

# Uso de DocBook en la Creación de Contenidos Educativos Interoperables

Martínez-Ortiz, I.<sup>1</sup>, Moreno-Ger, P.<sup>2</sup>, y Fernández-Manjón, B.<sup>2</sup>

<sup>1</sup>Centro de Estudios Superiores Felipe II. Madrid. Spain  
imartinez@cesfelipesecondo.com

<sup>2</sup>Facultad de Informática, Universidad Complutense de Madrid. Madrid. Spain  
pablom@fdi.ucm.es, balta@sip.ucm.es

**Resumen** En *e-learning* la base de los contenidos son los objetos educativos pero no está tan claro cómo diseñarlos, construirlos o mantenerlos a lo largo del tiempo. Los docentes están más familiarizados con la creación de contenidos educativos tradicionales (e.g. manuales técnicos o libros) y por tanto hay que buscar metodologías que simplifiquen su transformación en objetos educativos. En esta línea se propone como base la utilización de DocBook, un modelo de documento para documentación técnica, que mediante técnicas automáticas (o semiautomáticas) puede ayudar a simplificar la creación y gestión de objetos educativos.

## 1 Introducción

Entre las tareas que desarrolla un docente es habitual que se encargue de la creación de *apuntes* para apoyar sus clases, en las cuales se presentan los conceptos clave de la materia, permitiendo al alumno profundizar en la materia o clarificar los conceptos expuestos en clase. Habitualmente estos *apuntes* son creados en forma de conjuntos de temas o incluso de libro.

Ahora con la disposición de plataformas de enseñanza electrónica (Learning Management System, LMS) el docente tiene que incrementar su esfuerzo para transformar sus contenidos educativos a un formato electrónico que pueda ser utilizado dentro del LMS que haya sido implantado en su centro.

En e-learning junto al desarrollo de LMSs se han propuesto metodologías y estándares para permitir el intercambio y reutilización de contenidos didácticos entre dichos sistemas, de manera que se minimice el coste del desarrollo de contenidos educativos de alta calidad aunque el docente no sea un experto en el dominio de la materia sobre la que se está creando el contenido.

Esta metodología se denomina Modelo de Objetos Educativos (MOE) (ver [5], [11]) y se basa en la definición de “piezas” de contenidos, denominadas Objetos Educativos (OE), que cubren uno o un conjunto reducido de objetivos educativos. Estos OE tienen mayor probabilidad de ser reutilizados con otros propósitos que un curso monolítico. Además los OE pueden ser ensamblados o adaptados para ajustarse a las diferentes necesidades de los alumnos.

Para que tenga éxito este modelo es necesario que se propongan de estándares que definan el proceso para la creación, documentación, catalogación e intercambio de estos OEs. Este esfuerzo de estandarización tiene que tratar con conceptos como el intercambio de OEs (i.e. interoperabilidad) o el mantenimiento de los metadatos que describen los posibles usos de cada OE específico.

En resumen, un docente (o un creador de contenidos educativos) que desee crear contenidos educativos que sigan el MOE necesita crear dichos contenidos y adicionalmente documentarlos y empaquetarlos, según los estándares de intercambio. Es necesario que el creador de contenidos debe tener en cuenta que debería crear dichos contenidos en un formato que pueda ser entendido por el mayor número de herramientas que sea posible. Con este propósito nos planteamos utilizar DocBook un modelo de documentación formulado en XML y orientado inicialmente a la documentación técnica para simplificar el proceso de desarrollo de LO:

El resto de la comunicación se estructura de la siguiente manera: en la sección 2 presentaremos las principales características de los lenguajes de marcado, en particular DocBook, que hacen adecuado su uso como formato de intercambio de contenidos; en la sección 3 se describirá nuestro modo de trabajo para la creación / revisión / publicación de los contenidos educativos y las mejoras obtenidas mediante el uso de DocBook; y finalmente serán presentadas las conclusiones obtenidas y algunas líneas futuras de investigación.

## 2 Lenguajes descriptivos de marcado y DocBook

En el dominio documental se han propuesto distintas soluciones para simplificar la tarea y maximizar la productividad del creador de contenidos (documentalistas, investigadores, escritores, etc.). Los objetivos principales de dichas soluciones son:

1. Los autores sólo deben concentrarse en la creación y estructuración del contenido, no en su presentación final.
2. Los métodos de creación y de procesamiento/presentación deben estar separados.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE book PUBLIC "-//OASIS//DTD DocBook XML V4.2//EN"
"http://www.oasis-open.org/docbook/xml/4.2/docbookx.dtd" [
  <!ENTITY conceptosBasicos SYSTEM "basico.xml">
<chapter id="tema1">
  <chapterinfo>
    <abstract><para>Explica los <emphasis>conceptos
básicos</emphasis> sobre el lenguaje C++, la utilización de
<emphasis>variables</emphasis>, <emphasis>tipos de datos
predefinidos</emphasis>.</para></abstract>
  </chapterinfo>
  <title>Tema1: Introduccion al Lenguaje C++</title>
  &conceptosBasicos;
  ...
```

**Fig. 1.** Ejemplo de documento XML marcado con el vocabulario DocBook. Puede observarse la definición de metadatos dentro de <chapterinfo> y el uso de marcas referenciales.

En [3] y [6] se describe el uso de los lenguajes descriptivos de marcas con el objetivo de capturar el proceso de estructuración mental del creador de contenidos y como estos lenguajes de marcas permiten plasmar dicha estructura. Este hecho contrasta con el trabajo que se realiza habitualmente con herramientas tipo Word, donde la información de estructura normalmente se pierde ya que se hace mayor hincapié en aspectos de la presentación final (e.g. tamaños de letra, fuentes, etc.).

Actualmente el lenguaje de marcas más utilizado es el eXtensible Markup Language (XML) [17], siendo éste un metalenguaje que nos permite definir nuestros propios lenguajes de marcas, como por ejemplo, DocBook.

Una característica técnica adicional de los lenguajes de marcado es que son fácilmente tratables mediante aplicaciones software [10]. En particular, en el caso del XML, existen numerosas herramientas incluidas directamente en los propios sistemas operativos o en la mayoría de lenguajes de programación que nos permiten transformar automáticamente un documentos mediante el uso de XSL Transformation (XSLT) [16] o dar formato a los documentos mediante el XML StyleSheet Language (XSL).

## 2.1 DocBook

DocBook es un lenguaje de marcado creado originalmente por la editorial O'Reilly y la empresa HaL Computer Systems en 1991 con el objetivo de facilitar el intercambio de manuales de aplicaciones software entre distintas compañías y editoriales. Trataba de responder a los problemas que surgían debido a la variedad de plataformas y sistemas utilizados por los autores, revisores y editores.

DocBook poco a poco ha ido convirtiéndose en un estándar de-facto [4], para la creación de documentación de proyectos en el ámbito del software (JBoss, GNOME) o para la escritura de informes técnicos (CERN, Microsoft, Novell), y también se usa en la creación de libros [14].

DocBook es un lenguaje XML que define un conjunto de etiquetas que permite crear documentos siguiendo la estructura de un artículo (útil para documentos de pequeño tamaño), o de un libro (para documentos mayores). Para ambos tipos de documentos proporciona un conjunto de etiquetas para estructurarlos en divisiones de menor tamaño (partes, capítulos, secciones, párrafos). Además incluye un vocabulario particularmente útil para la documentación de aplicaciones software y dispositivos hardware.

Adicionalmente a la estructuración del documento en distintas divisiones anidadas, también proporciona soporte para otras características generales que permiten marcar:

- Contenidos multimedia.
- Referencias: Bibliográficas, a partes del documento, a recursos externos.
- Preguntas Frecuentes
- Recursos editoriales: Notas al pie, advertencias, consejos, trucos.
- Metadatos
- Etc.

Además de DocBook, existen otros lenguajes de marcas para documentación. Sin embargo las claves del éxito de DocBook son:

#### 4 Martínez-Ortiz, I.1, Moreno-Ger, P.2, y Fernández-Manjón, B.2

- Ha sido y está siendo utilizado por un amplio conjunto de organizaciones
- Existen herramientas de autoría y herramientas para su tratamiento y procesamiento que son de Código Abierto.
- La semántica del lenguaje está ampliamente documentada.
- Podemos crear fácilmente distintos tipos de salida (HTML, PostScript, PDF, RTF) mediante el uso de las hojas XSLT de DocBook [13].

Desde el punto de vista técnico cabe destacar:

- La modularidad de la definición de DocBook. Este hecho nos permitirá particularizar el lenguaje para dar soporte a nuestras necesidades como creadores de OE.
- Modularidad de las XSLT de DocBook. Siendo ésta la principal herramienta de trabajo, podremos crear herramientas de procesamiento de documentos DocBook adaptadas a nuestras necesidades sin empezar desde cero.

### 3 Uso de DocBook en la creación de contenidos educativos

Antes de impartir un curso, un profesor generalmente comienza con la definición de los objetivos del curso (figura 2, paso a)). Estos objetivos pueden ser descritos narrativamente de manera informal, o pueden ser descritos en mayor detalle enumerando o refinando los objetivos educativos que se definen como objetivo general del curso.

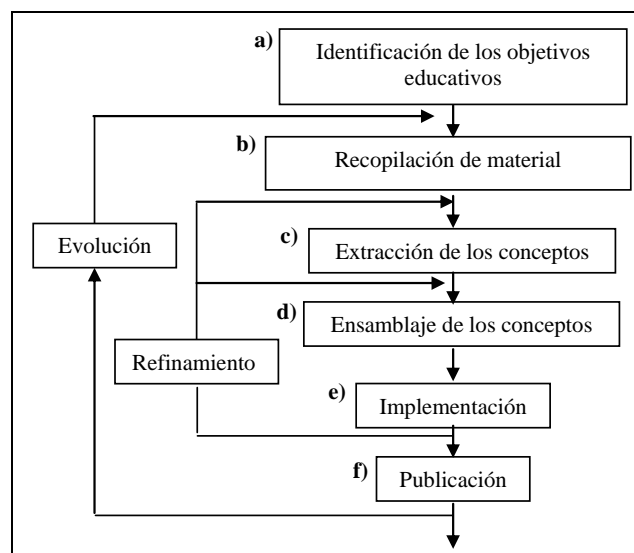


Fig. 2. Ciclo de creación del contenido educativo de un curso

Una vez que estos objetivos están claros, el profesor continúa su trabajo con la fase de recopilación de información, mediante el uso de buscadores tradicionales, libros o incluso utilizando un repositorio de OE como MERLOT [12] (figura 2, paso b)). Esta etapa puede ser simplificada mediante el uso de DocBook ya que existe un conjunto de herramientas que pueden ser aplicadas para realizar conversiones desde distintos formatos de archivos (HTML, XHTML, PDF, LaTeX, OpenOffice, etc.) a DocBook, permitiéndonos convertir gran cantidad de información directamente en formato DocBook.

Una vez que hemos hecho acopio de todo el material que creamos suficiente y que nos permita cubrir de manera global todos nuestros objetivos educativos, pasamos a realizar un proceso de edición más cuidadoso para extraer los conceptos principales de estos materiales, podemos comenzar a realizar resúmenes y destacar los puntos clave (figura 2, paso c)). Esta primera información además nos servirá como información para rellenar los metadatos de los OE.

En este paso del desarrollo, el profesor tiene que tratar con la sintaxis XML directamente, para ello podemos hacer uso de herramientas que nos faciliten esta tarea como Altova XMLSpy, Emacs, etc. Estas herramientas nos ayudan a generar un documento DocBook tratando directamente con la sintaxis XML. También es posible evitar el trabajo directo con XML mediante editores What You See Is What You Get (WYSIWYG) como Lyx y OpenOffice.

Además de la extracción de las ideas principales es necesario descomponer el contenido global o “en bruto” en OEs. La descomposición de los contenidos en OEs permite flexibilizar la reutilización de los mismos. El tamaño de descomposición en OE (i.e., su granularidad) es un punto clave para su reutilización [13].

Una vez que hemos descompuesto los materiales en bruto en OE, el siguiente paso requiere que se ensamblen los OE para cubrir los objetivos educativos propuestos para el curso (figura 2, paso d)).

En la fase de implementación (figura 2, paso e)), el profesor trata en más detalle la codificación del OE utilizando DocBook, por ejemplo, puede crear un glosario de términos para cada OE, puede incluir materiales multimedia, etc.

Tras codificar todos los OE, sólo queda publicar el contenido, haciendo uso de las potentes XSLT de DocBook [17] que nos permitirán generar numerosos formatos de salida (PDF, RTF, HTML), ya que estas hojas de estilo son independientes a los documentos que generamos y junto a su diseño modular, podemos aplicar distintas hojas de estilo para cubrir nuestras necesidades. Por ejemplo, podemos añadir características de accesibilidad a nuestra salida para que los lectores de páginas para deficientes visuales puedan trabajar de manera correcta, podemos añadir otras características como son el uso de un conjunto de colores de alto contraste también con el objetivo de permitir que nuestro contenido pueda ser accedido por un mayor número de alumnos, y así sucesivamente.

Finalmente, hay otras dos etapas de realimentación. La etapa de refinamiento puede ser necesaria mientras que se está impartiendo un curso, debido a que sea necesario modificar cómo están acoplados los OE de manera que se adapten de manera más adecuada a las capacidades de nuestros estudiantes. La etapa de evolución, como su nombre sugiere, permite al profesor hacer evolucionar el curso una vez han sido obtenidas las impresiones de los alumnos a lo largo del mismo. El uso de la metodología propuesta y las herramientas asociadas a DocBook pueden

ayudar a simplificar determinadas tareas asociadas con la autoría, la inclusión de metadatos o la coherencia con los estándares de e-learning en este desarrollo.

### 3.1 IMS Content Packaging y IEEE LOM Metadata: automatización de la creación de ficheros XML

La especificación IMS Content Packaging [9] describe cómo empaquetar distintos materiales educativos (un objeto educativo por sí mismo, un curso, etc.) para que puedan distribuirse y puedan intercambiarse entre distintos LMS, repositorios de OE o entre distintos profesores que utilicen herramientas compatibles con dicha especificación. Esta especificación describe la estructura y formato que tiene que tener una unidad de intercambio denominada Package Interchange Format (PIF).

De acuerdo a esta especificación un paquete PIF está formado por un conjunto de archivos físicos con su contenido y un manifiesto. El manifiesto es un archivo XML que refleja de manera comprensiva la estructura de los contenidos, el tipo de los mismos y sus posibles organizaciones. De manera más precisa un manifiesto contiene:

- **Meta información sobre el propio paquete.** Esta meta información sigue el estándar Learning Object Metadata (LOM) propuesto por el IEEE LTSC [7]. El propósito de estos metadatos es facilitar la gestión y la indexación de dichos contenidos para realizar búsquedas más exactas dentro de un repositorio de OE.
- **Descripción de los recursos del paquete.** En su manera más simple, un recurso está asociado con un archivo físico de contenido educativo. Así mismo, un recurso puede agrupar un conjunto de archivos físicos.
- **Organización de los recursos.** Una organización proporciona una vista jerárquica de los Recursos que forman el PIF. Hay que resaltar que un manifiesto puede incluir múltiples organizaciones, cada una de las cuales proporciona una vista alternativa a la organización de contenidos, y por tanto una vista diferente del PIF.

En particular, una de las tareas más tediosas es la anotación de los recursos educativos utilizando metadatos LOM debido a la gran cantidad de etiquetas disponibles. Como docentes y creadores de contenidos es necesario que incluyamos los metadatos referentes a la finalidad educativa, que corresponde a la categoría *Educational*, siendo ésta una categoría de las nueve propuestas en IEEE LOM.

Podemos utilizar las etiquetas disponibles en DocBook y aprovechar la modularidad de su DTD para añadir las etiquetas para anotar los contenidos durante su creación (en particular, las etiquetas pertenecientes a la categoría Educational de IEEE LOM). De esta manera tenemos los contenidos y los metadatos asociados en un mismo lugar, facilitando el mantenimiento de los mismos.

Sin embargo existe numerosa información que también es importante que tengan asociada, como es la información de contacto del autor o los autores y las revisiones realizadas sobre el contenido, requisitos técnicos de la versión publicada. En general, dicha meta-información ya está disponible en el documento DocBook y puede ser incluida de manera automática [2], simplificando la tarea del creador.

La necesidad de crear un manifiesto, hace habitualmente necesario el uso de una herramienta externa como Reload Editor<sup>1</sup> para la creación del mismo. Sin embargo, en el caso de DocBook podemos aprovecharnos de que es un lenguaje XML y de que en el desarrollo de los contenidos hemos definido de antemano una estructura en los propios contenidos. Esta estructura puede ser traducida automáticamente al formato IMS Content Packaging junto con los metadatos asociados que hayamos incluido dentro de la definición de los recursos.

Este proceso de conversión podrá ser llevado a cabo mediante un procesado del documento (mediante una hoja XSLT), siendo este proceso configurable según nuestras necesidades. Por ejemplo, en nuestras pruebas, consideramos la etiqueta <section> de DocBook como unidad mínima de información, siendo traducida a un <item> en el lenguaje IMS Content Packaging.

### 3.2 Integración con IMS Learning Design

Dentro del proceso de creación de una Unidad de Aprendizaje siguiendo la especificación IMS Learning Design [11] necesitamos crear los contenidos educativos que serán mostrados en las Actividades que realizarán los alumnos.

Dentro de dichos contenidos, podemos hacer uso de las propiedades disponibles en el nivel B de la especificación de IMS LD que permiten personalizar los contenidos para que el alumno se sienta más integrado en la experiencia educativa (utilizando información personal del alumno).

Adicionalmente podemos utilizar las propiedades para modificar el contenido en sí mismo, creando un camino de aprendizaje personalizado a las preferencias del alumno. La disponibilidad de los contenidos en XML, en particular DocBook, junto a la posibilidad de extender el lenguaje según nuestras necesidades, nos proporciona mayor flexibilidad a la hora de poder mejorar la experiencia educativa de nuestros alumnos.

### 3.3 Creación de Paquetes Compatibles con ADL SCORM

Además de las especificaciones propuestas por IMS existen otras iniciativas como la propuesta por el Advanced Distributed Learning (ADL). Esta iniciativa particulariza las especificaciones propuestas por IMS creando lo que comúnmente es denominado *perfil de aplicación*. Dentro del perfil de aplicación propuesto por ADL se encuentra el Sharable Content Object Referente Model (SCORM) Content Aggregation Model [1]. En esta especificación se define el modelo de componentes con los que trabaja SCORM. En particular se definen los Sharable Content Object (SCO), donde estos contenidos tienen la posibilidad de comunicarse con un Runtime Environment (RTE) para obtener/almacenar información a cerca de la experiencia educativa mediante el uso de ECMAScript (JavaScript) [8].

Además de definir el modelo particular del perfil de aplicación de SCORM también se definen los requisitos necesarios para que un PIF sea compatible con el

---

<sup>1</sup> El sitio oficial de Reload Editor es: <http://www.reload.ac.uk>

perfil de aplicación. Se describe los elementos que deben aparecer en el manifiesto, como los metadatos obligatorios y los requisitos necesarios de los SCO (en particular, estos últimos necesitan comunicarse con el RTE al menos para indicar que han comenzado a visualizarse y para indicar que termina su visualización).

Se puede simplificar la tarea de creación de contenidos educativos que sean compatibles con ADL SCORM. Por ejemplo, es posible generar el JavaScript necesario en los SCO para que los contenidos sean compatibles directamente aun cuando no hayan sido pensados para tales efectos. Además se puede simplificar la tarea de la comunicación con el RTE, ya que el uso de JavaScript necesario para la comunicación depende del formato del SCO. Por ejemplo, si hemos generado un PDF, una película Macromedia Flash o un gráfico en formato Scalar Vector Graphics (SVG) a partir del documento DocBook, es posible hacer uso de las posibilidades específicas que ofrecen los intérpretes de dichos formatos para la ejecución de JavaScript. De esta manera se libera al creador de contenidos de los detalles técnicos necesarios para llevar a cabo la comunicación.

## **4 Conclusiones**

Al optar por la codificación de contenidos mediante el uso de XML obtenemos múltiples beneficios, tales como simplificar la reutilización, el intercambio y la publicación multiformato, La clave es buscar métodos que permitan atraer a los autores de contenidos a estas tecnologías que simplifican la automatización, total o parcial, de las tareas tediosas que se deben realizar para la creación de los OE, y en particular las relativas a temas de estandarización.

En particular, al utilizar DocBook se obtienen ciertos beneficios añadidos a los sistemas de creación de contenidos basados en XML. Por ejemplo, se tienen disponible un amplio arsenal de herramientas para su autoría y tratamiento y se puede reutilizar gran cantidad de contenidos preexistentes.

No obstante esta metodología no está exenta de problemas. Es necesario tener un conocimiento de las tecnologías de marcado XML y DocBook y tener una idea clara del modelo subyacente a los OE. Actualmente estamos haciendo pruebas de esta metodología para la creación de un curso de programación en C++. En este caso los desarrolladores de contenidos conocen XML pero no son expertos en DocBook ni en los estándares educativos. En este caso podremos identificar cuáles son los problemas concretos con los que se enfrentan (v.g. deficiente estructuración) y cómo se puede mejorar el proceso o las herramientas asociadas para evitarlos.

Para finalizar, resaltar que en nuestro equipo de trabajo apostamos por el uso de estas técnicas y por el desarrollo de nuevas herramientas que faciliten el uso de esta tecnología, junto con las guías de estilo y desarrollo necesarias para la aplicación de DocBook al campo de la creación de OE que sigan las especificaciones de IMS y ADL/SCORM.

## Agradecimientos

Este trabajo ha sido financiado parcialmente por la Comisión Interministerial de Ciencia y Tecnología (proyectos TIC2001-1462 y TIN2004-08367-C02-02) y por un proyecto de innovación educativa de la Universidad Complutense de Madrid. Agradecemos los comentarios realizados por José Luis Sierra a una versión previa de este trabajo.

## Referencias

1. ADL, 2004, ADL SCORM Content Aggregation Model v. 1.3.1 (Disponible on-line: <http://www.adlnet.org/downloads/files/67.cfm>).
2. Cardinae, K., Meire, M., Duval, E., 2005, Automating Metadata Generation: the Simple Indexing Interface (International World Wide Web Conference Committee, IW3C2)
3. Coombs, J., Renear, A., DeRose S., 1987, Markup Systems and the Future of Scholarly Text Processing (Communications of the ACM, Volume 30, Issue 11) pp 933-947
4. Dougherty, D., 1999, The Making Of DocBook DTD (O'Reilly XML.com disponible on-line: <http://www.xml.com/pub/a/1999/10/docbook/docbook-making.html>).
5. Downes, S., 2001, Learning Objects: Resources For Distance Education Worldwide (International Review of Research in Open and Distance Learning. Accedido el 25 de Abril, 2005: <http://www.irrodl.org/content/v2.1/downes.html>)
6. Goldfarb, C.F., 1981, A Generalized Approach to Document Markup (Proceedings of the ACM SIGPLAN SIGOA symposium on Text manipulation) pp 68-73
7. IEEE, 2002, IEEE 1484.12.1-2002 Learning Object v1 Metadata Final Draft, (Accedido el 18 de Abril, 2005: [http://ltsc.ieee.org/wg12/files/LOM\\_1484\\_12\\_1\\_v1\\_Final\\_Draft.pdf](http://ltsc.ieee.org/wg12/files/LOM_1484_12_1_v1_Final_Draft.pdf))
8. IEEE, 2003, IEEE 1484.11.2 Standard for Learning Technology – ECMAScript Application Programming Interface for Content to Runtime Services Communication. (Disponible on-line: <http://ltsc.ieee.org/>)
9. IMS CP, IMS Content Packaging Specification v1.1.4, 2004 (Accedido el 19 de Abril, 2005: <http://www.imspj.org/content/packaging/>)
10. Khare, R., Rifkin, A., 1997, XML: a door to automated Web applications (IEEE Internet Computing, IEEE, Volume 1, Issue 4, July-Aug. 1997) pp. 78-87
11. Koper, E.J.R., 2003, Combining re-usable learning resources and services to pedagogical purposeful units of learning. ( A. Littlejohn (Ed.), Reusing Online Resources: A Sustainable Approach to eLearning London: Kogan Page. ) pp 46-59.
12. McMartin, F., 2004, MERLOT: A Model for User Involvement in Digital Library Design and Implementation ( Journal of Digital Information, Volume 5 Issue 3, Article No. 293, 2004-09-09)
13. Polsani, P., 2003, Use and Abuse of Reusable Learning Objects (Journal of Digital Information, Volume 3 Issue 4, Article No. 164, 2003-02-19)
14. Sklar, D., 2004, Writing Learning PHP 5. Accedido el 20 de Mayo de 2005: <http://www.onlamp.com/pub/a/php/2004/11/11/writingphp5.html>
15. Stayton, B., 2005, DocBook XSL: The Definitive Guide Third Edition (SageHill Enterprises)
16. W3C: World Wide Web Consortium, 2001b, XSL Transformations (XSLT). Accedido el 19 de Abril, 2005: <http://www.w3.org/Style/TR/xslt/>
17. W3C: World Wide Web Consortium, 2004, Extensible Markup Language (XML) (Third Edition). Accedido el 19 de Abril, 2005: <http://www.w3.org/TR/REC-xml/>