicho esqueleto es la siguiente:

"En relación a **obras** en general, tengo interés en expresar restricciones sobre:
- ☐ Título
- ☐ Autor
- ☐ Tema
- ☐ Año
- ☐ Lugar de Edición"

En nuestro ejemplo, supongamos que el usuario selecciona *Tema* y *Año* (porque quiere expresar restricciones sobre estos dos atributos). Esto significa que los documentos en los que el usuario está interesado tratan un cierto tema y pertenecen a una cierta época.

Finalmente, el Generador de la Interfaz de Consulta muestra las frases en Lenguaje Natural Acotado o las correspondientes Metáforas Cognitivas asociadas a los atributos *Tema* y *Año* para que el usuario exprese las condiciones que desea sobre ellos. Como ya sabemos, estas frases o Metáforas Cognitivas se consiguen del almacén de Frases y Metáforas (Componentes IU en el Mediador) usando el identificador que estos atributos tienen asociados en el Árbol de Conceptos. El atributo *Tema* tiene asociado un Esqueleto de Frase que una vez instanciado produce la frase en Lenguaje Natural Acotado siguiente (en la que las palabras en cursiva de dentro de los rectángulos son las que ya ha introducido el usuario):

"El **tema** de las obras debe estar definido por los siguientes fragmentos de palabras: *pecado, clero, Inquisición, hoguera, bruja*"

Igualmente, la frase en Lenguaje Natural Acotado para "Año" que se le presentaría al usuario sería la siguiente:

"El **año** de las obras debe estar entre [*1500*] y [*1650*]"

Después de rellenar los huecos, la consulta ya está completa y puede ser enviada al Constructor y Distribuidor de Consultas para que siga con el proceso.

Como vemos, la consulta completa que escribió el usuario es la siguiente:

"Me interesa encontrar obras de cualquier tipo. El tema de estas obras debe estar definido por los siguientes fragmentos de palabras: pecado, clero, Inquisición, hoguera, bruja. El año de estas obras debe estar entre el 1500 y el 1650."

Obsérvese que dicha consulta es compleja y sería difícil de expresar usando una técnica clásica de formularios.

7.3 Representación, Construcción y Distribución de Consultas

En esta sección se usará el ejemplo de consulta anterior para mostrar cómo se genera el documento XML, con las restricciones del usuario, una vez que la consulta completa ha sido formulada.

El estado de la consulta en XML después del primer paso se muestra en la Tabla 4. Como podemos comprobar, en este paso únicamente queda reflejado que el usuario está interesado en "Obras en general" y que las restricciones de la consulta se expresarán sobre atributos y conceptos de la Parte General de nuestro Árbol de Conceptos.

Tabla 4. Fragmento XML que expresa el primer paso de la consulta

```
<consulta>
   <concepto nombre="Obra">
   ...
   </concepto>
</consulta>
```

Tabla 5. Fragmento XML que expresa el segundo paso de la consulta

El documento XML resultante del segundo paso, donde el usuario indica que quiere expresar restricciones sobre los atributos "Tema" y"Año" de las obras se muestra en la Tabla 5.

Como podemos comprobar, todavía no se han expresado las condiciones. Con este paso el usuario simplemente ha decidido qué atributos y qué conceptos de los relacionados con el concepto "Obra" a través de una

Relación de Descripción, quiere restringir para describir las Obras que está buscando.

```
<consulta>
   <concepto nombre="Obra">
      <atributo nombre="Tema">
      ...
      </atributo>
      <atributo nombre="Año">
      ...
      </atributo>
   </concepto>
</consulta>
```

Tabla 6. XML que expresa la consulta completa

Por cada condición que finalmente establece el usuario sobre los atributos "Tema" y "Año", el Constructor y Distribuidor de Consultas construye la parte correspondiente del documento XML que almacena la consulta.

La consulta final, expresada en XML, se muestra en la Tabla 6.

```
<consulta>
   <concepto nombre="Obra">
      <atributo nombre="Tema">
         <contiene limite=5/>
         <valor cons="pecado"/>
         <valor cons="clero"/>
         <valor cons="inquisición"/>
         <valor cons="hoguera"/>
         <valor cons="bruja"/>
      </atributo>
      <atributo nombre="Año">
         <entre/>
         <valor cons="1500"/>
         <valor cons="1650"/>
      </atributo>
   </concepto>
</consulta>
```

La representación gráfica de esta consulta se muestra en la Fig. 68.

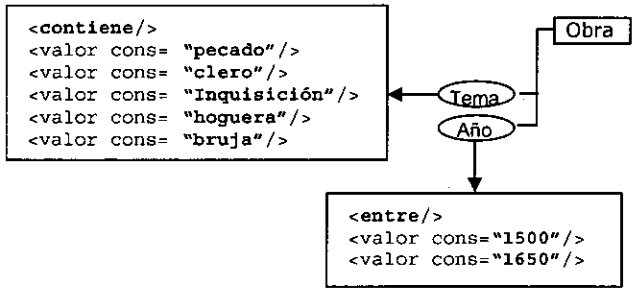


Fig. 68. Representación gráfica de la consulta de ejemplo

7.3.1 Otro ejemplo de consulta

A continuación se muestra otro ejemplo de consulta y su representación en el Lenguaje de Consulta. La consulta en lenguaje natural (cursiva) representa una consulta expresada en la Interfaz de Consulta.

Supongamos que una persona expresa las siguientes restricciones sobre los atributos "Año" y "Lugar" del concepto "Edición", e "Idioma" del concepto "Epigrama" de la Parte Experta del Árbol de Conceptos del sistema (Fig. 51), en concreto sobre el subárbol dedicado a Libros de Emblemas:

"Libros de Emblemas editados a partir de 1600 en Santiago de Compostela y que tengan el epigrama en latín."

El documento XML que genera el módulo Constructor y Distribuidor de Consultas a partir de las restricciones expresadas en la Interfaz de Consulta, se muestra en la Tabla 7.

Tabla 7. Consulta en XML

```
<consulta>
   <concepto nombre="Obra">
      <isa>
         <concepto nombre="Libro de Emblemas"
            <has>
               <concepto nombre="Edición">
                  <atributo nombre="Año">
                     <operador-binario op="mayor"/>
                     <valor cons="1600"/>
                  </atributo>
                  <atributo nombre="Lugar">
                     <operador-binario op="igual"/>
                     <valor cons="Compostela"/>
                  </atributo>
               </concepto>
               <concepto nombre="Emblema">
                  <has>
                     <concepto nombre="Epigrama">
                        <atributo nombre="Idioma">
                           <operador-binario op="igual"/>
                           <valor cons="Latín"/>
                        </atributo>
                     </concepto>
                  </has>
               </concepto>
            </has>
      </isa>
   </concepto>
</consulta>
```

La representación gráfica de esta última consulta se muestra en la Fig. 69.

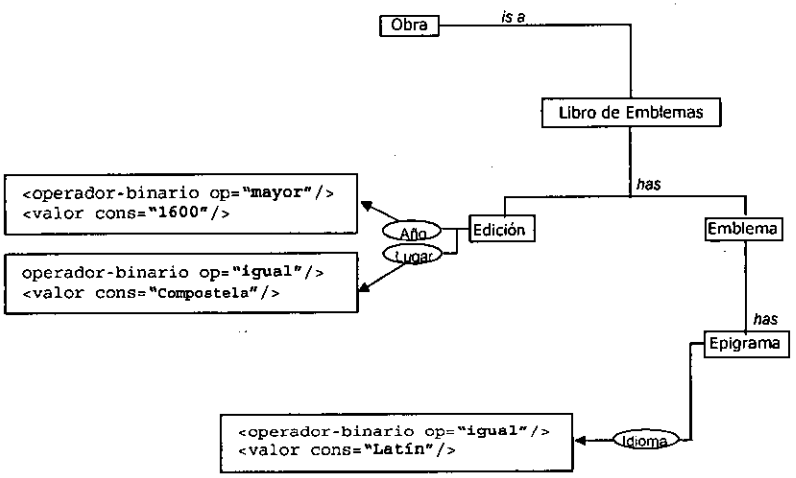


Fig. 69. Representación gráfica de la consulta de ejemplo.

7.3.2 Envío de la consulta a los Sistemas Envoltorio implicados

Dependiendo de los atributos (del Árbol de Conceptos) implicados en la consulta, esta puede ser enviada a todos los Sistemas Envoltorio o únicamente a parte de ellos.

En la consulta XML de la Tabla 6 intervienen los atributos "Tema" y "Año" del concepto "Obra" de la Parte General del Árbol de Conceptos. Podemos comprobar en el Árbol de Conceptos (apéndice I) que las listas de bases de datos que tienen asociadas ambos atributos contienen los identificadores de las tres bases de datos del sistema, por lo tanto, la consulta de la Tabla 6 será enviada a todas (las tres) bases de datos integradas en el sistema.

Por el contrario, la consulta de la Tabla 7 implica a los atributos "Año", "Lugar", "Epigrama" e "Idioma" de la Parte Experta de nuestro Árbol de Conceptos. En este caso, la consulta únicamente será enviada al Sistema Envoltorio de la bases de datos Libros de Emblemas.

7.4 Traducción de las Consultas

Los Traductores de Consultas de los Sistemas Envoltorio (capa 3) reciben la consulta del usuario expresada en el Lenguaje de Consulta. Su tarea es traducir dicha consulta al lenguaje de la base de datos a la que están asociados. En nuestro sistema todas las bases de datos son relacionales, por lo que todas ellas usan SQL. Para llevar a cabo esta tarea, los Traductores de

Consultas leen secuencialmente el documento XML y toman, para cada concepto o atributo, el fragmento de SQL (almacenado en el Árbol de Correspondencias) necesario para traducir la condición en XML establecida sobre dicho atributo o concepto.

Para una mejor comprensión del proceso de traducción, se presenta en la Fig. 70 el fragmento del Árbol de Correspondencias asociado a la base de datos Libros de Emblemas, que es necesario para traducir la consulta del ejemplo. En el Apéndice II se puede ver el documento XML completo correspondiente al Árbol de Correspondencias de la base de datos de Libros de Emblemas.

Recordemos que en los Árboles de Correspondencias existe un elemento `Correspondencia` para cada concepto o atributo. Hay que destacar que este elemento está compuesto por otros tres elementos identificados con las etiquetas `select`, `from` y `where`. Estos elementos contienen los fragmentos de SQL necesarios para acceder a la base de datos.

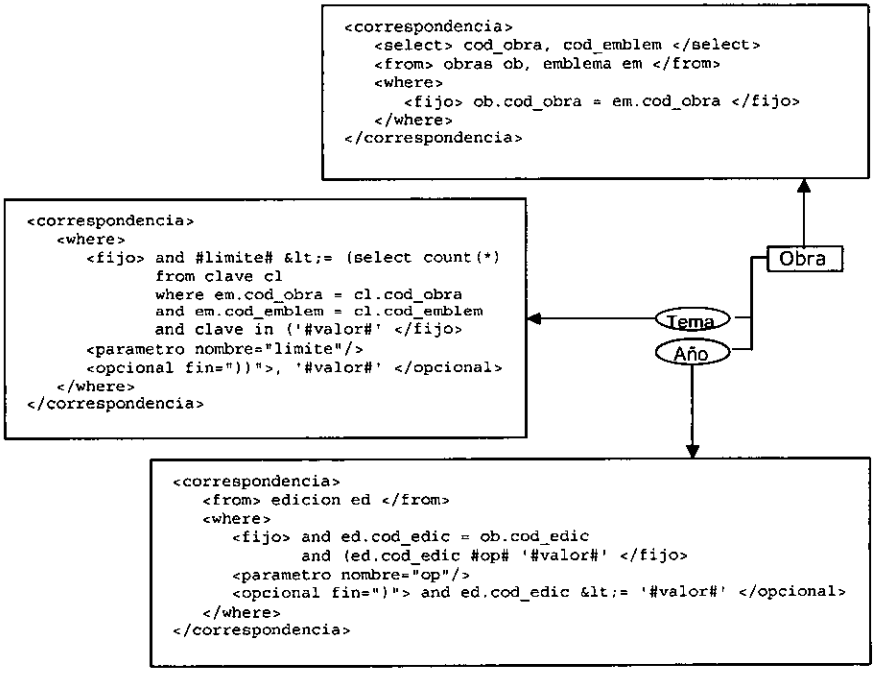


Fig. 70. Fragmento del Árbol de Correspondencias de Libros de Emblemas

El primer paso es traducir la parte de consulta XML que implica a un concepto. Como se puede comprobar la Fig. 70, la Información de Correspondencia del concepto "Obra" está dividida en la parte que hay que añadir a la cláusula `select`, a la cláusula `from` y a la cláusula `where` de la

SQL final. Por lo tanto, después del primer paso de traducción la consulta SQL quedaría como se muestra en la Tabla 8.

Tabla 8. Primer paso

CONSULTA EN XML	CONSULTA EN SQL
<consulta> <concepto nombre="Obra"> ... </concepto> </consulta>	select cod_obra, cod_emblem from obra ob, emblema em where ob.cod_obra = em.cod_obra

El segundo paso implica traducir una condición sobre el atributo "Tema". El proceso de traducción en el caso de un atributo consta de tres fases:

- En primer lugar es hacer una primera traducción de los valores que aparecen en la consulta XML a los valores que realmente se almacenan en la base de datos. Podemos comprobar en la Fig. 70 que para este atributo no es necesaria este tipo de traducción.
- En segundo lugar, se sustituyen los parámetros que aparezcan en la Información de Correspondencia del atributo por los valores concretos que estén almacenados en la consulta en XML. En este caso, es necesario sustituir dos parámetros: `#limite#` y `#valor#`, como se puede ver en la Fig. 70.
- En tercer lugar, se incorpora a la cláusula `where` de la consulta SQL final la Información de Correspondencia del atributo "Tema" una vez realizadas las sustituciones de las que hablábamos en el párrafo anterior.

  El resultado del segundo paso de la traducción se muestra en la Tabla 9.

Tabla 9. Paso dos

CONSULTA EN XML	CONSULTA EN SQL
<consulta> <concepto nombre="Obra"> <atributo= "Tema"> <contiene limite="5"/> <valor cons="pecado"/> <valor cons="clero"/> <valor cons="Inquisición"/> <valor cons="hoguera"/> <valor cons="bruja"/> </atributo> ... </concepto> </consulta>	select cod_obra, cod_emblem from obra ob, emblema em where ob.cod_obra = em.cod_obra and 5 = (select count(*) from clave cl where em.cod_obra = cl.cod_obra and em.cod_emblem = cl.cod_emblem and (cl.clave like 'pecado' or cl.clave like 'clero' or cl.clave like 'Inquisición' or cl.clave like 'hoguera' or cl.clave like 'bruja'))

El tercer paso de la traducción implica de nuevo a un atributo, en este caso "Año". El proceso es similar al paso anterior. La única diferencia está en que la Información de Correspondencia del atributo "Año" está dividida en dos partes, la parte que se debe añadir a la cláusula `from` de la SQL final y la parte que se debe añadir al `where`, después de sustituir los parámetros por los valores concretos de la consulta.

El tercer y último paso de la traducción se muestra en la Tabla 10. La consulta en SQL será la consulta que se ejecutará en la base de datos de Libros de Emblemas.

Tabla 10. Paso 3: Consulta para la bd de Libros de Emblemas

CONSULTA EN XML	CONSULTA EN SQL

`<consulta>`
`<concepto nombre="Obra">`
`<atributo nombre="Tema">`
`<contiene limite="5">`
`<valor cons="pecado"/>`
`<valor cons="clero"/>`
`<valor cons="Inquisición"/>`
`<valor cons="hoguera"/>`
`<valor cons="bruja"/>`
`</atributo>`
`<atributo nombre="Año">`
`<entre>`
`<valor cons="1500"/>`
`<valor cons="1650">`
`</atributo>`
`</concepto>`
`</consulta>`
	`select cod_obra, cod_emblem`
	`from obra ob, emblema em, edicion ed`
	`where ob.cod_obra = em.cod_obra`
	`and ed.cod_edic = ob.cod_edic`
	`and 5=`
	`(select count(*)`
	`from clave cl`
	`where em.cod_obra = cl.cod_obra`
	`and em.cod_emblem = cl.cod_emblem`
	`and (cl.clave like "pecado"`
	`or cl.clave like "clero"`
	`or cl.clave like "Inquisición"`
	`or cl.clave like "hoguera"`
	`or cl.clave like "bruja"))`
	`and (ed.cod_edic > 1500`
	`and ed.cod_edic < 1650)`

Las dos tablas siguientes (Tabla 11 y Tabla 12) muestran las sentencias SQL que serán ejecutadas por la base de datos Libros de Emblemas Traducidos y Relaciones de Sucesos, respectivamente. El proceso de construcción de estas dos consultas SQL es similar al descrito para la base de datos Libros de Emblemas.

Hay que destacar que la consulta final para la base de datos Libros de Emblemas Traducidos es muy similar a la consulta para la base de datos Libros de Emblemas. Sin embargo, y debido a que la base de datos de Relaciones de Sucesos está soportada por un sistema gestor de bases de datos Oracle 9i, con capacidades de recuperación de textos, la consulta final contiene una cláusula `contains`. Esta cláusula, ofrecida por el paquete Oracle *inter*Media, está incluida en Oracle desde la versión 8i y permite diferentes tipos de recuperación de textos.

Tabla 11. Consulta para la bd Libros de Emblemas Traducidos

```
select cdgo_de_obra, nmro_de_emblma
from obra ob, emblema em
where ob.cdgo_de_obra = em.cdgo_de_obra
and 5 = (select count(*)
   from palabraclave cl
   where em.cdgo_de_obra = cl.cdgo_de_obra
   and em.nmro_de_emblma = cl. nmro_de_emblma
   and (cl.clave like "pecado"
        or cl.clve like "clero"
        or cl.clve like "Inquisición"
        or cl.clve like "hoguera"
        or cl.clave like "bruja"))
and (ob.ano_de_edcn > 1500 and (ob.ano_de_edcn < 1650)
```

Tabla 12. Consulta para la bd de Relaciones de Sucesos

```
select tituloabre, cod_edic
from relacion rel, edicion ed
where rel.tituloabre = ed.tituloabre
and contains(rel.titulo, 'pecado & clero &
             Inquisición & hoguera & bruja', 10) > 0
and (rel.fecha_acon < 1500 and rel.fecha_acon > 1650)
```

7.5 Gestor de la Presentación

Como ya hemos visto en el capítulo anterior, una vez que el Constructor y Distribuidor de Consultas envía la consulta a las bases de datos que están implicadas en ella (en el ejemplo que venimos siguiendo son las tres bases de datos), le envía también al Gestor de la Presentación cierta información que le permite construir la primera página de respuesta.

En el caso de la consulta de nuestro ejemplo, la primera página de respuesta que muestra el sistema sería similar a la de la Fig. 71.

Como se puede observar, en esta pantalla se informa al usuario de las bases de datos a las que ha sido enviada la consulta, incluyendo para cada una de ellas cierta información básica que permite al usuario conocer la base de datos de la que va a recibir información.

A medida que se reciban las respuestas de las bases de datos, esta página se actualizará, actualizando el número etiquetado con "Resultados obtenidos". En cuanto este valor se actualice para una base de datos, se permitirá al usuario acceder a la Interfaz de Respuesta de dicha base de datos.

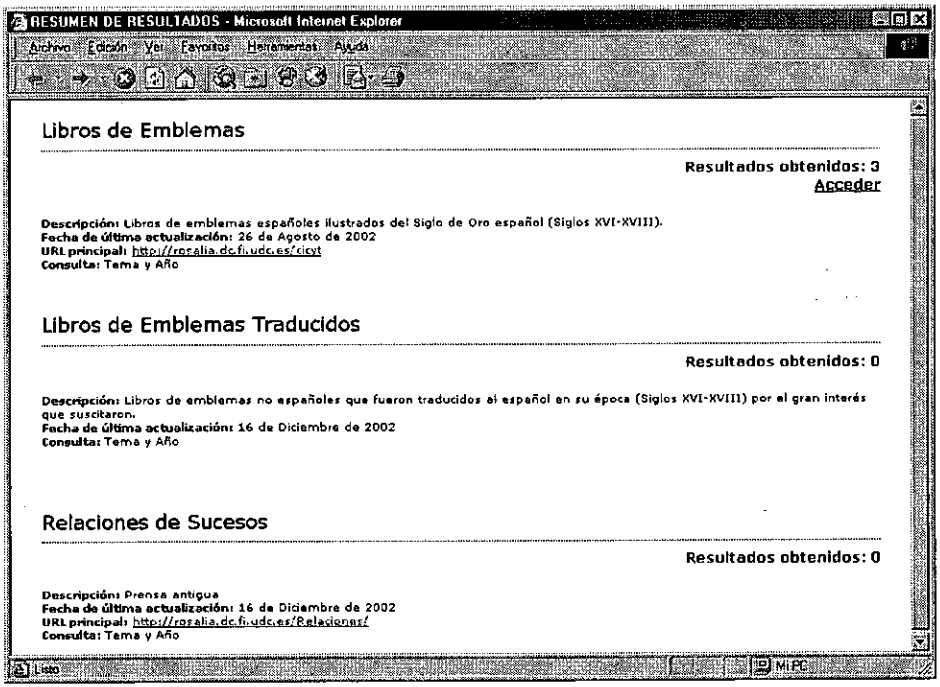


Fig. 71. Primera pantalla de respuesta

En el caso de nuestra consulta, como podemos ver en la Fig. 71, los resultados obtenidos para la base de datos Libros de Emblemas son 3, lo cual quiere decir que existen 3 emblemas que cumplen las condiciones que se han planteado en la consulta.

Pinchando sobre el enlace que se habilita en la página de resumen ("Acceder"), se carga la Interfaz de Respuesta propia de la base de datos Libros de Emblemas. Esta página se muestra en la Fig. 72.

En esta interfaz, el usuario podrá navegar a través de los datos, textos y páginas digitalizadas asociadas a las obras que ha recuperado en su consulta.

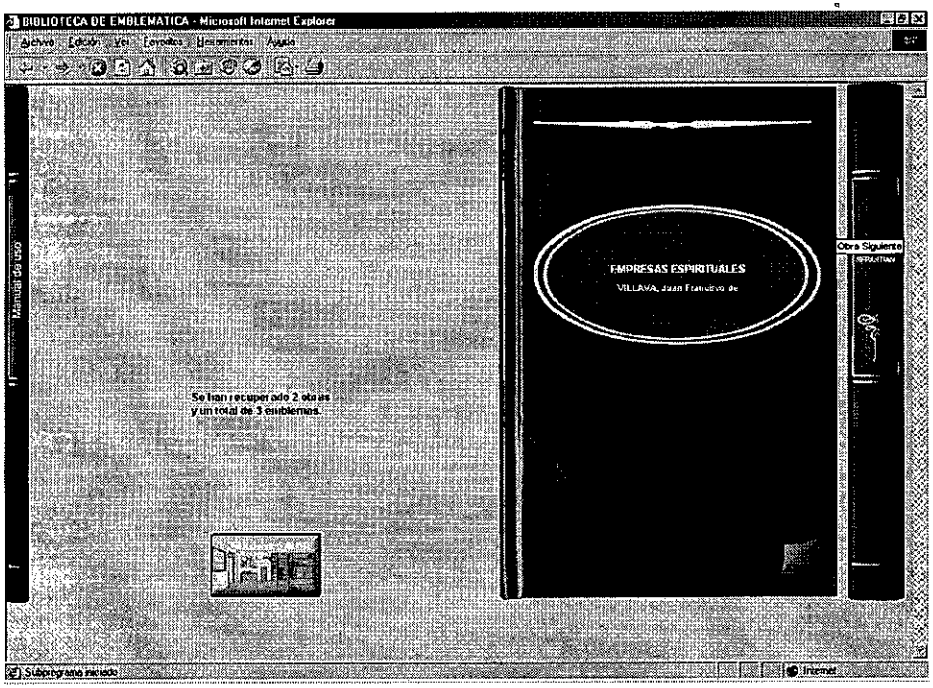


Fig. 72. Interfaz de Respuesta de la base de datos Libros de Emblemas

7.6 Resumen

En este capítulo hemos visto el funcionamiento del Sistema de Acceso Integrado a través del proceso completo de consulta.

Capítulo 8

Generalización de la arquitectura propuesta para Federación de Bibliotecas Digitales basada en Ontologías

8.1 Introducción

La arquitectura que hemos diseñado resuelve el problema concreto de integración del acceso a nuestras tres Bibliotecas Digitales del Siglo de Oro. Sin embargo, esta arquitectura ha sido diseñada de manera que pueda responder a la integración de Bibliotecas Digitales en general. Es decir, no se trata de una solución "ad hoc" sino de una solución perfectamente generalizable para el acceso integrado a bases de datos documentales de cualquier tipo e, incluso, a bases de datos en general. En este capítulo se va a detallar cómo se puede generalizar dicha arquitectura para adaptarse además a la federación de bases de datos más complejas y cómo se puede sacar partido del uso de ontologías para alcanzar dicho objetivo. En este sentido hemos logrado ya ciertas publicaciones relevantes [84, 85, 87, 91, 92,93, 95].

También en este capítulo se explicará cómo una pequeña modificación de los almacenes de datos de la capa del Mediador permiten que esta arquitectura se adapte y sea útil para la integración de bases de datos multilingües pudiendo además soportar preguntas en diferentes idiomas (cross-language). Es decir, consultas en cualquier idioma que recuperen datos en cualquier idioma de Bibliotecas Digitales Multilingües.

A pesar de que no hemos hecho ninguna implementación que soporte acceso multilingüe a Bibliotecas Digitales con documentos escritos en diferentes idiomas, la arquitectura adaptada a este fin ha sido publicada en [84, 92]. De hecho, esta es una de las líneas de trabajo futuro que pensamos desarrollar, ya que en estos momentos contamos con Bibliotecas Digitales en gallego y en español, por un lado, y, por otro lado, existe un claro interés por

parte de la comunidad internacional de investigación en Literatura Emblemática de crear un portal único que integre el acceso a las diferentes Bibliotecas Digitales de Literatura Emblemática que se han ido desarrollando en diferentes países[7]. Lógicamente, dicho portal deberá permitir consultas en varios idiomas, al menos en los principales idiomas europeos, a las citadas diferentes Bibliotecas Digitales de Literatura Emblemática. Es probable que nuestro Laboratorio se encargue de implementar dicho Portal Multilingüe de Acceso Integrado a las Bibliotecas Digitales de Literatura Emblemática Europea[8] [85].

8.2 Federación de bases de datos basada en ontologías

Cuando comenzamos el trabajo de investigación de esta tesis, estudiamos diferentes trabajos realizados en Federación de bases de datos usando ontologías [6, 46, 60]. Estos trabajos, aunque presentaban definiciones diversas de lo que se entendía por ontología, se adaptaban a la definición básica de ontología dada por Gruber en 1993:

> *"Una ontología es una especificación de una conceptualización" [34, 35]"*

Esta definición, como se observa, no presupone que la representación de la conceptualización capture más o menos semántica o más o menos restricciones en las relaciones entre los conceptos.

En 1998, Nicola Guarino, en un trabajo publicado en FOIS'98 (International Conference on Formal Ontology in Information Systems) [36], refina la definición de ontología dada por Gruber clarificando la diferencia entre una ontología y una conceptualización. Así, dice que, teniendo en cuenta que una ontología depende de un lenguaje y que una conceptualización no, es "imposible" alcanzar la "ontología perfecta" y clasifica las ontologías en poco detalladas (coarse) y de grano fino (fine-grained) según reflejen menos o más acertadamente dicha conceptualización. Sin embargo, aunque con calificativos (coarse o fine-grained) mantiene que todas ellas son ontologías.

Es más, en este mismo trabajo estudia los tipos de Sistemas de Información dirigidos por Ontologías y aclara:

---

[7] Habitualmente, esas Bibliotecas Digitales, como la nuestra, son monolingües dedicadas a la Literatura Emblemática del país en el que se crean.

[8] En estos momentos, hemos sido invitados, por la Society for Emblem Studies, a preparar una propuesta en este sentido y presentarla en su próxima reunión anual.

"Todos los Sistemas de Información (Simbólicos) tienen su propia ontología, ya que el sistema atribuye significado a los símbolos usados de acuerdo con un punto de vista particular del mini-mundo con el que trabaja". [36]

Al iniciar en 1999 este trabajo de tesis y hacer la presentación del proyecto en la comisión de doctorado, consideramos que el Árbol de Conceptos codificados en XML que representaría el conocimiento necesario para realizar el Acceso Integrado a las diferentes bases de datos era una ontología. Decidimos, además, que se adaptaba más a la definición de ontología que a la de esquema global, ya que en dicho Árbol no se representa, de ningún modo, el esquema global completo de las bases de datos que se integran, mientras que sí se representan las relaciones que tienen entre sí conceptos conocidos en el mundo real. Así, al depositar el proyecto de tesis, esta se tituló "Arquitectura para Federación de Bases de Datos Documentales basada en ontologías" y no "Arquitectura del Sistema de Acceso Integrado a Bases de Datos Documentales basada en Árboles de Conceptos", como la habríamos titulado hoy. De hecho, todos los trabajos publicados sobre esta tesis, en relación a la investigación realizada en integración de bases de datos, consideraban las Ontologías como la estructura en la que se basaba el Sistema de Acceso Integrado.

Sin embargo, los recientes trabajos de [15, 20, 40, 65], perfilan mucho más exactamente el concepto de ontología y se consensua que una ontología debe capturar de forma explícita las restricciones en las relaciones, tanto sintácticas como semánticas, definidas entre los conceptos, relaciones que una representación XML no es capaz de "enforzar". Es decir, se exige que la ontología, no sólo, represente las relaciones, sino también que obligue a que se cumplan. Evidentemente, XML es una representación estática y, por tanto, sólo capaz de forzar el mantenimiento de restricciones de tipo sintáctico, pero no las de tipo semántico.

Decidimos, por tanto, llamar Árboles de Conceptos a lo que hasta entonces habíamos denominado ontologías. Sin embargo, estamos convencidos de que nuestros Árboles de Conceptos en XML pueden ser perfectamente derivables de una ontología que capture realmente las relaciones existentes entre todos los conceptos relevantes para las bases de datos a integrar [15, 23, 40].

De hecho, la integración de las tres Bibliotecas Digitales del Siglo de Oro que hemos realizado representa un caso muy sencillo en las integraciones posibles con esta arquitectura, ya que siempre el usuario realiza búsquedas de "Obras". Si las bases de datos a integrar requirieran que el usuario pudiese hacer búsquedas no sólo de "Obras" sino también de "Bibliotecas", "Autores", "Impresores", etc. sería preciso generar, a partir de la ontología, diferentes Árboles de Conceptos, cada uno de los cuales tuviese como

concepto raíz los conceptos de "Biblioteca", "Autor", "Impresor", etc. Esta generación de los diferentes Árboles de Conceptos, a partir de una ontología real que capturase la globalidad del conocimiento del sistema, podría hacerse de una vez, cuando se crease la ontología o podría hacerse en tiempo de ejecución. Es decir, se podría generar, a partir de la ontología, el Árbol de Conceptos en el que el usuario estuviera interesado. A continuación se ilustra con un ejemplo sencillo cómo se pueden generar diferentes Árboles de Conceptos a partir de una ontología.

Supongamos que una ontología captura las relaciones existentes, en un conjunto de bases de datos, entre las bibliotecas en las que se almacenan los ejemplares de algunas de las diferentes ediciones que ha podido tener una obra que ha sido escrita por uno o varios autores. Con independencia de cómo se represente dicha ontología (lógica descriptiva, frames, etc.), es evidente que diferentes usuarios, pueden estar interesados en encontrar diferentes conceptos-objetos que requieran, por tanto, diferentes Árboles de Conceptos.

En la Fig. 73 se presentan tres Árboles de Conceptos que se podrían generar a partir de dicha ontología.

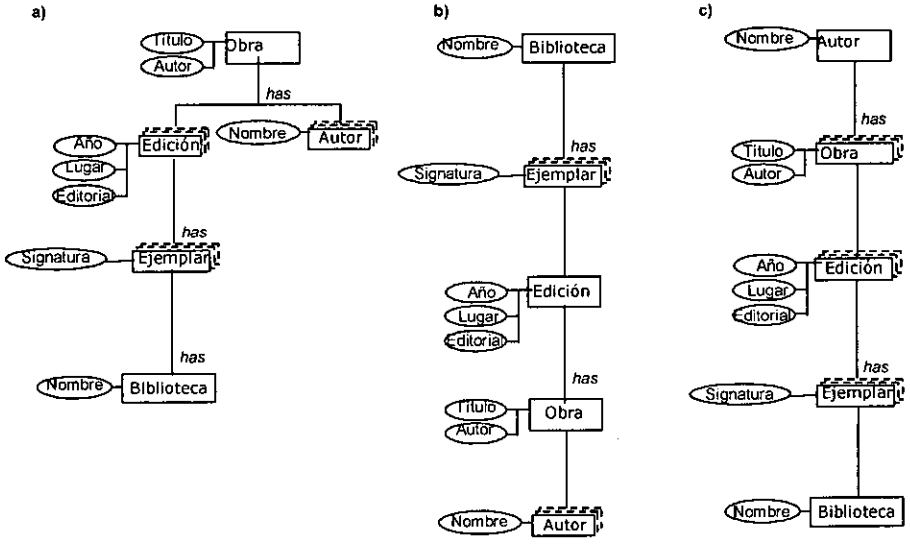


Fig. 73. Ejemplos de Árboles de Conceptos derivados de una ontología

Así, un usuario puede estar interesado en localizar las obras escritas por cierto autor o aquellas que tuvieron ediciones de cierta editorial. Para estos dos casos, el Árbol de Conceptos necesario sería aquel que tuviese obra como concepto raíz (Fig. 73.a). Sin embargo, otro usuario podría estar interesado en encontrar las bibliotecas en las que existen ejemplares de la edición realizada por cierta editorial de las obras escritas por cierto autor (Fig. 73.b). En este

caso, el Árbol de Conceptos debería tener como concepto raíz las bibliotecas. Por último, podría haber usuarios interesados en encontrar información sobre los autores (Fig. 73.c).

8.2.1 Arquitectura para Federación de Bases de Datos basada en Ontologías

Básicamente, la idea sería complementar la arquitectura del Sistema de Acceso Integrado a las tres Bases de Datos del Siglo de Oro español con un sistema que, utilizando la propia capacidad de inferencia de la representación ontológica generase el Árbol de Conceptos requerido para que el usuario exprese su consulta. Esta generación del Árbol de Conceptos podría hacerse, como ya se dijo, en tiempo de ejecución a partir de la respuesta del usuario a la pregunta inicial del Generador de la Interfaz de Consulta en la que se guiara al usuario en la elección del concepto raíz del Árbol de Conceptos que necesita.

Una posible y quizás más eficiente alternativa sería que los Árboles de Conceptos estuviesen ya generados a partir de la ontología y que el sistema simplemente escogiese el Árbol necesario a partir de la pregunta antes citada. En cualquiera de los dos casos, la ontología serviría para guiar la construcción automática del Árbol de Conceptos, una única vez antes de iniciar la explotación del sistema, o cada vez que un usuario se conecte. La representación de la arquitectura sería la que se presenta en la Fig. 74

Obsérvese como el Extractor de Árboles representa un módulo de software que utiliza las capacidades de inferencia de la ontología para generar los Árboles de Conceptos. Aunque nosotros representemos los Árboles de Conceptos como ya generados (Fig. 73), estos podrían realmente generarse en tiempo de ejecución dependiendo de las necesidades de implementación.

Evidentemente, con nuestra arquitectura en su estado actual, basada en Árboles de Conceptos y sin la existencia de una ontología propiamente dicha, se puede abordar, también, la existencia de un conjunto de Árboles de Conceptos útiles para permitir que el usuario haga preguntas sobre diferentes concepto-objetos (así lo describíamos en el apartado 5.4). Para ello, bastaría con tener los diferentes Árboles de Conceptos previamente generados en XML y que el Generador de la Interfaz de Consulta empezase por preguntar en cuál de los conceptos raíz de los Árboles está interesado. Sin embargo, esta aproximación tiene desventajas frente al uso de una representación ontológica de la que se deriven dichos Árboles de Conceptos automáticamente:

- Todos y cada uno de los Árboles de Conceptos hay que editarlos en XML manualmente, cada vez que se produzca cualquier modificación en las bases de datos,

– no hay ningún sistema que garantice la consistencia entre ellos.

Contando con una única representación ontológica de la que los Árboles se pudiesen generar automáticamente, bastaría con reflejar sólo en la ontología cualquier modificación en las bases de datos. Dicha modificación se reflejaría de forma automática en los Árboles de Conceptos al regenerarlos, y la consistencia entre ellos quedaría garantizada.

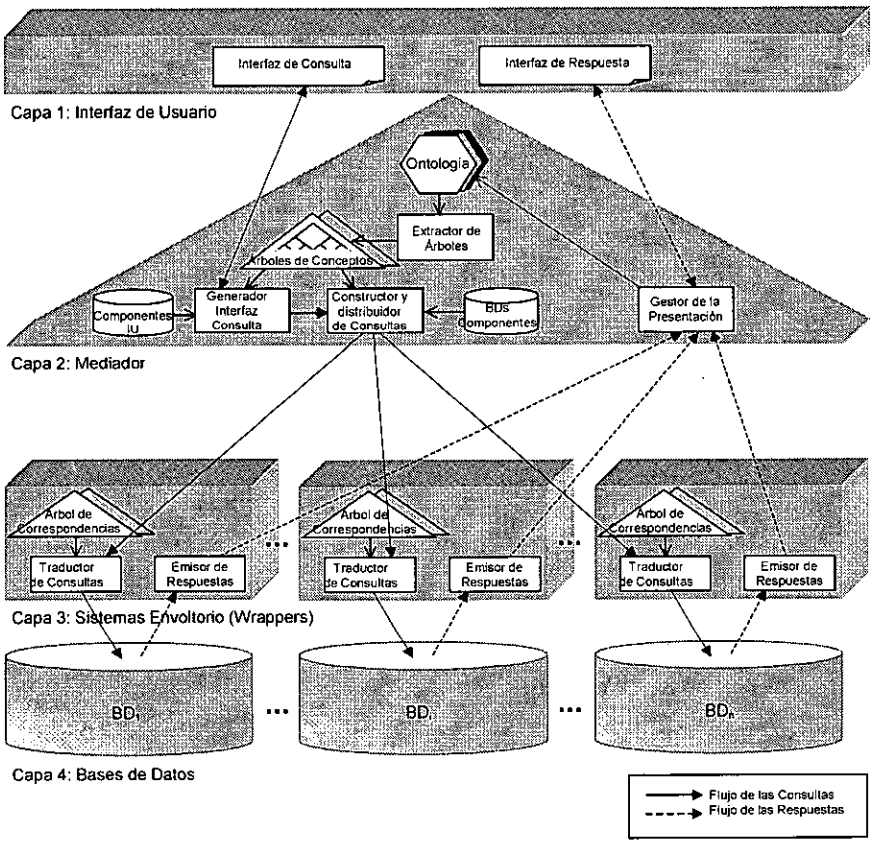


Fig. 74. Arquitectura del Sistema de Federación de Bases de Datos basada en Ontologías

8.2.2 Integración de las respuestas usando ontologías

Hemos denominado a la arquitectura que hemos diseñado arquitectura para el Acceso Integrado a Bases de Datos Documentales y no arquitectura de Federación de Bases de Datos Documentales puesto que consideramos que en la federación se asume un compromiso de integración de las respuestas que devuelven las consultas que plantean los usuarios.

Ya hemos justificado en el apartado 6.5 que, en el caso de bases de datos documentales, como las que aquí hemos estudiado, no tiene sentido la integración de las respuestas debido a que, incluso ejemplares duplicados en distintas bibliotecas, pueden ser de interés para los usuarios. Evidentemente, el que no sea necesario realizar una integración de las respuestas facilita enormemente la presentación de resultados, por lo que el Árbol de Conceptos que utilizamos es sencillo y ni siquiera corresponde a un esquema global completo de las bases de datos que forman parte del Sistema de Acceso Integrado.

Sin embargo, es fácil imaginar situaciones en las que sí que se requiera federar completamente las bases de datos de un sistema de modo que no sólo se desee tener un acceso integrado a las mismas sino también respuestas integradas e incluso reordenadas por nivel de relevancia a la consulta, y no clasificadas por corpus tal y como lo hace nuestro sistema. En este caso, de nuevo, la ontología puede jugar un papel fundamental en la federación como lo hace en sistemas como [6]. La idea es que ya que la ontología representa la totalidad del conocimiento existente en las diferentes bases de datos se tiene que poder utilizar dicho conocimiento para guiar, tanto la eliminación de duplicados, como la reorganización de los documentos y la gestión de su presentación. Evidentemente, conseguir derivar de la ontología este conocimiento para guiar a un módulo Generador de Interfaz de Respuesta es una tarea no abordada en esta tesis, ya que en sí mismo constituye un tema de investigación independiente.

8.3 Arquitectura integrar el acceso de Bibliotecas Digitales Multilingües

8.3.1 Motivación

Europa es un gran puzzle de lenguas y culturas de una riqueza que es necesario preservar. Sólo en la Unión Europea existen más de 50 lenguas autóctonas en uso, de las cuales sólo 11 de ellas son oficiales [28].

Con la Web convirtiéndose en el medio más importante para preservar y difundir cualquier manifestación cultural, está claro que sólo las lenguas con presencia en la Web tendrán la oportunidad de sobrevivir. Así, en toda Europa, se han creado bases de datos documentales que son auténticas Bibliotecas Digitales con documentos y obras literarias en distintas lenguas. El problema es que, en los casos en los que esas lenguas tienen pocos hablantes, los esfuerzos esporádicos por revivirlas no son suficientes, porque los visitantes de dichas Webs son pocos y su mantenimiento demasiado caro.

Integrar el acceso a Bibliotecas Digitales multilingües incrementará el número de visitantes a todos los sitios Web del sistema integrado. Este sistema se aprovechará del hecho de que algunas lenguas son lo suficientemente similares para ser entendidas por hablantes de otras lenguas (que son capaces de leer pero no de escribir una consulta directamente), y por otro lado, la integración facilitará a los investigadores internacionales el encontrar documentos de otras culturas, incluso cuando no entiendan el lenguaje en el que los documentos están escritos.

Supongamos, por ejemplo, que se desean encontrar los nombres de mujeres poetas que escribieron en finlandés en el siglo XIX, y los títulos de sus principales obras. A través de un Sistema de Acceso Integrado Multilingüe, un usuario podrá recuperar, aunque fuese capaz de leer ni escribir en finlandés, los nombres de dichas autoras y los títulos de sus obras. Incluso podía descargarlas para pedir a alguien la traducción de las mismas. De este modo, el Sistema de Acceso Integrado estaría dando un servicio que incrementaría el número de accesos a todas las bibliotecas integradas, pues facilitaría que cibernautas no hablantes de la lengua de una biblioteca concreta accedieran a la misma y sacaran partido de sus fondos y servicios.

Los Sistemas de Recuperación de Textos multilingües, por tanto, permitirán a sus usuarios realizar búsquedas de documentos escritos en lenguas que no les resulten familiares, a la vez que promoverán la difusión de culturas poco conocidas y el uso de lenguas minoritarias.

8.3.2 Arquitectura

La arquitectura del Sistema de Acceso Integrado que hemos construido ha sido generalizada para soportar la integración a Bibliotecas Digitales multilingües. Dicha generalización ha sido realizada exclusivamente en el plano teórico, pero aún así ha generado varias publicaciones [84, 92].

La arquitectura modificada para permitir acceso multilingüe conserva todas las ventajas de la Arquitectura para Acceso Integrado a Bibliotecas Digitales presentada en esta tesis, si se mantiene su facilidad de implementación, la facilidad para acomodar cambios y, lo que es más importante, la amigabilidad de la Interfaz de Usuario.

En este apartado se describen, las ampliaciones y/o modificaciones que serían necesarias realizar sobre la arquitectura de nuestro Sistema de Acceso Integrado para que soporte la integración de bases de datos de diferentes idiomas. En la Fig. 75 se muestra la capa 2 de la arquitectura, el Mediador, ya que es en esta capa donde es necesario realizar las modificaciones.

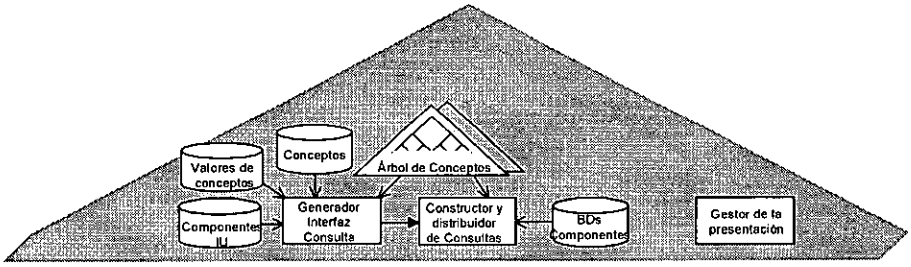


Fig. 75. Sistema de Acceso Integrado Multilingüe

8.3.3 Almacenes de Datos

Debido a que estamos tratando con bibliotecas digitales multilingües, y a que los usuarios deben poder elegir el lenguaje en el que interactuar con el sistema en la Interfaz de Usuario, necesitamos tener definido el marco (menús, ayudas, y demás elementos estáticos) de la Interfaz de Usuario en tantos idiomas como sea necesario. Además, necesitamos dos diccionarios, a mayores de los que ya describimos en la arquitectura monolingüe. Estos diccionarios están ubicados en la capa 2 (Mediador) de la arquitectura y son los siguientes:

- **Diccionario de Conceptos:** Almacena el nombre de los conceptos que aparecen en el Árbol de Conceptos en los idiomas disponibles en el sistema. En la Tabla 13 se muestra un ejemplo.

Tabla 13. Conceptos

English	*Galego*	*Castellano*	...
Sex	Sexo	Sexo	...
Genre	Xénero	Género	...
Date of Birth	Data de Nacemento	Fecha de Nacimiento	...
...	...	...	...

- **Valores de Conceptos:** si un concepto puede únicamente tomar un conjunto pequeño de valores, recordemos que en el Árbol de Conceptos estaban almacenados dichos valores. Para adaptar esta idea al sistema de acceso integrado a bases de datos multilingües, se ha incorporado un almacén en el que aparecen dichos valores en los idiomas disponibles en el sistema. En la Tabla 14 se muestra cómo está clasificada la información en este almacén.

Tabla 14. Valores de conceptos

English		*Galego*		*Castellano*		...
Sex	Female Male	Sexo	Muller Home	Sexo	Mujer Hombre	...
Genre	Drama Journal Novel Poetry Tale	Xénero	Teatro Xornalismo Novela Poesía Conto	Género	Teatro Periodismo Novela Poesía Cuento	...
	...		...		...	...

Aparte de incorporar dos diccionarios a la capa del Mediador, en la Arquitectura de Acceso Integrado Multilingüe es necesario modificar el almacén Componentes IU. Recordemos que en este almacén se encuentran las Frases en Lenguaje Natural Acotado y las Metáforas Cognitivas que servirán al sistema para construir la Interfaz de Consulta.

En la arquitectura de Acceso Multilingüe la información almacenada en el almacén Componentes IU está clasificada por idiomas. Si pensamos en las Frases en Lenguaje Natural Acotado, el texto de dichas frases deberá adaptarse al texto elegido por el usuario en la interfaz al comenzar la interacción con el sistema. De igual forma, el posible texto que pudiera aparecer en las Metáforas Cognitivas debe estar en el idioma seleccionado en la interfaz.

En el siguiente apartado se describe el funcionamiento del módulo Generador de Interfaces de Consulta de la Arquitectura de Acceso Integrado Multilingüe.

8.3.4 Generador de la Interfaz de Consulta

Básicamente, el funcionamiento de este módulo en la Arquitectura para Acceso Integrado Multilingüe es el mismo que en la Arquitectura Monolingüe. La diferencia radica en que el Generador de la Interfaz de Consulta debe recoger del usuario un primer parámetro, el idioma en el que el usuario desea interactuar con el sistema.

Una vez seleccionado este primer parámetro, el Generador de la Interfaz de Consulta ya sabrá en qué idioma mostrar el Árbol de Conceptos en la Interfaz de Consulta, qué Frases en Lenguaje Natural Acotado y qué

Metáforas Cognitivas tendrá que seleccionar del almacén Componentes IU, y qué lista de valores mostrar para los atributos que tengan finitas posibilidades.

8.4 Resumen

En este capítulo hemos planteado la generalización de la arquitectura del Sistema de Acceso Integrado para federar bases de datos documentales usando ontologías.

Además, se ha presentado una posible aplicación de dicha arquitectura que permite el acceso integrado a bases de datos de documentos escritos en diferentes idiomas a través de consultas expresadas también en diferentes idiomas.

Capítulo 9

Conclusiones y Trabajo Futuro

9.1 Introducción

En este capítulo se resumen los resultados obtenidos en este trabajo de tesis. En el apartado 9.2 se presentan nuestras conclusiones, indicando las aportaciones principales del trabajo realizado. Finalmente, en el apartado 9.3, se describen las líneas de trabajo futuro que abre esta tesis.

9.2 Conclusiones y aportaciones principales

Como ya se planteó en la introducción, la investigación realizada en esta tesis ha estado dirigida por dos objetivos principales:

- El diseño e implementación de Interfaces de Usuario intuitivas que propicien el éxito de cualquier sistema Web.
- La implementación de un Sistema de Acceso Integrado a las tres Bibliotecas Digitales del Siglo de Oro español, a través de una arquitectura fácilmente generalizable a la federación de bases de datos en general basada en ontologías.

Con respecto al primero de los objetivos, consideramos que con las tres técnicas de diseño de Interfaces de Usuario que hemos propuesto se pueden desarrollar Interfaces de Usuario tan intuitivas y fáciles de usar como nos propongamos, ya que dichas técnicas así lo permiten, tal y como ha quedado probado en la exitosa implementación de la interfaz de la Biblioteca Virtual Gallega, en donde se implementaron un buen número de las funciones necesarias en cualquier Biblioteca Digital.

Por otro lado, el Sistema de Acceso Integrado, así como cualquier otro sistema que se construya a partir de la arquitectura que hemos presentado, es

un sistema escalable, que se adapta fácilmente a los cambios que se produzcan en las bases de datos y, sobre todo, presenta, además una Interfaz de Usuario flexible, amigable y fácil de usar. Es decir, los Sistemas Acceso Integrado a Bibliotecas Digitales que se desarrollen de acuerdo a esta arquitectura, satisfarán los requisitos citados en el apartado 4.6. En concreto:

- **Escalabilidad del sistema**

  La escalabilidad del sistema está garantizada desde el momento que la Interfaz de Usuario actúa siguiendo la información almacenada en un único Árbol de Conceptos (una vez seleccionado éste por el usuario, en caso de haber más de uno, como se explicó en los apartados 5.4 y 8.2), independientemente del número de bases de datos integradas en el sistema. Del mismo modo, el Distribuidor de Consultas analiza las bases de datos a las que debe enviar la consulta, a partir de la información del Árbol de Conceptos, con independencia del número de bases de datos existentes en el sistema.

  Una vez que la consulta ha sido enviada a los diferentes Sistemas Envoltorio de las bases de datos, todos ellos trabajan en paralelo, por lo que de nuevo el número de bases de datos integradas no afecta al tiempo de respuesta.

  Por último, dado que el Gestor de la Presentación, simplemente, presenta el número total de documentos encontrados en cada base de datos, y facilita al usuario la conexión con cada una de las Bibliotecas Digitales implicadas, es evidente que, de nuevo, el que haya más Bibliotecas Digitales integradas no afecta al rendimiento.

- **Facilidad para adaptarse a cambios**

  El Árbol de Conceptos y los Árboles de Correspondencias minimizan los cambios que tienen que hacerse cuando una nueva base de datos se incorpora al sistema. De hecho, es necesario llevar a cabo únicamente dos tareas:

  A) Construir un Sistema Envoltorio ad hoc para la nueva base de datos lo que implica tan sólo crear el fichero XML con el Árbol de Correspondencias, ya que los módulos de software son iguales en todos los Sistemas Envoltorio. Evidentemente, también implica hacer los cambios necesarios para permitir que las respuestas de esa base de datos se ven a través del Sistema de Acceso Integrado.

  B) Completar el Árbol de Conceptos: añadir a los atributos del Árbol de Conceptos, relevantes para la nueva base de datos, el identificador de dicha base de datos, y completar, si fuera necesario, el Árbol de Conceptos con los nuevos conceptos y atributos que aparezcan en la nueva base de datos.

Sin embargo, no es necesario modificar el Mediador (ni el Generador de la Interfaz de Consulta ni el Constructor y Distribuidor de Consultas). El Generador de la Interfaz de Consulta generará dinámicamente la Interfaz de Consulta teniendo en cuenta los nuevos conceptos del Árbol de conceptos. De la misma forma, el Constructor y Distribuidor de Consultas podrá enviar las consultas del usuario a la nueva base de datos, sin más que leer la información asociada al Árbol de Conceptos.

Una cualidad interesante de nuestra arquitectura es que dota al sistema de una gran independencia lógica y física. El mantenimiento de la independencia lógica y física constituye un principio básico en bases de datos. Nuestra arquitectura extiende dicho principio a todo el Sistema de Acceso Integrado. Así los Árboles de Correspondencias proporcionan independencia física al sistema porque ningún cambio en las bases de datos componentes (SGBD, estructura de las tablas, etc.) va a afectar a los módulos del sistema. Únicamente afectarán a la información asociada a los términos de los Árboles de Correspondencias que se refleja, como ya se ha visto, en un documento XML fácil de modificar.

Por otro lado, el Árbol de Conceptos proporciona independencia lógica al sistema ya que el añadir, eliminar o modificar las bases de datos al Sistema de Acceso Integrado no afecta a ninguno de los módulos del sistema sino que sólo es necesario modificar el Árbol de Conceptos que, de nuevo, es un simple fichero de texto en XML fácil de modificar.

Por tanto, cualquier cambio en las bases de datos, sólo implica la modificación de ficheros de texto en XML, y no será necesario cambiar el código de ningún programa, ni, por tanto, recompilarlo.

- **Interfaz fácil de usar**

  La Interfaz de Usuario es amigable y fácil de usar. En su construcción hemos aplicado combinadas las tres técnicas de diseño que se han presentado.

  El hecho de que el Árbol de Conceptos pueda agrupar los conceptos en diferentes niveles de especialización (Parte General y Parte Experta, en nuestro caso), haciendo uso de las Relaciones de Generalización / Especialización, permite a los usuarios expresar consultas generales o consultas especializadas según su grado de conocimiento sobre el dominio de los corpus.

Además de estos tres requisitos que consideramos fundamentales que cumpla un Sistema de Acceso Integrado, particularmente para el Acceso Integrado a Bibliotecas Digitales es necesario (como ya vimos en el Capítulo 4) que se cumplan ciertos requisitos que también hemos tenido en cuenta en el sistema que hemos diseñado e implementado.

- **Conservación de capacidades para Recuperación de Textos**

  El Lenguaje de Consulta definido en nuestra arquitectura permite sacar partido de cualquier técnica de Recuperación de Textos que esté implementada en las bases de datos componentes. Además, como hemos visto, usando la técnica de Lenguaje natural Acotado permitimos que el usuario pueda realizar consultas por contenido de manera muy intuitiva.

- **Identificación de las fuentes de datos**

  Aunque, como hemos dicho, tanto el Sistema de Acceso Integrado a las tres Bibliotecas Digitales del Siglo de Oro, como las Interfaces de usuario propias de cada una de las Bibliotecas Digitales las hemos implementado nosotros, hemos diseñado un Gestor de la Presentación para nuestro Sistema de Acceso Integrado que crea Interfaces de Respuestas que permiten identificar las fuentes de datos de las que provienen las respuestas. Esta Interfaz de Respuesta detalla, como ya hemos explicado, las bases de datos de las que se ha obtenido respuesta a la consulta del usuario e incorpora links a sus propias Interfaces de Respuesta para que el usuario pueda navegar a través de la información y los documentos recuperados.

  De esta forma, cualquier Sistema de Acceso Integrado basado en nuestra arquitectura no será visto como una competencia a las Bibliotecas Digitales que integre, ya que dará completa visibilidad de las fuentes de datos.

- **Conservación de duplicados**

  Lógicamente, al identificar las fuentes de datos y permitir el acceso individual a los documentos recuperados en cada una de ellas, conservamos los posibles duplicados que puedan existir en las respuestas a las consultas del usuario.

Consideramos, que la arquitectura que hemos presentado es una solución elegante para facilitar la integración de Bibliotecas Digitales en un sistema escalable, fácil de adaptar a cambios y que incorpora una Interfaz de Usuario intuitiva, fácil de usar y que además es capaz de adaptarse al nivel de conocimiento que el usuario tenga de las fuentes de datos, permitiendo consultas complejas, para usuarios expertos, y consultas muy simples, para usuarios generales.

9.3 Líneas de trabajo futuro

Cada una de las líneas de trabajo abordados en esta tesis ha dado lugar a líneas de investigación futuras.

En cuanto a la línea de trabajo sobre de Interfaces de Usuario amigables, tenemos previsto construir una herramienta que facilite el diseño de imágenes para utilizarlas como Metáforas Cognitivas. Además, estamos estudiando modificar el algoritmo de Generación de la Interfaz de Consulta para permitir varias frases en Lenguaje Natural Acotado se presenten al mismo tiempo, haciendo así la interfaz un poco más amigable y de aspecto más comprensible.

Por otro lado, el diseño de la Arquitectura de Acceso Integrado ha abierto también nuevas líneas de investigación. En concreto, tenemos como objetivo el decidir el sistema adecuado de representación de ontologías, y construir el Generador de Árboles de Conceptos a partir de dicha representación ontológica, tal y como ya se indicó en el apartado 8.2.

Además, actualmente, nuestro Laboratorio está encargado de preparar una propuesta y un presupuesto para construir el "Portal Multilingüe de Acceso Integrado a las Bibliotecas Digitales de Literatura Emblemática Europea". Este es un proyecto que la Society for Emblem Studies pretende emprender en un futuro inmediato, de hecho es uno de los objetivos principales en su próxima reunión en septiembre de 2003, a la que nos han invitado expresamente para estudiar la posibilidad de que nos encarguemos del desarrollo de dicho sistema.

Esto es debido a que nuestra Biblioteca Digital de Literatura Emblemática es, en estos momentos, la biblioteca de referencia de lo que debe ser una Biblioteca Digital de este tipo de literatura. De hecho, nuestra Biblioteca Digital de Literatura Emblemática es la más accedida del mundo, por encima de cualquiera de las Bibliotecas Digitales de Literatura Emblemática inglesa, francesa, italiana o alemana, gracias a la amigabilidad y claridad del diseño de su Interfaz de Usuario y a la robustez de su funcionamiento, además de, por supuesto, a la riqueza de sus contenidos.

Si este proyecto sale finalmente adelante, nos daría la ocasión de probar nuestra arquitectura multilingüe con un gran número de Bibliotecas Digitales de gran calidad e interés científico, que son por supuesto heterogéneas no sólo en idioma sino en cuanto al conjunto de datos y su formato, lo que culminaría nuestro trabajo de investigación en Arquitecturas de Acceso integrado e Interfaces de Usuario.

Bibliografía

10.1 Referencias

1. Abelló, A., Oliva, Rodríguez, E. , Saltor, F. The BLOOM Model Revisited: An Evolution Proposal. In: ECOOP Workshop (Proc. ECOOP Workshops & posters, ECOOP'99. Lisbon, 1999).
2. Abelló, A., Oliva, Rodríguez, E. , Saltor, F. The syntax of BLOOM99 schemas. Report LSI-99-34-R, Dept LSI, UPC, Barcelona 1999.
3. Abelló, A., Oliva, M., Samos, J., Saltor. Information System Architecture for Data Warehousing from a Federation. In Int. Workshop on Engineering Federated Information Systems (EFIS'2000). Dublin (Ireland), 2000.
4. Abiteboul, S., Buneman, P., Suciu, D. Data on the Web. From Relations to Semistructured Data and XML, Morgan Kaufmann Publishers, 2000.
5. Altavista. http://www.altavista.com.
6. Arens, Y., Hsu, C., Knoblock, C. A. Query processing in the SIMS Information Mediator. *Advanced Planning Technology*, Austin Tate (Ed.), AAAI Press pp. 61-69, Menlo Park, CA, 1996.
7. Baeza-Yates, R.; Navarro, G. Integrating contents and structure in text retrieval. ACM SIGMOD Record, 25(1):67-79, Marzo 1996.
8. Baeza-Yates, R.; Navarro, G.; Vegas, J.; Fuente, P. A model and a visual query language for structured text. En Berthier Ribeiro-Neto (Eds.) Proc. of the 5th Symposium on String Processing and Information Retrieval, páginas 7-13, Santa Cruz, Bolivia, Sept 1998. IEEE CS Press.
9. Baeza-Yates, R.; Ribeiro-Neto, B. Modern Information Retrieval, Addison-Wesley, 1999.
10. Bell, D.; Grimson, J. Distributed Database Systems. Addison-Wesley, 1992.
11. Bertino y otros. Digital Libraries: Future Directions for a European Research Programme. *Braimstorming Report*. Alta Badia, Italia, Junio 2001.
12. Bib-1 Atribute Set. http://lcweb.loc.gov/z3950/agency/defns/bib1.html.
13. Biblioteca Virtual Galega. http://bvg.udc.es.
14. Brisaboa, N.R., Hernández, H. Iglesias, E.L., López J. R., Paramá, J.R., Penabad, M.R. Accesing a documental database through Internet. VI International Conference on

Extending Databases Technology (EDBT'98). Proceedings of Demo session *VI International Conference on Extending Databases Technology*.. Valencia, 1998.

15. Broekstra, J., Klein, M., Decker, S., Fensel, D., Horrocks, I. Adding formal semantics to the Web: Building on top of RDF Schema. In Proccedings of the ECDL Workshop on Semantic Web. 2000.
16. Busse, S., Kutsche, R.-D., Leser, U. Strategies for the Conceptual Design of Federated Information Systems In M. Roantree, W. Hasselbring, and S. Conrad (eds.), *Engineering Federated Information Systems, Proc. of the 3rd Workshop EFIS 2000*, pp. 23-32. Infix, June 2000.
17. Busse, S., Kutsche, R.-D., Leser, U., Weber H. Federated Information Systems: Concepts, Terminology and Architectures. Technical Report. Nr. 99-9, TU Berlin. April 1999.
18. Carey, M. y colaboradores. Towards Heterogeneous Multimedia Information Systems: The Garlic Approach. Actas del *Fifth International Workshop on Research Issues in Data Engineering(RIDE): Distributed Object Management.* 1995.
19. Chandrasekaran, B.; Josephson, R. What are ontologies, and why do we need them? In IEEE Inteligent systems, 1999.
20. Cui, Z., Jones, D., O'Brien, P. Issues in Ontology-based Information Integration. In Proceedings of the IJCAI-01 Workshop: Ontologies and Information Sharing, Vol. 47, pp. 141-146. Seatle, USA, Agosto 2001.
21. Dublin Core Metadata Iniciative. http://www.dublincore.org/.
22. Elmasri, R. WebOntEx: Extracting Ontologies from Web Pages. Invited Talk in V Jornadas de Ingeniería del Software y Bases de Datos (JISBD'2000). Valladolid, November 2000.
23. Erdmann, M., Studer, R. How to structure and access XML documents with ontologies. DKE 36(3): 317-335, 2001.
24. Escalona, M.J., Mejías, M., Torres, J. Getting requirements in web information sytems. Proceedings of the 14th International Conference of Software and Systems Engineering and their applications. París, Francia, 2001.
25. Escalona, M.J., Mejías, M., Torres, J., Reina, A. Definición de requisitos de interacción. II Jornadas Dolmen. Actas de las II Jornadas Dolmen. Valencia, 2002.
26. Escalona, M.J., Martín, A., Martínez, D. Un tesauro de patrimonio histórico en Internet mediante ASP. Propuesta al centro de documentación del IAPH. Boletín del Instituto Andaluz de Patrimonio Histórico. Volumen: 3/2002. Sevilla, 2002.
27. Escalona, M. J., Mejías, M., Torres, J., Reina, A. NDT: Una Técnica para el Desarrollo de la Navegación. Actas del 5º Workshop Iberoamericano de Ingeniería de Requisitos y Ambientes Software (Ideas'2002), pp. 305-315. La Habana, Cuba, 2002.
28. Euromosaic: The production and reproduction of the minority language groups in the European Union, ISBN 92-827-5512-6. Luxembourg (1996).
29. Fox, E. et al. Guía de Tesis y Disertaciones Electrónicas. http://www.etdguide.bibliored.cl/guide/
30. Frakes, W. Baeza-Yates, R. (Eds.) Information Retrieval. Data Structures & Algorithms. Prentice-Hall, 1992.
31. Gonçalves, M. A., France, R. K., Fox, E. A., Doszkocs, T. E. MARIAN Serching and Querying across Heterogeneous Federated Digital Libraries. *DELOS Workshop: Information Seeking, Searching and Querying in Digital Libraries 2000*. 2000.
32. Government Information Locator Service. http://www.gils.net/.
33. Google. http://www.google.com

34. Gruber, T. Toward Principles for the Design of Ontologies Used for Knowledge Sharing. IJHCS, 43 (5/6): 907-928. 1994.

35. Gruber, T. http://www-ksl.stanford.edu/kst/what-is-an-ontology.html

36. Guarino, N. (ed.), Formal Ontology in Information Systems. Proceedings of FOIS'98. Amsterdam, IOS Press, pp. 3-15. , Trento, Italy, 6-8 June 1998.

37. Hasselbring, W.; van den Heuvel, W.-J.; Houben, G.J.; Kutsche, R.-D.; Rieger, B.; Roantree, M.; Subieta, K. Research and Practice in Federated Information Systems. Report of the EFIS'2000 International Workshop. ACM SIGMOD RECORD Web Edition. Volumen 29, Número 4. December 2000.

38. ISO/IEC 9075:1999, Information Technology-Database Languages-SQL, The International Organization for Standardization, 1999.

39. Kirk,T., Levy, A. Y., Sagiv, Y., Srivastava, D. The Information Manifold. In Proc. of the AAAI Spring Symposium on Information Gathering in Distributed Heterogeneous Environments, pp. 85-91, 1995.

40. Klein M., Fensel D., Harmelen F., and Horrocks, I. The Relation between Ontologies and Schema-Languages: Translating OIL-Specifications to XML-Schema. Proceedings of the Workshop on Applications of Ontologies and Problem-solving Methods, 14th European Conference on Artificial Intelligence ECAI'00, Berlin, Germany, August 2000.

41. Laboratorio de Bases de Datos. http://emilia.dc.fi.udc.es/labBD/

42. Literatura Emblemática Hispánica. http://rosalia.dc.fi.udc.es/cicyt.

43. Lucas, W. Search engines, relevancy, and the World wide Web. En Goyal, A. (ed.), Text Databases & Document Management: Theory & Practice. Idea Group Publishing, 2001.

44. Marc Standards. http://www.loc.gov/marc/.

45. Marcondes, C. H., Sayão, L. F. Documentos digitais e novas formas de cooperação entre sistemas de informação. En Reunión Internacional de Especialistas en Información Científica Digital. Tema: Ciencias de la Información & Computación. *Anais. São Paulo: BIREME/OPS/OMS y la UNESCO.* São Paulo, 2002.

46. Mena, E., Illarramendi, A., Kashyap, V., Sheth, A. OBSERVER: An Approach for Query Processing in Global Information Systems based on Interoperation across Pre-existing Ontologies. Published in the journal Distributed And Parallel Databases (DAPD). 1998.

47. Microsoft Corporation. http://www.microsoft.com.

48. Norman, D. A. The Pshichology of EverydayThings. Basic Books, Inc. 1993.

49. Open Archives Iniciative. http://www.openarchives.org.

50. Ozsu, M.T.; Valduriez, P. Distributed Databases: Principles and Systems. Prentice Hall, 1991.

51. Powell, J., Fox, E. Multilingual Federated Searching Across Heterogeneous Collections, D-Lib Magazine, 1998. http://www.dlib.org/dlib/september98/powell/09powell.html.

52. Ramachandran, V. Design Patterns for Building Flexible and Maintainable J2EE Applications. http://developer.java.sun.com/developer/technicalArticles/J2EE/despat.

53. Relaciones de Sucesos. http://rosalia.dc.fi.udc.es/Relaciones.

54. Troll, D., Moen, Bill. Report to the DLF on the Z39.50 Implementers' Group. DLF: 2001. http://www.diglib.org/architectures/zig0012.htm.

55. Samos, J., Abelló, A., Oliva, M. Rodríguez, E., Saltor, F., Sistac, J. Araque F., Desgado, C., Garví, E., Ruiz, E. Sistema Cooperativo para la Integración de fuentes Heterogéneas de Información y Almacenes de Datos. Novatica ATI, 1999.

56. *Scientific and Technical Attribute Set.* http://www.cas.org/stas.html.

57. Sheth, A. P., Larson, J. A. Federated databases for managin distributed, heterogeneous, and autonomous databases, Computing Surveys 22:3 (1990), pp. 183-236.

58. Sociedad Española de Emblemática. http://rosalia.dc.fi.udc.es/sociedad/.

59. Society for Emblem Studies (SES). http://www.emblems.arts.gla.ac.uk/SES/.

60. Stuckenschmidt, H., Wache, H. Context modelling and transformation for semantic interoperability, In *Knowledge Representation Meets Databases (KRDB'2000)*. 2000.

61. Subrahmanian, V.S., Adali, S., Brink, A., Emery, R.,. Lu, J., Rajput, A., Rogers, T., Ross, R., Ward. C. HERMES: A heterogeneous reasoning and mediator system. Technical report, University of Maryland, 1995.

62. Sudarshan Chawathe, Hector Garcia-Molina, Joachim Hammer, Kelly Ireland, Yannis Papakonstantinou, Jeffrey Ullman, and Jennifer Widom. The TSIMMIS project: Integration of heterogenous information sources. 16th Meeting of the Information Processing Society of Japan, pp. 7-18, Tokyo, Japan, October 1994.

63. The Bath Profile. http://www.ukoln.ac.uk/interop-focus/bath/.

64. Yahoo!. http://www.yahoo.com.

65. Wache, H., Vögele, T., Visser, U., Stuckenschmidt, H., Schuster, G., Neumann, H. and Hübner S. Ontology-Based Integration of Information - A Survey of Existing Approaches. In Proceedings of the IJCAI-01 Workshop: Ontologies and Information Sharing, pp. 108-118. Seatle, USA, Agosto 2001.

66. World Wide Web Consortium. Standard XML http://www.w3.org/XML.

67. Zetie, C. Practical user interface design: Making GUIs work. McGraw Hill, 1995.

10.2 Publicaciones de la doctoranda derivadas de la investigación realizada en esta tesis

10.2.1 Publicaciones sobre Interfaces de Usuario

Revistas y Congresos Internacionales de reconocido prestigio

68. Brisaboa, N.R., Durán, M.J., Lalín, C., López, J.R., Places, A.S. Interfaz de Consulta a una base de datos de Literatura Emblemática a través de Internet. Emblemata Aurea: La Emblemática en el arte y la Literatura del Siglo de Oro. Rafael Zafra y José Javier Azanza (eds.), pp. 79-90. Pamplona, 1999.

69. Brisaboa, N.R., Durán, M.J., Penabad, M.R., y Places, A. S. A Collaborative Framework for a Digital Library. Proceedings of the Sixth International Workshop on Groupware (CRIWG'2000). IEEE Computer Society Press. ISBN 0-7695-0828-6. Isla Madeira, Portugal, 2000.

70. Brisaboa, N.R., Iglesias, E.L., Places, A. S. Bases de Datos Documentales. Nuevos Desafíos en la Web. *Ingeniería del Software y Bases de Datos. Tendencias Actuales*. Ediciones de la Universidad de Castilla - La Mancha, ISBN:84-8427-077-7, pp. 133-156. Cuenca, 2000.

71. Brisaboa, N. R., Penabad, M. R., Places, A. S., Rodríguez, F. J. A documental Database Query Language. *IEEE Computer Society PR01192* (SPIRE'01), pp. 242-245. Laguna de San Rafael, Chile, 2001.

72. Brisaboa, N.R., Penabad, M.R., Places, A.S., Rodríguez, F.J. A Document Database Query Language. *Lecture Notes in Computer Science (LNCS 2405), Springer Verlag* (BNCOD'02), pp. 183-198. Sheffield, England. Julio 2002.

73. Brisaboa, N. R., Penabad, M. R., Places, A. S., Rodríguez, F. J., Pérez-Sanjulián, C.F., Tato-Fontaiña, L., Lourenço-Modia, C., Vizcaíno-Fernández, C. BVG: Una Biblioteca Virtual de Literatura Gallega. *Mercator Media Forum*, Vol. 6, pp. 62-79. 2002.

74. Brisaboa, N.R.; Penabad, M.R.; Places, A.S.; Rodríguez, F.J.. Problems and Solutions to federate Digital Libraries. *Poster en 5th European Conf. on Research and Advanced Technology for Digital Libraries* (ECDL'2001). Darmstadt, Alemania, 2001.

75. Brisaboa, N. R., Penabad, M. R., Places, A.S., Rodríguez, F. J. Tools for the design of user friendly Web applications. *Lecture Notes in Computer Science (LNCS 2115), Springer Verlag* (EC-WEB'2001), pp. 29-38. Munich, Alemania 2001.

Otros Congresos Internacionales

76. Brisaboa, N.R., Durán, M.J., Lalín, C., López, J.R., Paramá, J.R., Penabad, M.R., Places, A.S. Using Bounded Natural to Query Databases on the Web. Proceedings of the *Information Systems, Analysis and Synthesis* (ISAS'99), pp. 518-522. Orlando, Florida, Agosto 1999.

77. Brisaboa, N.R., Durán, M.J., Penabad, M.R., y Places, A. S. Entorno Colaborativo para una Biblioteca Digital. *Memorias del VII Congreso Internacional de Investigación en Ciencias Computacionales* (CIICC'00), ISBN 970-18-5410-1. México, 2000.

78. Brisaboa, N.R., Penabad, M.R., Places, A.S., Rodríguez, F.J. A Database Query Technique for Text Retrieval. *Memorias del 8º Congreso Internacional de Investigación en Ciencias Computacionales* (CIICC'01). Colima, México, 2001.

79. Brisaboa, N. R., Ocaña, E., Penabad, M. R., Places, A. S., Rodríguez, F. J. Biblioteca Virtual de Literatura Gallega. In *Proceedings of the 5th Workshop Iberoamericano de Ingeniería de Requisitos y Ambientes Software* (IDEAS'2002), pp. 68-77, La Habana, Cuba, 2002.

Congresos Nacionales

80. Amil, C., Brisaboa, N.R., Cotelo-Lema, J.A., Fariña, A., Luaces, M.R., Penabad, M.R., Places, A.S., Viqueira, J.R. Una Interfaz Web para un Sistema Geográfico de Información Turística. Memorias de las *II Jornadas de Sistemas de Información Geográficos* (JSIG'02), pp. 173-175. El Escorial, Madrid, 2002.

81. Brisaboa, N.R., Escalona, M.J., Mejías, M., Places, A.S., Rodríguez, F.J., Torres, J. Sistema de Consulta vía Web para el Instituto Andaluz de Patrimonio Histórico. *Actas de las II Jornadas de Bibliotecas Digitales* (JBIDI'2001), pp. 99-116. Almagro, España, Noviembre 2001.

82. Brisaboa, N. R., Paramá, J. R., Penabad, M. R., Places, A. S., Rodríguez. BVG. La Biblioteca Virtual Gallega. *III Jornadas de Bibliotecas Digitales* (JBIBI'2002), pp. 163-172. El Escorial, Madrid, 2002.

83. Brisaboa, N.R., Places, A.S., Rodríguez, F.J. Biblioteca Virtual de Literatura Emblemática de la Universidade da Coruña. Memorias del IV Congreso Internacional de la Sociedad Española de Emblemática. Aceptado y pendiente de publicación. Palma de Mallorca, 2001.

10.2.2 Publicaciones sobre Integración

Revistas y Congresos Internacionales de reconocido Prestigio

84. Brisaboa, N.R., Paramá, J.R., Penabad, M.R., Places, A.S., Rodríguez, F.J. Solving Language Problems in a Multilingual Digital Library Federation. *Lecture Notes in Computer Science (LNCS 2510), Springer Verlag* (EURASIA-ICT'2002), pp. 503-510. Teheran, Irán, Octubre 2002.

85. Brisaboa, N.R., Penabad, M.R., Places, A.S. Especificación del Portal Multilingüe de Acceso Integrado a las Bibliotecas Digitales de Literatura Emblemática Europea. Working conference (Arbeitsgespräch) at the Herzog August Bibliothek, Wolfenbüttel (Alemania), 11-13 Septiembre 2003.

86. Brisaboa, N.R., Penabad, M.R., Places, A.S, Rodríguez, F.J. Ontologías en Federación de Bases de Datos. *XML: ¿el ASCII del siglo XXI? (Núm. 158), Novática (ISSN 0211-2124)*, pp. 45-53. Julio 2002.

Otros Congresos Internacionales

87. Brisaboa, N.R., Durán, M.J., Iglesias, E.L., López, J.R., Paramá, J.R., Penabad, M.R., Places, A.S. Integrating the access to documental databases on the web. Proceedings of the Fifth World Conference on Integrated Design and Process Technology (IDPT'2000). Dallas, Texas, Junio 2000.

88. Brisaboa, N.R., Durán, M.J., Lalín, C., López, J.R., Paramá, J.R., Penabad, M. R., Places, A.S. An Architecture for Virtual Libraries. Proceedings of the *Information Systems, Analysis and Synthesis* (ISAS'99). Orlando, Florida, Agosto 1999. pp. 512-517.

89. Brisaboa, N.R., Durán, M.J.; Lalín, C., López, J.R.; Penabad, M.R., y Places, A.S. Arquitectura para Bibliotecas Virtuales. *Memorias del VI Congreso Internacional de Investigación en Ciencias Computacionales* (CIICC'99), pp. 12-23. Cancún, México, Septiembre 1999.

90. Brisaboa, N.R., Penabad, M.R., Places, A.S, Rodríguez, F.J. An Architectural Proposal for Digital Libraries Federation. *Actas del 4th Encuentro Internacional de Computación* (ENC'01), pp. 695-704. Aguascalientes, México, Septiembre 2001.

91. Brisaboa, N.R., Penabad, M. R., Places, A.S., Rodríguez, F. J.: Using ontologies for federation of Web accesibles databases. Proceedings of the 13th International Conference on Software Engineering & Knowledge Engineering (SEKE'2001), pp. 87-94. Buenos Aires, Argentina, Julio 2001.

92. Brisaboa, N.R., Places, Á.S., Pérez-Sanjulián, C.F., Rodríguez, F.J. An architectural proposal for a cross-language system to federate multilingual digital libraries. In proceedings of the *Third All-Russian Scientific Conference "Digital Libraries: promising Advanced Methods, Techniques and Technologies, electronic Digital Collections"* (RCDL'2001), ISBN 5-9274-0055-8, Petrozavodsk. Rusia, 2001.

93. Brisaboa, N.R.; Places, A.S.; Rodríguez, F. J. Arquitectura para Federación de Bases de Datos Documentales Basada en Ontologías. *Memorias de las 4ª Jornadas Iberoamericanas de Ingeniería de Requisitos y Ambientes de* Software (IDEAS'2001), pp. 252-262. Heredia, Costa Rica, Abril 2001.

Congresos Nacionales

94. Brisaboa, N.R., Durán, M.J., Lalín, C., López, J.R., Paramá, J.R., Penabad, M.R., Places, A.S. Propuesta de Arquitectura para un Sistema de Bases de Datos Documentales. *Actas de las IV Jornadas de Ingeniería del Software y Bases de Datos* (JISBD'99), pp. 85-96. Cáceres, Noviembre 1999.

95. Brisaboa, N. R., Penabad, M.R., Places, A. S. Arquitectura para Federación de Bases de Datos Documentales del Siglo de Oro español. *Taller sobre Integración Semántica de Fuentes de Datos Distribuidas y Heterogéneas* (REDBD'2002). El Escorial, Madrid, 2002.

10.3 Publicaciones de la doctoranda no relacionadas con esta tesis

96. Brisaboa, N. R., Callón, C., López, Juan-Ramón, Places, A. S., Sanmartín, G. Stemming Galician Texts. *Lecture Notes in Computer Science (LNCS 2476),. Springer-Verlag* (SPIRE'2002), pp. 91-97. Lisboa, Portugal, 2002.

97. Brisaboa, N.R., López, J.R., Penabad, M.R., Places, A.S. Diachronic Stemmed Corpus and Dictionary of Galician Language. *Lecture Notes in Computer Science (LNCS 2588), Springer-Verlag* (CICLing'2003), pp. 412-417. México, 2003.

98. Brisaboa, N.R., Paramá, J.R., Penabad, M.R. y Places, A.S. Integración de bases de datos en un Sistema de Información Geográfico. Fundación DINTEL. ISBN: 84-931933-1-3, Vol. 2., pp. 161-175. Madrid, 2000.

99. Iglesias, E.L., Brisaboa, N.R., Penabad, M.R., Places, A.S., Rodríguez, F.J.: A Digital Library Application Generator. *Proceedings of the 1st International Workshop on New Developments in Digital Libraries (NDDL'2001)*, pp. 67-78. Setúbal, Portugal (2001)

100. Paramá, J.R., Brisaboa, N.R., Penabad, M.R., Places, A.S. A Semantic Query Optimization Approach to Optimize Linear Datalog Programs. *Lecture Notes in Computer Science (LNCS 2435), Springer-Verlag* (ADBIS'2002), pp. 277-290. Bratislava, Slovakia, 2002.

101. Penabad, M.R., Fariña, A., Brisaboa, N.R., Paramá, J.R., Places, A.S. Recuperación de textos en sistemas de gestión de bases de datos actuales. CUORE, Círculo de Usuarios Oracle de España. Num. 21. Valencia, 2002.

Apéndice I

Árbol de Conceptos del Sistema

1.1 Introducción

En este anexo se describe el DTD de los Árboles de Conceptos, así como el documento XML que almacena el Árbol de Conceptos de nuestro Sistema de Acceso Integrado.

Se ha usado XML para definirlo por varias razones, algunas de las cuales son las siguientes:

- Los documentos XML se autodescriben. A diferencia de los registros en sistemas de bases de datos tradicionales, los documentos XML no requieren un esquema relacional, tablas de descripción de ficheros, definiciones externas de tipos de datos, etc., porque los propios documentos contienen esta información.
- Tiene la capacidad de representar el esquema de cualquier base de datos de cualquier SGBD o formatos de ficheros tradicionales: Los documentos XML pueden contener cualquier tipo de dato, desde los tipos clásicos como texto y numéricos, hasta objetos multimedia como imágenes y sonidos, o formatos activos como applets de Java o componentes ActiveX.
- Las modificaciones en la presentación de los datos no necesitan ningún tipo de reprogramación. Es posible cambiar el aspecto de los documentos con hojas de estilo XSL sin manipular los datos en absoluto.
- XML soporta documentos multilingües y el estándar Unicode.
- Además, XML es abierto y extensible

1.2 DTD de los Árboles de Conceptos

En la Tabla 15 se muestra el DTD completo de los Árboles de Conceptos del sistema. En este apartado describiremos dicho DTD sentencia a sentencia.

Tabla 15. DTD de los Árboles de Conceptos

```
<!ELEMENT arbol-conceptos (concepto)>
<!ELEMENT concepto (IU?,((atributo+, has?, isa?) | (has, isa)))>
<!ATTLIST concepto nombre CDATA #REQUIRED>
<!ELEMENT atributo (IU, bd+, valor*)>
<!ATTLIST atributo nombre CDATA #REQUIRED>
<!ELEMENT IU EMPTY>
<!ATTLIST IU id CDATA #REQUIRED>
<!ELEMENT bd EMPTY>
<!ATTLIST bd id CDATA #REQUIRED>
<!ELEMENT valor EMPTY>
<!ATTLIST valor etiqueta CDATA #REQUIRED>
<!ELEMENT has (concepto+)>
<!ELEMENT isa (IU?, concepto+)>
```

Como se puede comprobar, se define un Árbol de Conceptos como un elemento `concepto`:

```
<!ELEMENT arbol-conceptos (concepto)>
```

Como ya hemos dicho, un concepto tiene atributos y Relaciones de Descripción y/o de Generalización / Especialización con otros conceptos. Además, cada concepto tiene asociado el identificador de la frase en Lenguaje Natural Acotado o la Metáfora Cognitiva (es decir, el elemento IU) desde la que se permitirá al usuario establecer condiciones sobre dicho concepto. El fragmento de DTD en el que define un concepto es el siguiente:

```
<!ELEMENT concepto (IU?, ((atributo+, has?, isa?) | (has, isa)))>
<!ATTLIST concepto nombre CDATA #REQUIRED>
```

El atributo `nombre` del elemento `concepto` almacena como es evidente, el nombre del concepto. Cada elemento `atributo` almacena un atributo del concepto. Un atributo se define como sigue:

```
<!ELEMENT atributo (IU, bd+, valor*)>
<!ATTLIST atributo nombre CDATA #REQUIRED>
```

Cada atributo del Árbol de Conceptos tiene asociada, por un lado, la Metáfora Cognitiva o la frase en Lenguaje Natural Acotado que se le mostrará al usuario para que pueda expresar restricciones sobre dicho atributo (elemento `IU`, de nuevo):

```
<!ELEMENT IU EMPTY>
<!ATTLIST IU id CDATA #REQUIRED>
```

Por otro lado, cada atributo de un Árbol de Conceptos tiene almacenada la lista de identificadores de bases de datos en las que dicho atributo existe. Cada uno de estos identificadores se almacena en un elemento `bd`. En cada elemento `atributo` existen tantos elementos `bd` como bases de datos tengan el atributo que representa. El fragmento de DTD en el que se define este elemento es el siguiente:

```
<!ELEMENT bd EMPTY>
<!ATTLIST bd id CDATA #REQUIRED>
```

Finalmente, para aquellos atributos de los Árboles de Conceptos que sólo pueden tomar un número de valores limitado, se almacena dicha lista de valores (elemento `valor`). De esta manera, podremos ofrecerle dichos valores al usuario en la interfaz para que elija uno de ellos de una lista desplegable. Cada uno de los posibles valores que pueda tomar se almacena en el atributo `etiqueta` de un elemento `valor`:

```
<!ELEMENT valor EMPTY>
<!ATTLIST valor etiqueta CDATA #REQUIRED>
```

Por su parte, los elementos `has` e `isa` identifican los dos tipos de relaciones que puede presentar un concepto (Relación de Descripción y Relación de Generalización / Especialización, respectivamente).

```
<!ELEMENT has (concepto+)>
<!ELEMENT isa (IU?, concepto+)>
```

El elemento `has` es opcional y, en caso de existir, estaría formado por uno o más elementos `concepto`. El elemento `has` permite relacionar un elemento `concepto` dado otros elementos del mismo tipo que contienen atributos que lo describen, es decir, representa una Relación de Descripción.

El elemento `isa` está a su vez formado por uno o más elementos `concepto`. Permite reflejar las Relaciones de Generalización / Especialización. Además el elemento `isa` puede tener un elemento `IU`, que almacena el identificador de la Metáfora Cognitiva o la Frase en Lenguaje Natural Acotado que se mostraría al usuario cuando fuese oportuno.

1.3 XML del Árbol de Conceptos de nuestro sistema

A continuación se presenta el Árbol de Conceptos de nuestro Sistema de Acceso Integrado, en su versión actual.

```
<?xml version="1.0" encoding="ISO-8859-1" standalone="no"?>
<!DOCTYPE arbol-conceptos SYSTEM "arbol-conceptos.DTD">
```

```
<arbol-conceptos>
<!-- Inicio Obra -->
   <concepto nombre="Obra">
      <atributo nombre="Título">
        <IU id="ske1-" />
        <bd id="1" /> <bd id="2" /> <bd id="3" />
      </atributo>
      <atributo nombre="Autor">
        <IU id="ske5-" />
        <bd id="1" /> <bd id="2" /> <bd id="3" />
      </atributo>
      <atributo nombre="Tema">
        <IU id="ske36-" />
        <bd id="1" /> <bd id="2" /> <bd id="3" />
      </atributo>
      <atributo nombre="Año">
        <IU id="ske10-" />
        <bd id="1" /> <bd id="2" /> <bd id="3" />
      </atributo>
      <atributo nombre="Lugar de edición">
        <IU id="ske9-" />
        <bd id="1" /> <bd id="2" /> <bd id="3" />
      </atributo>
      <isa>
<!-- Inicio de Libro de Emblemas (isa de Obra) -->
         <IU id="metaphor" />
         <concepto nombre="Libro de Emblemas">
            <atributo nombre="Título">
              <IU id="ske1-" />
              <bd id="1" /> <bd id="2" />
            </atributo>
            <atributo nombre="Autor">
              <IU id="ske5-" />
              <bd id="1" /> <bd id="2" />
            </atributo>
            <has>
<!-- Inicio de Edición (has de Libro de Emblemas) -->
               <concepto nombre="Edición">
                  <atributo nombre="Año">
                    <IU id="ske10-" />
                    <bd id="1" /> <bd id="2" />
                  </atributo>
                  <atributo nombre="Lugar">
                    <IU id="ske9-" />
                    <bd id="1" /> <bd id="2" />
                  </atributo>
                  <atributo nombre="Editor">
                    <IU id="ske7-" />
                    <bd id="1" /> <bd id="2" />
                  </atributo>
                  <atributo nombre="Impresor">
                    <IU id="ske8-" />
                    <bd id="1" /> <bd id="2" />
                  </atributo>
                  <atributo nombre="Promotor">
                    <IU id="ske6-" />
                    <bd id="1" /> <bd id="2" />
                  </atributo>
               </concepto>
```

```
<!-- Fin Edición (has de Libro de Emblemas) -->
<!-- Inicio Emblema (has de Libro de Emblemas) -->
                <concepto nombre="Emblema">
                   <atributo nombre="Palabras Clave">
                     <IU id="skel2-" />
                     <bd id="1" /> <bd id="2" />
                   </atributo>
                   <atributo nombre="Onomásticas">
                     <IU id="skel3-" />
                     <bd id="1" /> <bd id="2" />
                   </atributo>
                   <atributo nombre="Exempla">
                     <IU id="skel6-" />
                     <bd id="1" /> <bd id="2" />
                   </atributo>
                   <has>
<!-- Inicio Mote  (has de Emblema) -->
                      <concepto nombre="Mote">
                         <atributo nombre="Texto">
                           <IU id="ske18-" />
                           <bd id="1" /> <bd id="2" />
                         </atributo>
                         <atributo nombre="Tipo">
                           <IU id="ske17-" />
                           <bd id="1" /> <bd id="2" />
                         </atributo>
                      </concepto>
<!-- Fin Mote (has de Emblema) -->
<!-- Inicio Imagen ( has de Emblema) -->
                      <concepto nombre="Imagen">
                         <atributo nombre="Motivo">
                           <IU id="ske2-" />
                           <bd id="1" /> <bd id="2" />
                         </atributo>
                         <atributo nombre="Objeto">
                           <IU id="ske15-" />
                           <bd id="1" /> <bd id="2" />
                         </atributo>
                      </concepto>
<!-- Fin Imagen ( has de Emblema) -->
<!-- Inicio Epigrama ( has de Emblema) -->
                      <concepto nombre="Epigrama">
                    <!-- <IU id="seleccion" /> -->
                         <atributo nombre="Texto">
                           <IU id="ske3-" />
                           <bd id="1" /> <bd id="2" />
                         </atributo>
                         <atributo nombre="Estrofa">
                           <IU id="ske19-" />
                           <bd id="1" /> <bd id="2" />
                         </atributo>
                         <atributo nombre="Tipo de verso">
                           <IU id="ske21-" />
                           <bd id="1" /> <bd id="2" />
                         </atributo>
                         <atributo nombre="Número de versos">
                           <IU id="skel1-" />
                           <bd id="1" /> <bd id="2" />
                         </atributo>
                         <atributo nombre="Idioma">
```

```
                        <IU id="ske20-" />
                        <bd id="1" /> <bd id="2" />
                      </atributo>
                    </concepto>
<!-- Fin Epigrama ( has de Emblema) -->
<!-- Inicio Glosa ( has de Emblema) -->
                    <concepto nombre="Glosa">
                  <!-- <IU id="seleccion" /> -->
                      <atributo nombre="Resumen">
                        <IU id="ske4-" />
                        <bd id="1" /> <bd id="2" />
                      </atributo>
                    </concepto>
<!-- Fin Glosa ( has de Emblema) -->
<!-- Inicio Autoridad ( has de Emblema) -->
                    <concepto nombre="Autoridad">
                      <IU id="seleccion" />
                      <atributo nombre="Autoridad General">
                        <IU id="ske14-" />
                        <bd id="1" /> <bd id="2" />
                      </atributo>
                    </concepto>
<!-- Fin Autoridad ( has de Emblema) -->
                  </has>
                </concepto>
<!-- Fin Emblema (has de Libro de Emblemas) -->
            </has>
          </concepto>
<!-- Fin Libro de Emblemas (isa de Obra) -->

<!-- Inicio Relación de Sucesos (isa de Obra) -->
          <concepto nombre="Relación de Sucesos">
            <atributo nombre="Título">
              <IU id="ske22-" />
              <bd id="3" />
            </atributo>
            <atributo nombre="Autor">
              <IU id="ske23-" />
              <bd id="3" />
            </atributo>
            <atributo nombre="Lugar referido">
              <IU id="ske24-" />
              <bd id="3" />
            </atributo>
            <atributo nombre="Año del acontecimiento">
              <IU id="ske25-" />
              <bd id="3" />
            </atributo>
            <atributo nombre="Tipología">
              <IU id="ske26-" />
              <bd id="3" />
            </atributo>
            <atributo nombre="Subgénero">
              <IU id="ske27-" />
              <bd id="3" />
            </atributo>
            <atributo nombre="Modalidad del discurso">
              <IU id="ske28-" />
              <bd id="3" />
            </atributo>
```

```
            <has>
<!-- Inicio de Edición (has de Relación de Sucesos) -->
               <concepto nombre="Edición">
              <!-- <IU id="seleccion" /> -->
                   <atributo nombre="Año">
                     <IU id="ske29-" />
                     <bd id="3" />
                   </atributo>
                   <atributo nombre="Lugar">
                     <IU id="ske30-" />
                     <bd id="3" />
                   </atributo>
                   <atributo nombre="Editor">
                     <IU id="ske31-" />
                     <bd id="3" />
                   </atributo>
                   <atributo nombre="Impresor">
                     <IU id="ske32-" />
                     <bd id="3" />
                   </atributo>
                   <has>
<!-- Inicio Ejemplar (has de Edición) -->
                      <concepto nombre="Ejemplar">
                         <atributo nombre="Encuadernación">
                           <IU id="ske33-" />
                           <bd id="3" />
                         </atributo>
                         <atributo nombre="Exlibris">
                           <IU id="ske34-" />
                           <bd id="3" />
                         </atributo>
                         <has>
<!-- Inicio Biblioteca (has de Ejemplar) -->
                            <concepto nombre="Biblioteca">
                               <atributo nombre="Nombre">
                                 <IU id="ske35-" />
                                 <bd id="3" />
                               </atributo>
                            </concepto>
<!-- Fin Biblioteca (has de Ejemplar) -->
                         </has>
                      </concepto>
<!-- Fin Ejemplar (has de Edición) -->
                   </has>
               </concepto>
<!-- Fin de Edición (has de Relación de Sucesos)-->
            </has>
         </concepto>
<!-- Fin Relación de Sucesos (isa de Obra)-->
      </isa> <!—Eliminar cuando se inserte Emblemas Traducidos -->
<!--Emblemas Traducidos
      </isa>  -->
   </concepto>
<!-- Fin de Obra -->
</arbol-conceptos>
```

Apéndice II

Árboles de Correspondencias del Sistema

2.1 Introducción

En este anexo se describe el DTD de los Árboles de Correspondencias de nuestra arquitectura y se presentan los documentos XML de los Árboles de Correspondencias de dos de las tres bases de datos integradas en nuestro sistema.

2.2 DTD de los Árboles de Correspondencias

En el DTD de la Tabla 16 aparece definida la estructura de los Árboles de Correspondencias. La estructura de estos árboles es idéntica a la estructura de los Árboles de Conceptos. Por lo tanto, el documento XML de un Árbol de Correspondencias está compuesto por un elemento `concepto` que representa el concepto raíz del árbol. La diferencia entre ambos está en la información que está asociada a cada concepto y a cada atributo, así como, lógicamente, en el propósito para el que son usados ambos tipos de árboles.

Tabla 16. DTD de los Árboles de Correspondencias

```
<!ELEMENT ac (concepto)>
<!ELEMENT concepto (correspondencia,
                    ((atributo+, has?, isa?) | (has, isa)))>
<!ATTLIST concepto nombre CDATA #REQUIRED>
<!ELEMENT atributo (correspondencia, (valor,traduccion)*)>
<!ATTLIST atributo nombre CDATA #REQUIRED>
<!ELEMENT valor (#PCDATA)>
<!ELEMENT traduccion (#PCDATA)>
<!ELEMENT correspondencia (select?, from?, where*)>
<!ELEMENT select (#PCDATA)>
<!ELEMENT from (#PCDATA)>
<!ELEMENT where (fijo, parametro*, opcional?)>
```

```
<!ELEMENT fijo (#PCDATA)>
<!ELEMENT parametro EMPTY>
<!ATTLIST parametro etiqueta CDATA #REQUIRED>
<!ELEMENT opcional (#PCDATA)>
<!ATTLIST opcional fin CDATA #REQUIRED>
<!ELEMENT has (concepto+)>
<!ELEMENT isa (concepto+)>
```

Observando los DTDs de los Árboles de Conceptos y los Árboles de Correspondencias (Tabla 15 y Tabla 16, respectivamente), se puede comprobar, como ya hemos comentado, son idénticos en cuanto a su estructura y diferentes en cuanto a la información que almacenan. Esta información se recoge en el elemento `correspondencia`, que está asociado a los elementos `concepto` y a los elementos `atributo`.

El elemento `correspondencia` está formado por los elementos `select`, `from` y `where`, tal y como se muestra en el DTD:

```
<!ELEMENT correspondencia (select?, from?, where*)>
```

El elemento `select` es opcional. La cadena de caracteres que se almacena en este elemento se incorporará a la cláusula select de la consulta SQL final durante la traducción.

El elemento `from` es también un elemento opcional en el que se almacena el fragmento de la sentencia que se incorporará a la cláusula from de la sentencia SQL final.

El elemento `where` es también opcional y almacena el fragmento de la sentencia que se incorporará a la cláusula `where` de la sentencia SQL final. El fragmento de DTD correspondiente a la cláusula where es el siguiente:

```
<!ELEMENT where (fijo, parametro*, opcional?)>
```

Los elementos que forman un where son:

- `fijo`: Esta cadena de caracteres contiene ciertas variables, que dependen de los valores de las condiciones introducidas por el usuario. Estas variables están marcadas por el caracter "#".
- `parametro`: lista de variables que aparecen en el elemento `fijo` y que se deben sustituir por los valores concretos que contengan estos elementos `parametro` en la consulta en XML.
- `opcional`: Existe para optimizar las consultas SQL finales. Se usa en los casos en los que es necesario expresar varias condiciones sobre el mismo atributo. El elemento `opcional` contiene la cadena de caracteres de dicha subconsulta que es necesario repetir en el supuesto de que existan

varias condiciones sobre el atributo al que permite acceder. El elemento `opcional` tiene además un atributo llamado `fin`. La cadena de caracteres almacenada en este atributo se añadirá a la cláusula `where` final después de incluir la cadena almacenada en el elemento `opcional` tantas veces como sea necesario. Permite cerrar la subconsulta que irá dentro de la cláusula `where` general.

2.3 XML de los Árboles de Correspondencias

Se presentan en este apartado los documentos XML de los Árboles de Correspondencias de las bases de datos Libros de Emblemas y Relaciones de Sucesos, en su estado actual.

2.3.1 Base de datos de Libros de Emblemas

```
<?xml version="1.0" encoding="ISO-8859-1" standalone="no"?>
<!DOCTYPE ac SYSTEM "ac.DTD">

<ac>
<!-- Inicio concepto Obra -->
  <concepto nombre="Obra">
    <correspondencia>
        <select> cod_obra, cod_emblem </select>
        <from> obras ob, emblema em </from>
        <where><fijo> ob.cod_obra = em.cod_obra </fijo></where>
    </correspondencia>
    <atributo nombre="Título">
      <correspondencia>
        <where>
          <fijo> and ob.titulo like '%#valor#%' </fijo>
          <opcional fin=""> and ob.titulo like '%#valor#%' </opcional>
        </where>
      </correspondencia>
    </atributo>
    <atributo nombre="Autor">
      <correspondencia>
        <where>
          <fijo> ob.cod_obra = em.cod_obra
                 and ob.autor like '%#valor#%' </fijo>
        </where>
      </correspondencia>
    </atributo>
    <atributo nombre="Tema">
      <correspondencia>
        <where>
          <fijo> and #limite# &lt;= (select count(*)
                 from clave cl
                 where em.cod_obra = cl.cod_obra
                 and em.cod_emblem = cl.cod_emblem
                 and clave in ('#valor#' </fijo>
          <parametro nombre="limite"/>
```

```
          <opcional fin="))">, '#valor#' </opcional>
        </where>
      </correspondencia>
    </atributo>
    <atributo nombre="Año">
      <correspondencia>
        <from> edicion ed </from>
        <where>
          <fijo> and ed.cod_edic = ob.cod_edic
                 and (ed.cod_edic #op# '#valor#' </fijo>
          <parametro nombre="op"/>
          <opcional fin=")"> and ed.cod_edic &lt;= '#valor#' </opcional>
        </where>
      </correspondencia>
    </atributo>
    <atributo nombre="Lugar de edición">
      <correspondencia>
        <from> edicion ed </from>
        <where>
          <fijo> and ed.cod_edic = ob.cod_edic
                 and ed.lugar like '%#valor#%' </fijo>
        </where>
      </correspondencia>
    </atributo>
    <isa>

<!-- Inicio de Libro de Emblemas (isa de Obra) -->
      <concepto nombre="Libro de Emblemas">
        <correspondencia>
          <select> cod_obra, cod_emblem </select>
          <from> obras ob, emblema em </from>
          <where><fijo> ob.cod_obra = em.cod_obra </fijo></where>
        </correspondencia>
        <atributo nombre="Título">
          <correspondencia>
            <where>
              <fijo> ob.titulo like '%#valor#%' </fijo>
              <opcional   fin="">   and   ob.titulo   like   '%#valor#%'
</opcional>
            </where>
          </correspondencia>
        </atributo>
        <atributo nombre="Autor">
          <correspondencia>
            <where><fijo> ob.autor like '%#valor#%' </fijo></where>
          </correspondencia>
        </atributo>
        <has>

<!-- Inicio de Edición (has de Libro de Emblemas) -->
          <concepto nombre="Edición">
            <correspondencia>
              <select> </select>
              <from> edicion ed </from>
              <where><fijo> ed.cod_edic = ob.cod_edic </fijo></where>
            </correspondencia>
            <atributo nombre="Año">
              <correspondencia>
                <where>
                  <fijo> (ed.cod_edic #op# '#valor#' </fijo>
```

```
                <parametro nombre="op"/>
                <opcional  fin=")"> and  ed.cod_edic  &lt;=  '#valor#'
</opcional>
              </where>
            </correspondencia>
          </atributo>
          <atributo nombre="Lugar">
            <correspondencia>
              <where><fijo> ed.lugar like '%#valor#%' </fijo></where>
            </correspondencia>
          </atributo>
          <atributo nombre="Editor">
            <correspondencia>
              <where><fijo> ed.editor like '%#valor#%' </fijo></where>
            </correspondencia>
          </atributo>
          <atributo nombre="Impresor">
            <correspondencia>
              <where><fijo>      ed.impresor      like      '%#valor#%'
</fijo></where>
            </correspondencia>
          </atributo>
          <atributo nombre="Promotor">
            <correspondencia>
              <where><fijo>      ed.promotor      like      '%#valor#%'
</fijo></where>
            </correspondencia>
          </atributo>

<!-- Fin de Edición (has de Libro de Emblemas) -->
        </concepto>

<!-- Inicio de Emblema (has de Libro de Emblemas) -->
        <concepto nombre="Emblema">
          <correspondencia>
            <select> </select>
            <from> </from>
            <where><fijo> </fijo></where>
          </correspondencia>
          <atributo nombre="Palabras Clave">
            <correspondencia>
              <where>
                <fijo> #limite# &lt;= (select count(*)
                       from clave cl
                       where em.cod_obra = cl.cod_obra
                       and em.cod_emblem = cl.cod_emblem
                       and clave in ('#valor#' </fijo>
                <parametro nombre="limite"/>
                <opcional fin="))">,'#valor#' </opcional>
              </where>
            </correspondencia>
          </atributo>
          <atributo nombre="Onomásticas">
            <correspondencia>
              <where>
                <fijo> #limite# &lt;= (select count(*)
                       from onomast ono
                       where ob.cod_obra = ono.cod_obra
                       and em.cod_emblem = ono.cod_emblem
                       and ono.nombre in ('#valor#' </fijo>
```

```
                    <parametro nombre="limite"/>
                    <opcional fin="))">,'#valor#' </opcional>
                  </where>
                </correspondencia>
              </atributo>
              <atributo nombre="Exempla">
                <correspondencia>
                  <where>
                    <fijo> cod_emblem in
                           (select cod_emblem
                           from exempla ex
                           where ex.cod_obra = ob.cod_obra
                           and ex.cod_emblem = em.cod_emblem
                           and ex.exempla like '%#valor#%' </fijo>
                    <opcional  fin=")"> and  ex.exempla  like  '%#valor#%'
</opcional>
                  </where>
                </correspondencia>
              </atributo>
              <has>

<!-- Inicio de Mote (has de Emblema) -->
                <concepto nombre="Mote">
                  <correspondencia>
                    <select> </select>
                    <from> </from>
                    <where><fijo> </fijo></where>
                  </correspondencia>
                  <atributo nombre="Texto">
                    <correspondencia>
                      <where>
                        <fijo> cod_emblem in
                               (select cod_emblem
                               from mote mo
                               where mo.cod_obra = ob.cod_obra
                               and mo.cod_emblem = em.cod_emblem
                               and mo.mote like '%#valor#%' </fijo>
                        <opcional  fin=")"> and  mo.mote  like  '%#valor#%'
</opcional>
                      </where>
                    </correspondencia>
                  </atributo>
                  <atributo nombre="Tipo">
                    <correspondencia>
                      <where>
                        <fijo> cod_emblem in
                               (select cod_emblem
                               from mote mo
                               where mo.cod_obra = ob.cod_obra
                               and mo.cod_emblem = em.cod_emblem
                               and mo.tipo #op# '#valor#') </fijo>
                        <parametro nombre="op"/>
                      </where>
                    </correspondencia>
                  </atributo>
<!-- Fin de Mote (has de Emblema) -->
                </concepto>

<!-- Inicio de Imagen (has de Emblema) -->
                <concepto nombre="Imagen">
```

```
                <correspondencia>
                  <select> </select>
                  <from> </from>
                  <where><fijo> </fijo></where>
                </correspondencia>
                <atributo nombre="Motivo">
                  <correspondencia>
                    <where>
                      <fijo> cod_emblem in
                             (select cod_emblem
                             from im_ppal im
                             where im.cod_obra = ob.cod_obra
                             and im.cod_emblem = em.cod_emblem
                             and im.motivo like '%#valor#%' </fijo>
                        <opcional fin=")"> and im.motivo like '%#valor#%'
</opcional>
                    </where>
                  </correspondencia>
                </atributo>
                <atributo nombre="Objeto">
                  <correspondencia>
                    <where>
                      <fijo> #limite# &lt;=
                             (select count(*)
                             from obj_im obj
                             where ob.cod_obra = obj.cod_obra
                             and em.cod_emblem = obj.cod_emblem
                             and obj.objeto in ('#valor#' </fijo>
                        <parametro nombre="limite"/>
                        <opcional fin="))">,'#valor#' </opcional>
                    </where>
                  </correspondencia>
                </atributo>
<!-- Fin de Imagen (has de Emblema) -->
              </concepto>

<!-- Inicio de Epigrama (has de Emblema) -->
              <concepto nombre="Epigrama">
                <correspondencia>
                  <select> </select>
                  <from> </from>
                  <where><fijo> </fijo></where>
                </correspondencia>
                <atributo nombre="Texto">
                  <correspondencia>
                    <where>
                      <fijo> em.resu_epi like '%#valor#%' </fijo>
                      <opcional fin=""> and em.resu_epi like '%#valor#%'
</opcional>
                    </where>
                  </correspondencia>
                </atributo>
                <atributo nombre="Estrofa">
                  <correspondencia>
                    <where><fijo> em.estrofa = '#valor#' </fijo></where>
                  </correspondencia>
                </atributo>
                <atributo nombre="Tipo de verso">
                  <correspondencia>
                    <where>
```

```
                    <fijo> em.tipo_verso = '#valor#' </fijo>
                    <!--<parametro nombre ="op"/>  -->
                  </where>
                </correspondencia>
              </atributo>
              <atributo nombre="Número de versos">
                <correspondencia>
                  <where>
                    <fijo> (em.no_versos #op# '#valor#' </fijo>
                    <parametro nombre="op"/>
                    <opcional   fin=")">   and   em.no_versos   &lt;=
'#valor#' </opcional>
                  </where>
                  </correspondencia>
              </atributo>
              <atributo nombre="Idioma">
                <correspondencia>
                  <where>
                    <fijo> em.idioma_epi = '#valor#' </fijo>
                    <!-- <parametro nombre="op"/> -->
                  </where>
                </correspondencia>
              </atributo>
<!-- Fin de Epigrama (has de Emblema) -->
            </concepto>

<!-- Inicio de Glosa (has de Emblema) -->
            <concepto nombre="Glosa">
              <correspondencia>
                <select> </select>
                <from> </from>
                <where><fijo> </fijo></where>
              </correspondencia>
              <atributo nombre="Resumen">
                <correspondencia>
                  <where>
                    <fijo> em.resu_glosa like '%#valor#%' </fijo>
                    <opcional   fin="">   and   em.resu_glosa   like
'%#valor#%' </opcional>
                  </where>
                </correspondencia>
              </atributo>
<!-- Fin de Glosa (has de Emblema) -->
            </concepto>

<!-- Inicio de Autoridad (has de Emblema) -->
            <concepto nombre="Autoridad">
              <correspondencia>
                <select> </select>
                <from> </from>
                <where><fijo> </fijo></where>
              </correspondencia>
              <atributo nombre="Autoridad General">
                <correspondencia>
                  <where>
                    <fijo> #limite# &lt;=
                           (select count(*)
                           from autorida au
                           where ob.cod_obra = au.cod_obra
                           and em.cod_emblem = au.cod_emblem
```

```
                                and au.autgeneral like '%#valor#%') </fijo>
                        <parametro nombre="limite"/>
                        <!--<opcional  fin=")">  and  au.autgeneral  like
'%#valor#%' </opcional>-->
                    </where>
                  </correspondencia>
                </atributo>
<!-- Fin de Autoridad (has de Emblema) -->
              </concepto>

            </has>
<!-- Fin de Emblema (has de Libro de Emblemas)-->
          </concepto>
        </has>
<!-- Fin de Libro de Emblemas (isa de Obra)-->
      </concepto>
    </isa>

<!-- Fin de Obra -->
  </concepto>
</ac>
```

2.3.2 Base de datos de Relaciones de Sucesos

```
<?xml version="1.0" encoding="ISO-8859-1" standalone="no"?>
<!DOCTYPE ac SYSTEM "ac.DTD">

<ac>
<!-- Inicio de Obra -->
  <concepto etiqueta="Obra">
    <correspondencia>
      <select> tituloabre, cod_edic </select>
      <from> relacion rel, edicion ed </from>
      <where><fijo> rel.tituloabre = ed.tituloabre </fijo></where>
    </correspondencia>
    <atributo etiqueta="Título">
      <correspondencia>
        <where>
          <fijo> and rel.titulo like '%#valor#%' </fijo>
          <opcional fin=""> and rel.titulo like '%#valor#%' </opcional>
        </where>
      </correspondencia>
    </atributo>
    <atributo etiqueta="Autor">
      <correspondencia>
        <where>
          <fijo> and rel.autor like '%#valor#%' </fijo>
        </where>
      </correspondencia>
    </atributo>
    <atributo etiqueta="Tema">
      <correspondencia>
        <where>
          <fijo>contains (rel.titulo, '#valor# </fijo>
          <opcional fin = "', 10) &gt 0"> && #valor# </opcional>
        </where>
      </correspondencia>
    </atributo>
    <atributo etiqueta="Año">
```

```
      <correspondencia>
        <where>
          <fijo> and (rel.fecha_acon #op# '#valor#' </fijo>
          <parametro etiqueta="op"/>
          <opcional   fin=")">   and   rel.fecha_acon   &lt;=   '#valor#'
</opcional>
        </where>
      </correspondencia>
    </atributo>
    <atributo etiqueta="Lugar de edición">
      <correspondencia>
        <where>
          <fijo> and rel.lugar_ref like '%#valor#%' </fijo>
        </where>
      </correspondencia>
    </atributo>
    <isa>

<!-- Inicio de Relación de Sucesos (isa de Obra) -->
      <concepto etiqueta="Relación de Sucesos">
        <correspondencia>
          <select> tituloabre, cod_edic </select>
          <from> relacion rel, edicion ed </from>
          <where><fijo> rel.tituloabre = ed.tituloabre </fijo></where>
        </correspondencia>
        <atributo etiqueta="Título">
          <correspondencia>
            <where>
              <fijo> rel.titulo like '%#valor#%' </fijo>
              <opcional   fin="">   and   rel.titulo   like   '%#valor#%'
</opcional>
            </where>
          </correspondencia>
        </atributo>
        <atributo etiqueta="Autor">
          <correspondencia>
            <where><fijo> rel.autor like '%#valor#%' </fijo></where>
          </correspondencia>
        </atributo>
        <atributo etiqueta="Lugar referido">
          <correspondencia>
            <where><fijo> rel.lugar_ref like '%#valor#%' </fijo></where>
          </correspondencia>
        </atributo>
        <atributo etiqueta="Año del acontecimiento">
          <correspondencia>
            <where>
              <fijo> (rel.fecha_acon #op# '#valor#' </fijo>
              <parametro etiqueta="op"/>
              <opcional  fin=")">  and  rel.fecha_acon  &lt;=  '#valor#'
</opcional>
            </where>
          </correspondencia>
        </atributo>
        <atributo etiqueta="Tipología">
          <correspondencia>
            <where>
              <fijo> rel.tipologia = '#valor#' </fijo>
              <!-- <parametro etiqueta="op"/> -->
            </where>
```

```
          </correspondencia>
        </atributo>
        <atributo etiqueta="Subgénero">
          <correspondencia>
            <where><fijo> rel.subgenero = '#valor#' </fijo></where>
          </correspondencia>
        </atributo>
        <atributo etiqueta="Modalidad del discurso">
          <correspondencia>
            <where><fijo> rel.prosa_vers = '#valor#' </fijo></where>
          </correspondencia>
        </atributo>
        <has>

<!-- Inicio de Edición (has de Relación de Sucesos) -->
          <concepto etiqueta="Edición">
            <correspondencia>
              <where><fijo> </fijo></where>
            </correspondencia>
            <atributo etiqueta="Año">
              <correspondencia>
                <where>
                  <fijo> (ed.cod_edic #op# '#valor#' </fijo>
                  <parametro etiqueta="op"/>
                  <opcional  fin=")"> and ed.cod_edic &lt;= '#valor#'
</opcional>
                </where>
              </correspondencia>
            </atributo>
            <atributo etiqueta="Lugar">
              <correspondencia>
                <where><fijo> ed.lugar like '%#valor#%' </fijo></where>
              </correspondencia>
            </atributo>
            <atributo etiqueta="Editor">
              <correspondencia>
                <where><fijo> ed.editor like '%#valor#%' </fijo></where>
              </correspondencia>
            </atributo>
            <atributo etiqueta="Impresor">
              <correspondencia>
                <where><fijo>     ed.impresor     like     '%#valor#%'
</fijo></where>
              </correspondencia>
            </atributo>
            <has>

<!-- Inicio de Ejemplar (has de Edición) -->
              <concepto etiqueta="Ejemplar">
                <correspondencia>
                  <select> </select>
                  <from> ejemplar ej </from>
                  <where>
                    <fijo> ej.tituloabre = rel.tituloabre
                           and ej.cod_edic = ed.cod_edic </fijo>
                  </where>
                </correspondencia>
                <atributo etiqueta="Encuadernación">
                  <correspondencia>
```

```
                    <where><fijo>      ej.encuadern      =      '#valor#'
</fijo></where>
                  </correspondencia>
                </atributo>
                <atributo etiqueta="Exlibris">
                  <correspondencia>
                    <select> </select>
                    <from> </from>
                    <where>
                      <fijo> ej.ex_libris like '%#valor#%' </fijo>
                      <opcional    fin="">    and    ej.ex_libris    like
'%#valor#%' </opcional>
                    </where>
                  </correspondencia>
                </atributo>
                <has>

<!-- Inicio de Biblioteca (has de Ejemplar) -->
                  <concepto etiqueta="Biblioteca">
                    <correspondencia>
                      <select> </select>
                      <from> bibliote bi </from>
                      <where><fijo>    ej.cod_biblio    =    bi.cod_biblio
</fijo></where>
                    </correspondencia>
                    <atributo etiqueta="Nombre">
                      <correspondencia>
                        <where><fijo>    bi.nombre    like    '%#valor#%'
</fijo></where>
                      </correspondencia>
                    </atributo>
<!-- Fin de Biblioteca (has de Ejemplar) -->
                  </concepto>

                </has>
<!-- Fin de Ejemplar (has de Edición) -->
              </concepto>

            </has>
<!-- Fin de Edición (has de Relación de Sucesos)-->
          </concepto>
        </has>
<!-- Fin de Relación de Sucesos (isa de Obra)-->
      </concepto>

    </isa>
<!-- Fin de Obra -->
  </concepto>
</ac>
```

Apéndice III

Esqueletos de Frase y Metáforas Cognitivas

3.1 Introducción

En este apéndice se presentan las frases en Lenguaje Natural Acotado que hemos usado en el Sistema de Acceso Integrado.

Como ya hemos comentado en el Capítulo 3, esta es nuestra propuesta de Frases en Lenguaje Natural Acotado para consultar las tres bases de datos. Sin embargo, tanto las técnicas de diseño de Interfaces de Usuario como el diseño del Generador de Interfaces de Consulta y los Árboles de Conceptos, permiten de forma muy sencilla, cambiar el aspecto y el tipo de consultas permitidas en la Interfaz de Consulta.

3.2 Frases en Lenguaje Natural Acotado

En la Tabla 17 se muestran las frases en Lenguaje Natural Acotado que actualmente tienen asociados los atributos del concepto "Obra" de la Parte General del Árbol de Conceptos del Sistema de Acceso Integrado a las tres Bibliotecas Digitales del Siglo de Oro.

Tabla 17. Frases en LNA para la Parte General

El **título** de la obra debe contener los siguientes fragmentos de palabras: #valor1#, #valor2#, #valor3#
El **autor** de la obra debe ser: #valor#
El **tema** de la obra debe estar definido por las siguientes palabras clave: #valor1#, #valor2#, #valor3#
El **año de edición** debe estar entre: #valor1#, #valor2# El **lugar de edición** debe ser: #valor#

Tabla 18. Frases en LNA para el subárbol de *Relaciones de Sucesos*

El **título** de la relación debe contener los siguientes fragmentos de palabras: #valor1#, #valor2#, #valor3#
El **autor** de la relación debe ser: #valor#
El **lugar referido** debe ser: #valor#
El **año del acontecimiento** debe estar entre: #valor1#, #valor2#
La **tipología** debe ser: LISTA DESPLEGABLE
El **subgénero** debe ser: #valor#
La **modalidad del discurso** debe ser: LISTA DESPLEGABLE
El **año** de edición debe estar entre: #valor1#, #valor2#
El **lugar** de edición debe ser: #valor#
El **editor** debe ser: #valor#
El **impresor** debe ser: #valor#
La **encuadernación** debe ser: #valor#
El **exlibris** debe ser: #valor#
La **biblioteca** debe ser: #valor# LISTA DESPLEGABLE

Tabla 19. Frases en LNA para el subárbol de *Libros de Emblemas*

El **título** de la obra debe contener los siguientes fragmentos de palabras: #valor1#, #valor2#, #valor3#
El **motivo** de la imagen de los emblemas debe contener los siguientes fragmentos de palabras: #valor1#, #valor2#, #valor3#
El **epigrama** del emblema debe contener los siguientes fragmentos de palabras: #valor1#, #valor2#, #valor3#
La **glosa** del emblema debe contener los siguientes fragmentos de palabras: #valor1#, #valor2#, #valor3#
El **autor** de la obra debe ser: #valor#
El **promotor** debe ser: #valor#
El **editor** debe ser: #valor#
El **impresor** debe ser: #valor#
El **lugar** de edición debe ser: #valor#
El **año** de edición debe estar entre: #valor1#, #valor2#
El **número de versos** del epigrama debe estar entre: #valor1#, #valor2#

El emblema debe tener al menos #límite# de las siguientes **palabras clave:** #valor1#, #valor2#, #valor3#

El emblema debe tener al menos #límite# de las siguientes **onomásticas:** #valor1#, #valor2#, #valor3#

El emblema debe tener al menos #límite# de las siguientes **autoridades generales:** #valor1#, #valor2#, #valor3#

La imagen del emblema debe tener al menos #límite# de los siguientes **objetos:** #valor1#, #valor2#, #valor3#

El emblema debe tener algún **exempla** que contenga alguno de los siguientes fragmentos de palabras: #valor1#, #valor2#, #valor3#

Los **motes** de los emblemas deben:
- Coincidir en la imagen y en el cuerpo del emblema
- Aparecer exclusivamente en la imagen
- Aparecer exclusivamente en el cuerpo

El **mote**
- original
- traducido

debe contener el siguiente fragmento de palabra #valor#

La **estrofa** del epigrama debe ser: #valor# LISTA DESPLEGABLE

El **idioma** del epigrama debe ser: #valor# LISTA DESPLEGABLE

El **tipo de verso** del epigrama debe ser: #valor# LISTA DESPLEGABLE

Apéndice IV

Lenguaje de Consulta

4.1 Introducción

En este apéndice se presenta la especificación formal del Lenguaje de Consulta que hemos definido para comunicar el Mediador con los Sistemas Envoltorios de las bases de datos componentes.

4.2 DTD del Lenguaje de Consulta

Como ya se ha comentado, el Lenguaje de Consulta del sistema está definido en XML. Su DTD se muestra en la Tabla 20. A continuación se describe el Lenguaje de Consulta en términos de los elementos por los que está formado su DTD.

Tabla 20. DTD del Lenguaje de Consulta

```
<!ELEMENT consulta (concepto)>
<!ELEMENT concepto (isa | (atributo*, has?))>
<!ATTLIST concepto nombre CDATA #REQUIRED>
<!ELEMENT isa (CONCEPTO)>
<!ELEMENT has (CONCEPTO+)>

<!ELEMENT atributo ((operador-binario, valor) |
                    (contiene, valor+, thesaurus*) |
                    (entre, valor, valor))
<!ATTLIST atributo nombre CDATA #REQUIRED>

<!ELEMENT operador-binario EMPTY>
<!ATTLIST operador-binario op (igual|
                               distinto|
                               menor|
                               menor-igual|
                               mayor|
                               mayor-igual) "igual">
```

```
<!ELEMENT entre EMPTY>
<!ELEMENT contiene EMPTY>
<!ATTLIST contiene limite CDATA #REQUIRED>

<!ELEMENT valor EMPTY>
<!ATTLIST valor constante CDATA #REQUIRED>

<!ELEMENT thesaurus EMPTY>
<!ATTLIST thesaurus nombre (sinonimos|
                            antonimos|
                            palabrasrelacionadas) "sinonimos">
```

Como se puede ver en la Tabla 20, la estructura del Lenguaje de Consulta es idéntica a la estructura del Árbol de Conceptos. La diferencia radica en la información asociada a los términos. Por un lado, los conceptos no tienen información asociada. Por otro lado, los atributos llevan asociada la condición que el usuario haya expresado sobre ellos. En términos del DTD:

El elemento `atributo` almacena una única condición sobre un atributo de un concepto. Para ello, este elemento almacena: el nombre del atributo que restringe, (en su propiedad `nombre`) el operador que se usa para restringirlo y el valor o valores con los que se compara el atributo. El elemento `atributo` está compuesto por uno de los conjuntos (pares o triples) de elementos que veremos a continuación. El elemento `valor`, almacena una constante que puede representar, entre otras cosas, un número, una cadena de texto o una fecha. Dependiendo del primer elemento del par `(operador-binario)`, pueden existir uno o varios elementos `valor`.

A continuación se describen los elementos por los que está formado un elemento `atributo` que, en conjunto, permiten almacenar cualquier condición que haya expresado el usuario:

- `(operador-binario, valor)`: `operador-binario` es un elemento vacío con una propiedad `op` que puede ser instanciada con los elementos de una lista de operadores preestablecidos. Cada uno de ellos representa el operador que lleva implícito en el propio nombre. La lista de operadores permitidos son: `igual`, `distinto`, `menor`, `menor-igual`, `mayor` y `mayor-igual`. El elemento `valor` almacena una constante introducida por el usuario.
- `(contiene, valor+, thesaurus*)`: este tipo de condición está diseñado para almacenar consultas sobre el contenido de los documentos usando alguna técnica de recuperación de textos. La semántica de esta condición es que el atributo, al que se refiera, debe contener todas las palabras claves almacenadas individualmente en elementos `valor` o, al menos, un cierto número de ellas cuando dicho número así se especifique

en el atributo `limite` (propiedad del elemento `contiene`). Con la propiedad `nombre` del elemento `thesaurus` se puede forzar a que se realice la búsqueda de los sinónimos, antónimos o palabras relacionadas de las palabras escritas por el usuario y almacenadas, como hemos dicho, en el elemento `valor`. Esta característica, que es opcional, posibilita el uso de varios diccionarios simultáneamente.

- `(entre, valor, valor)` : además de los operadores menor y mayor tratados en el primer tipo de condición, hemos considerado un operador de segundo nivel que equivale a dos de las condiciones anteriores con los operadores mayor y menor.