

# SPAR 1.0 - REPOSITORIO DIGITAL DE ACCESO PÚBLICO BASADO EN SCORM

---

**CARLOS COBOS**

*Ingeniero de Sistemas, Magíster en Informática  
Departamento de Sistemas, Facultad de Ingeniería Electrónica y Telecomunicaciones  
Universidad del Cauca  
ccobos@unicauca.edu.co*

**ERWIN MEZA**

*Ingeniero de Sistemas, Magíster en Informática  
Departamento de Sistemas, Facultad de Ingeniería Electrónica y Telecomunicaciones  
Universidad del Cauca  
emezav@unicauca.edu.co*

**MARTHA MENDOZA**

*Ingeniera de Sistemas, Magíster en Informática  
Departamento de Sistemas, Facultad de Ingeniería Electrónica y Telecomunicaciones  
Universidad del Cauca  
mmendoza@unicauca.edu.co*

**JESÚS MUÑOZ**

*Estudiante de Ingeniería de Sistemas Décimo Semestre  
Programa de Ingeniería de Sistemas, Facultad de Ingeniería Electrónica y Telecomunicaciones  
Universidad del Cauca  
jmunozramirez@unicauca.edu.co*

**JORGE GIRALDO**

*Estudiante de Ingeniería de Sistemas Décimo Semestre  
Programa de Ingeniería de Sistemas, Facultad de Ingeniería Electrónica y Telecomunicaciones  
Universidad del Cauca  
jgiraldoj@unicauca.edu.co*

*Fecha Recepción: 7 de marzo de 2006*

*Fecha Aceptación: 13 de junio de 2006*

## RESUMEN

*El desarrollo de objetos de aprendizaje ha promovido el estudio de diferentes implementaciones que puedan garantizar la interoperabilidad de estos objetos entre los diferentes repositorios de contenidos, los cuales deben permitir buscar y acceder su propia información de forma independiente a los detalles de cada implementación. Este artículo describe el trabajo desarrollado en el trabajo de I+D titulado "SPAR 1.0 - Repositorio de Acceso Público basado en SCORM" desarrollado dentro del programa de Ingeniería de Sistemas de la Universidad del Cauca. En el momento, se ha liberado la primera versión del repositorio de contenidos probado con la especificación SCORM 1.2, el estándar LOM 1.0 y la especificación de tarjetas virtuales personales y organizacionales VCARD 1.0, para lo cual se hace uso de una interfaz basada en una base de datos relacional y un conjunto de servicios Web. De forma más concreta el presente artículo describe el modelo relacional de la Base de Datos que hace posible la manipulación de los meta-datos de los objetos de aprendizaje almacenados dentro del repositorio digital y una corta presentación de la interfaz del sistema.*

**PALABRAS CLAVE:** *Repositorio de contenidos, meta-datos, objetos de aprendizaje, SCORM 1.2, IMS DRI.*

## ABSTRACT

*The development of learning objects has promoted the study about different implementations that permit the interoperability of these objects between different content repositories, each one of them must allow to search and access their own information of independent way to the details of each implementation. This paper describes the work*

developed in research project titled "SPAR 1.0 - SCORM Public-Access Repository" developed inside the program of Engineering of Systems of the University of the Cauca. In the current time, a first version of the content repository has been liberated after to do multiple proves with the specification SCORM 1.2, the standard LOM 1.0 and the specification of personal and organizational virtual cards VCARD 1.0, using an interface based on a relational database and a set of Web services. In a more specific way the present paper describes the relational model of the database that makes possible the manipulation of the metadata of learning objects stored inside a digital repository and a short presentation about the system's interface.

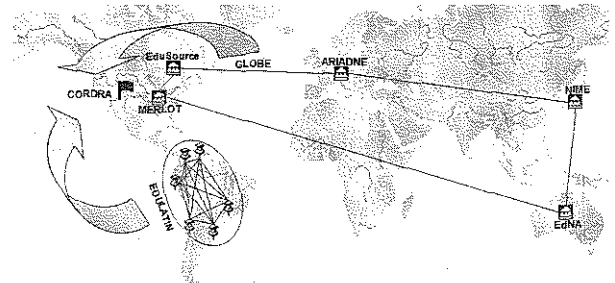
**KEYWORDS:** Content repositories, meta-data, learning objects, SCORM 1.2, IMS DRI.

## INTRODUCCIÓN

En la actualidad el Modelo de Referencia de Objetos de Contenido Compartible (SCORM) [1] integra el esquema de meta-datos IEEE LOM [2] para describir los objetos de aprendizaje (también conocidos como Sharable Content Objects, SCOs) a un ambiente de ejecución (Run-Time Environment), y una interfaz de programación de aplicaciones (API), para ofrecer un mecanismo a través del cual los objetos de aprendizaje puedan ser integrados en diferentes cursos de diversos Sistemas de Gestión de Aprendizaje (Learning Management Systems, LMS). A partir de este desarrollo se ha impulsado la creación de repositorios digitales, que buscan ofrecer servicios para el almacenamiento, la organización y administración, la búsqueda y el acceso a los objetos de aprendizaje [3]. No obstante, a pesar de la existencia de LMSs y repositorios que hacen uso de SCORM como modelo de referencia, no existe un estándar de acogida general que permita la interoperabilidad en la búsqueda y acceso de los objetos de aprendizaje o sus meta-datos entre distintos repositorios. Una estrategia para solucionar este problema consiste en desarrollar especificaciones de arquitecturas para repositorios de objetos de aprendizaje, tales como la especificación para la interoperabilidad de repositorios digitales (IMS DRI) [4], que enumera una serie de servicios básicos que debe ofrecer un repositorio, o el modelo CORDRA™ (Content Object Repository Discovery and Registration/Resolution Architecture) [5] [6], que presenta un modelo para la búsqueda y el acceso de objetos de aprendizaje basado en un esquema federado de repositorios, que realizan registros a un sistema central. Otra aproximación consiste en desarrollar protocolos y modelos específicos a partir de implementaciones existentes, como

Merlot [7], EdNA [8], ARIADNE[9], EduSource Canada[10], NIME [11] que tienen como objetivo ofrecer una serie de servicios para la búsqueda y recuperación de objetos de aprendizaje y meta-datos de sus respectivos repositorios.

Partiendo de la base que en un futuro cercano existirá una gran cantidad de objetos de aprendizaje almacenados en diversos repositorios "gemelos" de América Latina que se ajustan a especificaciones de meta-datos como IEEE LOM (en este caso SPAR [12][13][14] en Colombia sería uno de ellos), nuestro objetivo consistió en desarrollar un modelo que ofrezca una alternativa para que los objetos de aprendizaje puedan ser localizados y utilizados en los LMS de los diferentes países, inicialmente a nivel de América Latina y posteriormente de forma global. En la **Figura 1** se muestra cómo Merlot, EduSource Canada, Ariadne, NIME y EdNA están uniéndose a través de un proyecto denominado GLOBE (Global Learning Object Brokered Exchange), y cómo el desarrollo de los repositorios distribuidos "gemelos" (denominada Edulatin) se puede integrar dentro de una federación de repositorios, como CORDRA.



**Figura 1.** SPAR y EduLatin integrado al mundo de los repositorios.

Con el fin de evitar el esfuerzo requerido para desarrollar nuevos repositorios de contenido y meta-datos, el proyecto SPAR proporciona la implementación inicial del repositorio que cumple con las especificaciones de interoperabilidad del modelo desarrollado, y que podrá ser adaptado a las necesidades individuales de cada comunidad. El uso de esta implementación permitirá que cada institución o país se concentre en la creación de sus propios objetos de aprendizaje y cursos en general, y use las funcionalidades del repositorio para la gestión y divulgación de dichos recursos. Por otra parte el modelo podrá ser integrado en los repositorios actuales, en forma de una interfaz pública de servicios que extenderá sus funcionalidades para ofrecer servicios de búsqueda y acceso a los objetos de aprendizaje y los meta-datos almacenados en cualquier repositorio que pertenezca a la Red Latinoamericana de Repositorios (EduLatin). Gracias a que esta red se basará en servicios

web y hará uso de IEEE LOM como estándar de meta-datos, será posible integrarla en el futuro dentro del registro distribuido de repositorios de GLOBE, o en el registro centralizado de CORDRA.

En SPAR se espera contar con búsquedas distribuidas, es decir, que se permitan la búsqueda y el acceso a los objetos y sus meta-datos desde cualquier nodo de la red de repositorios "gemelos". Además, La estructura generada da soporte a diferentes esquemas de meta-datos que puedan ser mapeados a XML (XML Binding), como LOM o Dublin Core [16], y contempla otras especificaciones que forman parte de los esquemas de meta-datos pero que no se mapean a XML, como la especificación VCARD [17].

La siguiente sección presenta una descripción general del modelo de repositorios, después se describe el modelo físico de la base de datos y algunas interfaces de la aplicación web, y finalmente se presentan unas conclusiones y el trabajo futuro que se espera desarrollar.

## MODELO DE INTEROPERABILIDAD DE REPOSITORIOS DIGITALES

Para el desarrollo del modelo se analizaron algunos estándares y especificaciones para el aprendizaje en línea, entre ellas la iniciativa de Advanced Distributed Learning (ADL [1]) y el Modelo de Referencia de Objetos de Contenido Compartible (Sharable Content Object Reference Model, SCORM) [1], el IMS Global Learning Consortium Inc. (<http://www.imspj.org/>), el Aviation Industry CBT (Computer-Based Training) Committee (AICC), the Alliance of Remote Instructional Authoring & Distribution Networks for Europe (ARIADNE [9]) y el Technology Standards Committee of the IEEE Computer Society [2]. De estos aportes se tomaron aquellos aspectos que están relacionados con el proyecto y que se muestran a continuación:

Un conjunto de campos descriptivos de los medios/recursos (assets), de los objetos de contenido compartible (Sharable Content Object) y de los cursos, denominados meta-datos (Learning Object Meta-data, LOM). Estos meta-datos y algunas recomendaciones para la construcción de los SCOs permiten aumentar la posibilidad de compartir recursos de contenido entre diferentes instituciones que cuentan con diferentes Sistemas de Gestión de Aprendizaje (Learning Management System, LMS).

\* Una especificación para implementar Repositorios Digitales (Digital Repositories Interoperability) de SCOs,

cursos completos y medios con la capacidad de intercambiar información y prestar servicios a diversos LMSs. Definen el siguiente conjunto principal (core) de operaciones [4]:

\* Search/Find: La habilidad de ubicar/encontrar un objeto en el repositorio (puede incluir la habilidad de visualizarlo).

\* Request: Solicitar un objeto de aprendizaje que ha sido localizado.

\* Retrieve: Recuperar un objeto de aprendizaje que ha sido localizado.

\* Submit: Enviar un objeto de aprendizaje a un repositorio para su almacenamiento. Adicionar un objeto al repositorio.

\* Store: Almacenar un objeto de aprendizaje en el repositorio con un identificador único global.

\* Gather (push/pull): Obtener información de los meta-datos de objetos en otros repositorios (repositorios federados).

\* Publish: Proveer información de los meta-datos de los objetos del repositorio a otros repositorios y sistemas.

Esta especificación se esta complementado con el trabajo que desarrolla el Laboratorio de Arquitectura en Sistemas de Aprendizaje (Learning Systems Architecture Lab) de la Universidad de Carnegie Mellon, quienes están definiendo un Modelo para Repositorios Federados en el proyecto CORDRA [5].

El modelo propuesto a partir del desarrollo del proyecto SPAR, se basa en una arquitectura de repositorios que cooperan en forma distribuida, para ofrecer algunos servicios de búsqueda y acceso a los objetos de aprendizaje y sus meta-datos (**Figura 2**). Una de las principales características de esta propuesta es que cada repositorio puede implementar su lógica de manera diferente, pero está en capacidad de interactuar con los demás repositorios de la red si se ajusta a una interfaz de servicios definida por el modelo. Para la implementación de la interfaz de búsqueda y acceso a los objetos de aprendizaje y meta-datos se planteó el uso de servicios Web XML, como mecanismo de comunicación y transferencia de información entre repositorios. Debido a que se basan en protocolos estándar como HTTP, los servicios Web no requieren el desarrollo de nuevos protocolos o Middleware propietarios. Esto permitirá integrar el modelo a las implementaciones de repositorios existentes, al desarrollar una capa adicional de servicios que ocultan los detalles de funcionamiento propio de cada repositorio.

En la **Figura 2** se describen los principales componentes del modelo.

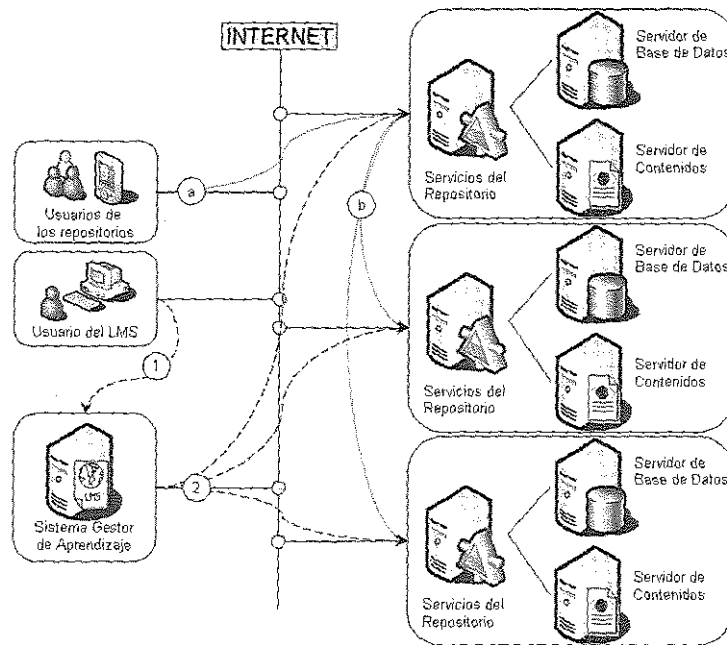


Figura 2. Modelo distribuido de repositorios.

El objetivo de los Sistemas de Gestión de Aprendizaje (LMS) consiste en ofrecer los recursos necesarios para desarrollar experiencias de aprendizaje significativas. En este sentido los LMS hacen uso de los repositorios digitales, y permiten que éstos realicen las tareas de gestión de los objetos de aprendizaje y de los meta-datos. Dentro de su lógica, los LMS deben ofrecer servicios para la búsqueda, visualización y evaluación de los objetos de aprendizaje, con lo cual se convierten en usuarios de los repositorios y actúan como intermediarios que permiten el acceso transparente a sus clientes, que solicitan los contenidos, y los repositorios, que los administran (en la **Figura 2** son los pasos indicados como 1 y 2, el primero hace referencia a la solicitud de un objeto de aprendizaje por parte del usuario, y el segundo a la solicitud del objeto de aprendizaje al repositorio, por parte del LMS).

Los repositorios digitales se pueden definir como una colección de recursos a los cuales se puede tener acceso por medio de una red, sin conocimiento de su estructura. Los recursos almacenados consisten en objetos de aprendizaje, organizados en una estructura de directorios accesible a través de Internet, y los meta-datos, almacenados en forma de archivos XML, o en un sistema gestor de bases de datos (Database Management System - DBMS). En el proyecto SPAR se estudió la posibilidad de utilizar un DBMS con soporte nativo XML para

almacenar los esquemas de los meta-datos, pero debido a que este soporte no se encuentra de forma predeterminada en todos los motores se optó por utilizar un esquema de tablas en un modelo relacional (ver **Figura 3**). Con esto se busca que la implementación del repositorio se pueda realizar en una mayor cantidad de sistemas y plataformas. Este modelo permite además almacenar esquemas de meta-datos jerárquicos como LOM, o de la forma atributo-valor que no se mapean a XML. Esta ventaja del modelo relacional hace posible almacenar en el mismo esquema de datos, las especificaciones completas, como el caso de la especificación VCARD 1.0 (atributo-valor) que forma parte de LOM (mapeada a XML). En la **Figura 3** se pueden apreciar los componentes internos de un repositorio que podrá formar parte de la red EduLatin. Cada repositorio deberá contar con los siguientes componentes:

\* Un modelo de bases de datos que se encarga de dar persistencia a los **meta-datos de los objetos de aprendizaje**. Este modelo (compuesto por tablas y relaciones en su mayoría auto reflexivas) permite ser implementado sobre cualquier motor de bases de datos relacionales, sin necesidad de que la base de datos cuente con soporte para XML en forma nativa. En la actualidad, el modelo ha sido probado para almacenar SCORM 1.2, LOM 1.0 y VCARD 1.0, y se están desarrollando pruebas con el esquema de meta-datos de Dublin Core.

\* Un **catalogo maestro de distribución**, que permite el registro de los repositorios "gemelos" que existen en la red, y que contiene además la información necesaria para hacer uso de las operaciones distribuidas, como la búsqueda de objetos de aprendizaje en todos los repositorios que conforman la red. Se estima que en el futuro se podrá utilizar este catálogo para realizar replicación de datos por demanda. Este catalogo y la lógica de **servicios de distribución**, permiten que un repositorio establezca comunicación con la red, así no cuente con una base de datos distribuida (No implementado en la versión actual).

\* Un conjunto de **Servicios básicos del repositorio** que cumplen con IMS DRI y que son los encargados de dar la funcionalidad principal del repositorio, como son: Search/Find, Request, Retrieve, Submit, Store, Gather (push/pull no completamente terminado en la versión actual) y Publish.

\* Un conjunto de **datos propios y servicios extendidos del repositorio**, que dan identidad propia a cada repositorio de la red. Entre estos servicios se puede mencionar el pago y facturación por el uso de los objetos de aprendizaje privados, y el seguimiento al uso de los objetos de aprendizaje. Dependen de la entidad encargada de la gestión del repositorio.

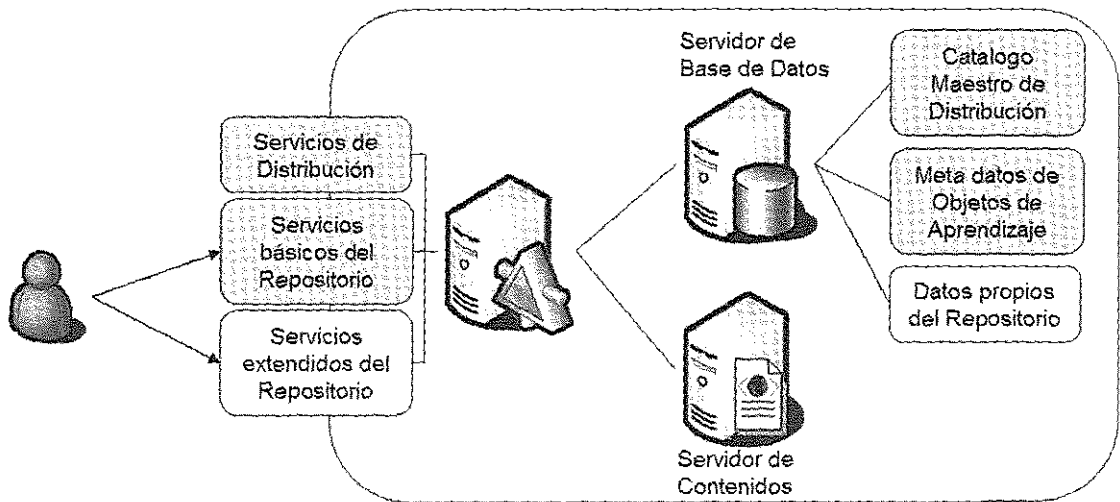


Figura 3. Vista interna de un repositorio.

Una característica importante del modelo consiste en que la procedencia de los objetos de aprendizaje es indiferente al LMS que los solicita, lo cual permite que se puedan utilizar objetos de aprendizaje o meta-datos almacenados en diferentes repositorios de la red, dentro de un mismo curso. Esta característica requiere que cada LMS implemente la porción de la interfaz pública que ofrece los servicios para realizar la conexión a la red de repositorios, solicitar los objetos de aprendizaje, recuperarlos y presentarlos a los usuarios dentro de su ambiente de aprendizaje, de forma transparente al repositorio en el cual se encuentran almacenados. La red de repositorios será la encargada de realizar las labores necesarias para garantizar la localización y entrega de los objetos de aprendizaje a los LMS.

El modelo plantea además que cada repositorio que cumpla con la interfaz requerida y que pretenda formar parte de la red se puede registrar con cualquier miembro activo de ésta. Al ingresar a la red de repositorios se crea un conjunto

de registros en el catálogo de distribución local y otros en los repositorios de la red. Con el fin de evitar la duplicidad de los objetos de aprendizaje se utilizan identificadores globales únicos (GUID) para los objetos de aprendizaje dentro de toda la red de repositorios. Este esquema permite que un LMS pueda buscar y recuperar objetos de aprendizaje sin importar en cuál repositorio se encuentre almacenado físicamente, ya que los miembros de la red deberán implementar los mecanismos para la recuperación de los objetos de aprendizaje o meta-datos y su respectiva entrega. De la misma forma, al registrar un nuevo objeto de aprendizaje se debe replicar la información a todos los miembros de la red de repositorios. Para ello cada repositorio deberá implementar los servicios requeridos: Gather, que le permita recuperar la información de los objetos de aprendizaje y meta-datos de los demás repositorios, y Publish, para proporcionar la información de sus objetos de aprendizaje a los demás repositorios.

## ARQUITECTURA DE SERVICIOS WEB

A partir de la especificación IMS DRI se definió una serie de requerimientos que deben cumplir los repositorios que se integren a EduLatin. Estos requerimientos fueron planteados en términos de los servicios que deben ser implementados por los repositorios, y que les permite interactuar con los LMS y con otros repositorios, a nivel de servicios web. Los requerimientos más importantes y su relación con IMS DRI se pueden observar en la **Figura 4** y se describen a continuación:

**\* Estructura de solicitudes y respuestas:** Con el fin de proporcionar un alto grado de interoperabilidad entre diversas implementaciones de repositorios digitales, se tomó la decisión de utilizar mensajes SOAP simples para realizar las solicitudes y/o emitir las respuestas. Estos mensajes utilizan tipos definidos en el documento WSDL de la interfaz pública de los repositorios, que permiten ocultar la funcionalidad de éstos y buscan que los usuarios no realicen ningún tipo de suposición acerca de la estructura de la base de datos en la cual se almacenan los recursos y sus meta-datos. Los servicios planteados son del tipo solicitud/respuesta sin manejo de estado, pero el modelo plantea el futuro uso de otro tipo de mensajes.

**\* Autenticación de usuarios:** Uno de los aspectos más importantes del modelo de interoperabilidad de repositorios que no está especificado completamente dentro de la especificación IMS DRI es la capacidad que deben tener los repositorios para reconocer a los usuarios que hacen uso de sus servicios, de forma independiente a si éstos son sistemas de gestión de aprendizaje, aplicaciones basadas en servicios web u otros repositorios. Para cumplir con este objetivo se definió una política de seguridad basada en claves de usuario similar a la utilizada por el servicio de búsqueda de Google [18], en la cual cada usuario que desea utilizar los servicios de un repositorio de la red debe ser autenticado por medio de una llave. Esta llave se obtiene por medio de un proceso simple de registro, y debe ser pasada como parámetro a cada invocación de los servicios, debido a que las operaciones planteadas dentro del modelo no tienen estado.

**\* Información de los repositorios:** Este servicio permite obtener información acerca de los esquemas de meta-datos que soporta cada repositorio que pertenece a la red EduLatin, y puede ser utilizado posteriormente para estructurar las solicitudes de búsqueda, adición de recursos y administración de meta-datos. El usuario deberá especificar su clave única, y el repositorio responderá a la solicitud con una lista de los modelos de referencia y sus

componentes soportados, así como las versiones de los esquemas de meta-datos.

**\* Búsqueda y localización de recursos:** Este servicio está basado en la función Search/Expose definida en IMS DRI, y permite buscar y localizar recursos dentro de los repositorios que conforman la red EduLatin. Para realizar una búsqueda, el usuario establece una conexión con cualquier repositorio de la red en el cual se encuentra suscrito, y debe especificar un patrón, su llave única y el tipo de recursos que desea localizar (Assets, SCO o agregaciones de contenidos). Con estos parámetros el repositorio que recibe la solicitud realiza una búsqueda en su base de datos local, al mismo tiempo que reenvía la solicitud a los repositorios en los cuales se encuentra registrado. Una vez que se obtienen los resultados, estos son enviados al usuario en forma de un documento XML en el cuerpo del mensaje SOAP. Dentro de la información retornada al usuario se encuentra el identificador único de cada objeto localizado, su nombre, localización y tipo, definido en términos de la especificación que utilice cada repositorio. Con estos datos el usuario puede realizar otras solicitudes al repositorio, tales como obtener la información de sus meta-datos (gather/expose), o solicitar el recurso para su almacenamiento o visualización (request/deliver).

**\* Adición de nuevos recursos:** Este servicio sigue los lineamientos establecidos por la función Submit/Store de IMS DRI, y permite adicionar nuevos recursos dentro de un repositorio de la red EduLatin. Para hacer uso de esta función el usuario suscrito a cualquier repositorio de la red envía la solicitud submit, en la cual incluye su llave única, el tipo de recurso a almacenar, una dirección URL en donde se encuentra el recurso (empaquetado de acuerdo a la especificación IMS Content Packaging) y una información básica acerca del recurso. Al recibir estos parámetros, el repositorio debe obtener el recurso (por medio de una solicitud HTTP o FTP a la dirección URL proporcionada por el usuario), verificar sus contenidos y almacenarlo dentro de su propia estructura de directorios. Una vez que el recurso ha sido transferido correctamente, el repositorio determina el esquema de meta-datos al cual pertenece el nuevo recurso, y llena los campos clave del esquema de meta-datos correspondiente con la información básica proporcionada por el usuario. Finalmente el repositorio retorna un mensaje de éxito o fracaso al usuario (Store), en el cual especifica el identificador global del recurso almacenado y su URL, o un error, según el caso.

**\* Adición / actualización de meta-datos:** Este servicio permite adicionar o actualizar los meta-datos de un recurso dentro del repositorio. Para hacer uso de este servicio, el

usuario debe especificar su llave única, el identificador del recurso y los meta-datos en forma de un string XML. El repositorio toma estos datos y luego de realizar una verificación de la estructura de los meta-datos (de acuerdo con el tipo definido para el recurso) elimina la información anterior y llena el modelo de datos con los nuevos valores.

**\* Publicación de meta-datos:** Este servicio permite obtener el registro de meta-datos asociado con un recurso dentro de un repositorio, y puede ser utilizado por los sistemas de gestión de aprendizaje o por otros repositorios. Se estima que en un futuro este servicio podrá ser extendido para soportar las funciones "Push" Gather y "Pull" Gather, que permitirán compartir los meta-datos entre los repositorios que integrarán la red EduLatin.

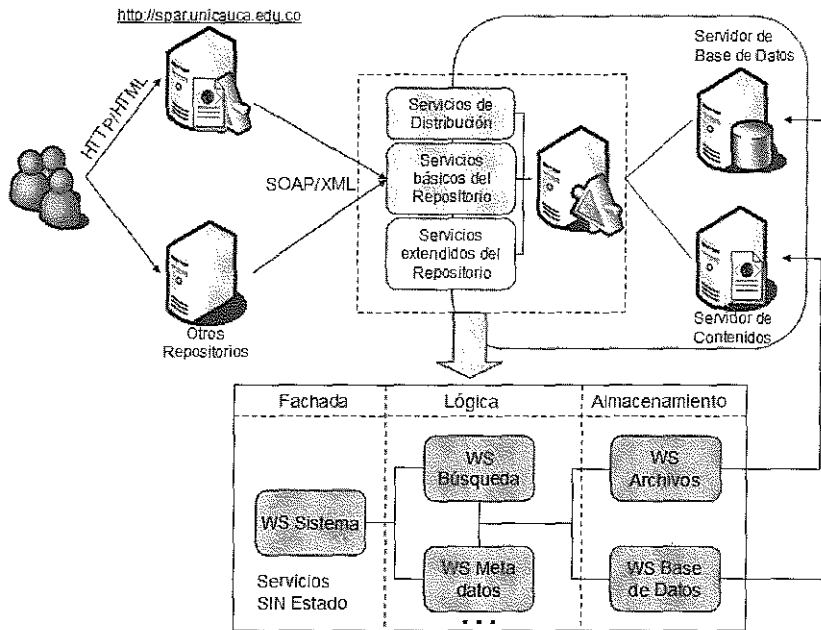


Figura 4 Arquitectura de los Servicios Web XML e interacción con otros sistemas

En la Figura 5 se presenta el "Examinador de objetos" de Visual Studio .NET mostrando la descripción de los métodos públicos del repositorio SPAR 1.0 con base en la arquitectura anteriormente definida. Se muestra por ejemplo,

el método `BuscarRecursos(string)` que permite buscar un recurso que cumpla con una condición de consulta específica.

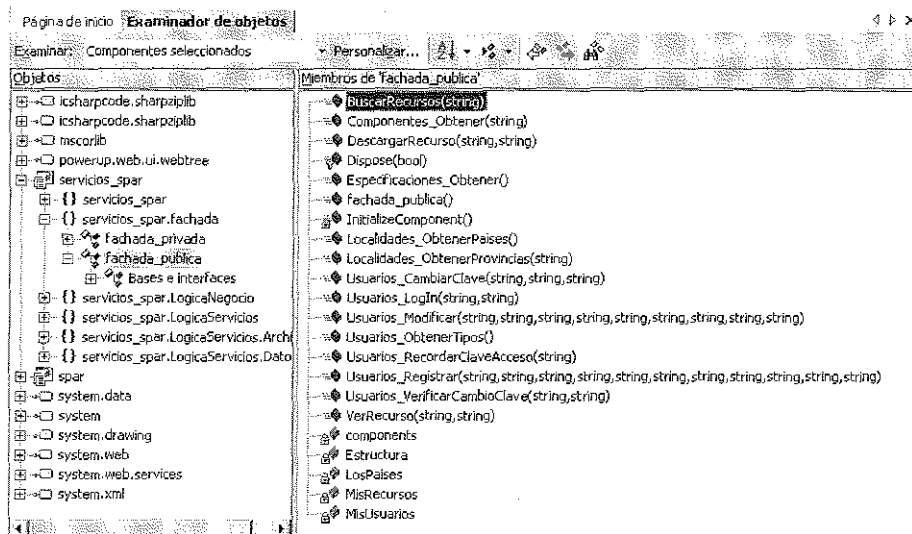


Figura 5 Métodos públicos de los servicios del repositorio

Aunque los servicios públicos dan servicio a cualquier petición externa (de un LMS o de otro repositorio), el repositorio cuenta con unos métodos privados optimizados para dar soporte a las funciones de los usuarios en la interfaz Web del repositorio (que se explican más adelante en la sección "HERRAMIENTA SOFTWARE SPAR 1.0"

## MODELO FÍSICO PARA EL ALMACENAMIENTO DEL OS META-DATOS

En el marco del modelo de datos, se realizó una revisión del manejo de esquemas y documentos XML en campos texto de la base de datos, en archivos planos y el almacenamiento en un sistema de bases de datos XML nativo [19][20]. En esta revisión se encontraron diferentes productos tecnológicos, entre los que se encuentran Ariadne, ImseVisme, LOM editor, EUN, ADL SCORM, Reggie, EM2, Reload Editor y Meta-data Tool [21][22][23][24]. Estos últimos con interfaces variadas, desde aplicaciones cliente independientes "standalone", hasta aplicaciones Web integradas con repositorios existente, que permiten importar y exportar los meta-datos en forma de documentos XML. Del anterior estudio se descartó el almacenamiento de los documentos en archivos planos, debido al rendimiento que debe tener el repositorio. También se descartó el uso de bases de datos XML nativas (XQuery y XPath) debido a que en la actualidad no todos los motores cuentan con esta característica, y se requiere que el modelo de datos planteado en el repositorio se pueda instanciar en la mayor cantidad posible de motores de bases de datos.

Con SPAR también se logró establecer un modelo relacional que permite almacenar cualquier tipo de especificación o estándar de meta-datos para SCOs (no sólo SCORM 1.2 y LOM 1.0) sin ningún tipo de restricción en cuanto a multiplicidad de datos a digitar, ni el manejo de elementos del modelo que no puedan ser mapeados a XML, como las tarjetas virtuales personales y organizacionales (VCARD), como sucede con otras aplicaciones (Reload Editor entre ellas). La aplicación Web desarrollada crea dinámicamente los formularios de entrada de datos de los meta-datos, basado en los campos y atributos de la especificación (esquema) que el usuario seleccione para un objeto de aprendizaje determinado.

Luego del estudio de estos requerimientos, se decidió retomar la idea de Meta-data Model (Meta-data tool de Learning Systems Architecture Lab, Carnigie Mellon University), realizando un modelo relacional que permite almacenar diferentes esquemas de meta-datos (probado actualmente con SCORM 1.2, LOM 1.0 y VCARD 1.0) y los valores de los objetos de aprendizaje que se almacenen en el repositorio. Este modelo almacena la jerarquía de los esquemas y de los documentos XML en tablas auto reflexivas (recursivas). Es preciso tener en cuenta que con el uso de bases de datos con soporte nativo para XML este manejo puede ser más sencillo, pero no es posible garantizar que se ofrezca un mejor desempeño. Este análisis se desarrollará en una etapa posterior del proyecto, ya que la mayoría de pruebas o comparaciones entre bases de datos XML nativas y bases de datos que almacenan XML [25] en campos de texto no aplican para el modelo propuesto.

A continuación se presenta el modelo físico (según la herramienta case Power Designer [26]) para el almacenamiento de los meta-datos, el cual soporta cualquier especificación de meta-datos que existe en la actualidad y que se basa en el estándar LOM 1.0 [2]. Para una mejor comprensión del mismo se ha dividido en cinco partes de forma que se puedan apreciar claramente las diferentes tablas y relaciones del modelo. Las partes se han dividido de acuerdo a la funcionalidad que cumplen dentro del modelo.

De tal forma, en la **Figura 6** se muestra el modelo relacional a nivel físico de las tablas (incluyendo los campos, llaves primarias, llaves foráneas, tipos de datos) y las relaciones que permiten almacenar diferentes especificaciones (por ejemplo SCORM 1.2 [1], LOM 1.0 [2] y VCARD 1.0 [17] para la versión actual de SPAR) y en la **Tabla 1** se hace una breve descripción de dichas tablas sin hacer énfasis en los diferentes campos o columnas de las tablas (ver documentación técnica en <http://spar.unicauca.edu.co>)



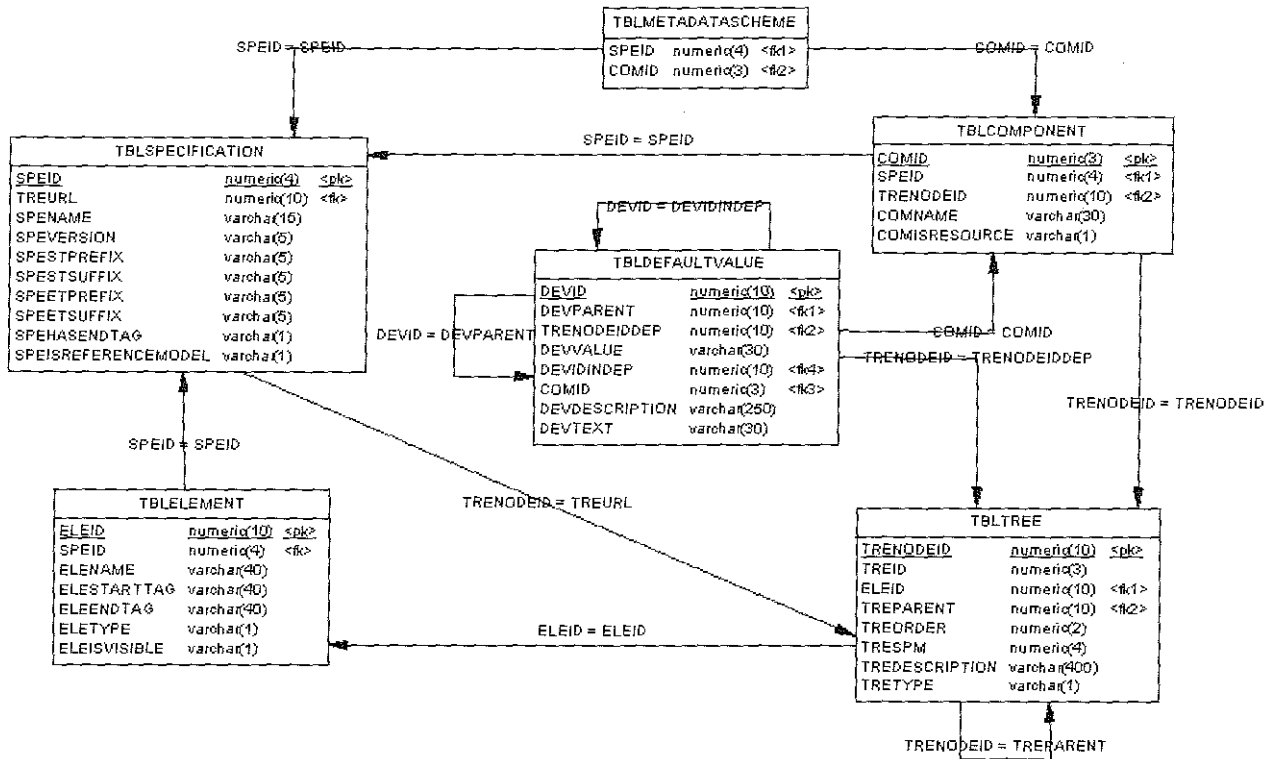


Figura 6. Persistencia de la especificación y sus elementos

Tabla 1. Descripción de las tablas relacionadas con la persistencia de una especificación

Nombre de la Tabla	Descripción
TBLSPECIFICATION	Almacena los registros de las diferentes especificaciones o estándares de meta-datos. Define como se exportan las etiquetas de los elementos a archivos XML.
TBLCOMPONENT	Almacena los diferentes componentes de una especificación, por ejemplo para la especificación SCORM 1.2 almacena Sco, Asset y Content Aggregation (curso).
TBLMETA-DATAScheme	Relación que se establece entre las tablas TBLSPECIFICATION y TBLCOMPONENT para determinar el esquema de meta-datos que cumplen los diferentes componentes de una determinada especificación.
TBLELEMENT	Almacena los diferentes elementos (similares a etiquetas) de cada especificación o estándar, los cuales se encuentran asociados a diferentes nodos de la tabla TBLTREE.
TBLTREE	Esta tabla almacena la estructura jerárquica de las especificaciones. En ella se registran las relaciones de contención (con una relación auto reflexiva) que existe entre los diferentes elementos contemplados dentro de las especificaciones almacenadas en la tabla TBLSPECIFICATION.
TBLDEFAULTVALUE	Almacena los vocabularios que aplican a ciertos elementos de las especificaciones. Cuenta con dos relaciones auto reflexivas; la primera para establecer las relaciones de contención que existen entre los vocabularios, y la segunda para informar que un campo toma datos de acuerdo al valor seleccionado en otro campo, cuyo valor depende del valor seleccionado previamente.

En la **Figura 7** y en la **Tabla 2** se muestra y describe el modelo relacional físico de las tablas y las relaciones que permiten almacenar los datos registrados por el usuario

para un objeto de aprendizaje de acuerdo a una especificación.

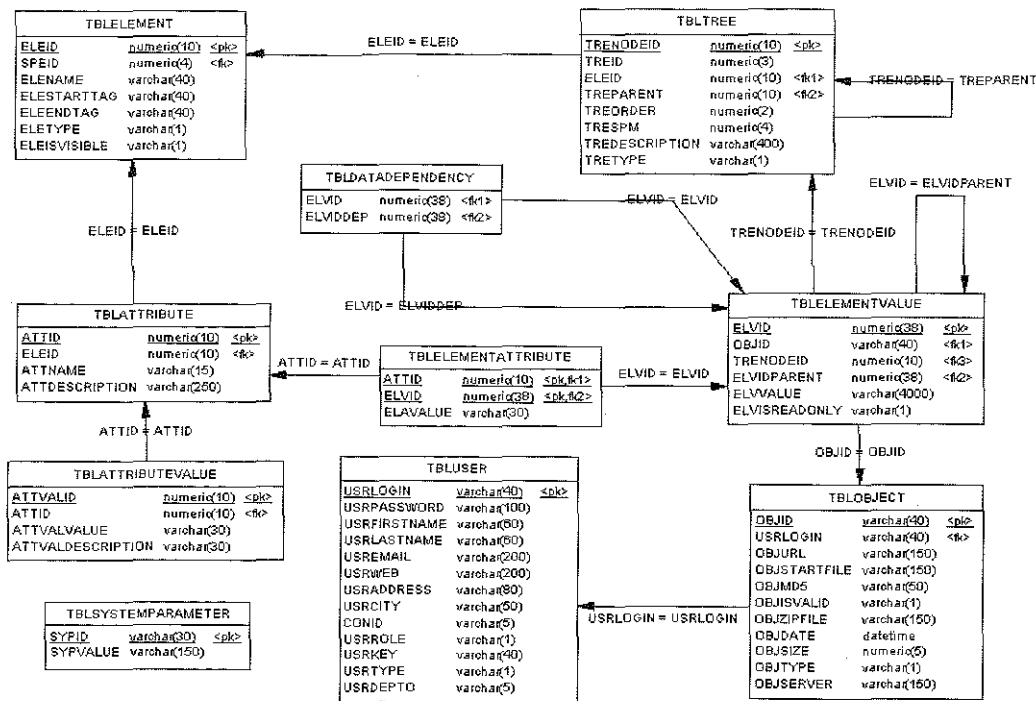


Figura 7. Persistencia de los meta-datos registrados por un usuario para un objeto de contenido

Tabla 2. Descripción de las tablas relacionadas con la persistencia de los meta-datos de un objeto de contenido.

Nombre de la Tabla	Descripción
TBLTREE	Esta tabla almacena la estructura jerárquica de las especificaciones. En ella se registran las relaciones de contención (con una relación auto reflexiva) que existe entre los diferentes elementos contemplados dentro de las especificaciones almacenadas en la tabla TBLSPECIFICATION.
TBLELEMENT	Almacena los diferentes elementos (similares a etiquetas) de cada especificación o estándar, los cuales se encuentran asociados a diferentes nodos de la tabla TBLTREE.
TBLATTRIBUTE	Almacena los atributos de los elementos dentro de la especificación que lo requieren, por ejemplo el atributo XML:LANG de los elementos LANGSTRING de la especificación SCORM 1.2.
TBLATTRIBUTEVALUE	Almacena los diferentes valores que puede tomar un atributo de un elemento, por ejemplo ES (Español) para el atributo XML:LANG de la especificación SCORM 1.2.
TBLELEMENTVALUE	Esta tabla almacena el registro de los valores (datos) de los meta-datos de cada objeto de aprendizaje. Cuenta con una relación auto reflexiva que le permite tener una relación de los valores almacenados en la jerarquía de la especificación, esta relación genera una sobrecarga de almacenamiento, pero con ella se permite dar cabal cumplimiento al almacenamiento de todas las multiplicidades y valores de los meta-datos.
TBLELEMENTATTRIBUTE	Almacena los valores específicos de los atributos para los elementos que cuentan con atributos para cada objeto de aprendizaje.
TBLDATADEPENDENCY	Almacena los registros de datos que dependen en su valor de otros valores registrados para cada objeto de aprendizaje.
TBLSYSTEMPARAMETER	Almacena parámetros básicos del sistema, como por ejemplo la dirección actual donde se están almacenando los Objetos de Aprendizaje. Esto permite el balanceo de la carga en los discos del sistema de archivos del servidor o de los servidores de contenido.
TBLOBJECT	Almacena la información básica de los objetos de aprendizaje (SCOs, Assets y Content Aggregations) del repositorio, como es el identificador único global (GUID), la URL donde se almacenó en el sistema de archivos, el archivo de inicio para su visualización (por ejemplo para múltiples archivos html e imágenes en una carpeta), y la huella del Objeto con MD5 para evitar el registro de Objetos de Aprendizaje repetidos.
TBLUSER	Almacena los datos de los usuarios del sistema, entre ellos el login, la clave autenticada con MD5, los nombres, el correo electrónico y la URL de la página Web.

En la **Figura 8** y en la **Tabla 3** se muestra y describe el modelo relacional físico de las tablas y las relaciones que permiten almacenar la clasificación taxonómica (Ingenierías, Ciencias de la Tierra, Artes y otras) que presenta un objeto

de aprendizaje y el componente de la especificación al cual pertenece dicho objeto.

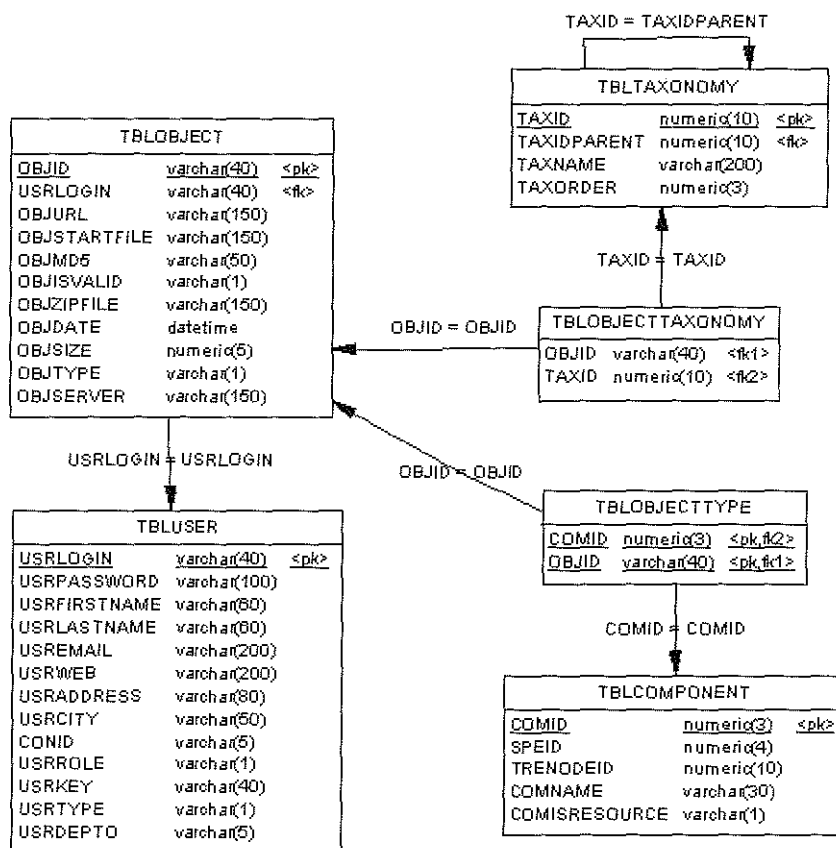


Figura 8. Persistencia de la clasificación taxonómica de los objetos

Tabla 3. Descripción de las tablas relacionadas con la persistencia de la clasificación taxonómica de los objetos

Nombre de la Tabla	Descripción
TBLUSER	Almacena los datos de los usuarios del sistema, entre ellos el login, la clave autenticada con MD5, los nombres, el correo electrónico y la URL de la página Web.
TBLOBJECT	Almacena la información básica de los objetos de aprendizaje (SCOs, Assets y Content Aggregations) del repositorio, como es el identificador único global (GUID), la URL donde se almacenó en el sistema de archivos, el archivo de inicio para su visualización (por ejemplo para múltiples archivos html e imágenes en una carpeta), y la huella del Objeto con MD5 para evitar el registro de Objetos de Aprendizaje repetidos.
TBLCOMPONENT	Almacena los diferentes componentes de una especificación, por ejemplo para la especificación SCORM 1.2 almacena Sco, Asset y Content Aggregation (curso).
TBLOBJECTTYPE	Permite definir de qué tipo de componente de una especificación es un objeto. Esto a su vez permite generar el árbol y los formularios para la entrada de datos en la interfaz de usuario.
TBLOBJECTTAXONOMY	Almacena los diferentes objetos de contenido con su respectiva ruta taxonómica (Clasificación).
TBLTAXONOMY	Almacena la clasificación taxonómica que puede ser asociada a un objeto de aprendizaje en base a diversas áreas del conocimiento en las que puede ser enmarcado dicho objeto. Esta tabla tiene una relación auto reflexiva que permite almacenar las relaciones de contención que existen dentro de la taxonomía elegida.

En la **Figura 9** y en la **Tabla 4** se muestra y describe el modelo relacional físico de las tablas y las relaciones que permiten almacenar la información relacionada con los usuarios del repositorio de datos, las estadísticas de acceso y la calificación de los objetos de contenido que presentan los diferentes objetos de contenido dentro del

repositorio de datos. Estas estadísticas permiten que en una versión futuro se pueda implementar un sistema de toma de decisiones, con respecto al perfil de los usuarios, la demanda de los contenidos y la calidad de los contenidos de los repositorios.

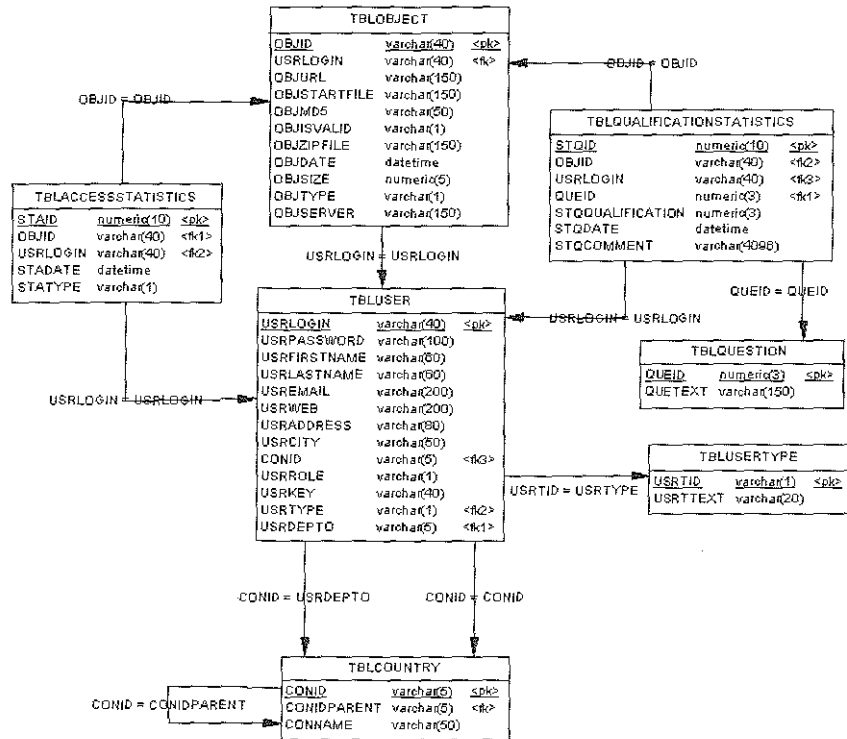


Figura 9. Persistencia de la información de usuario y estadísticas de acceso

Tabla 4. Descripción de las tablas relacionadas con la persistencia de la información de usuario y estadísticas de acceso

Nombre de la Tabla	Descripción
TBLUSER	Almacena los datos de los usuarios del sistema, entre ellos el login, la clave autenticada con MD5, los nombres, el correo electrónico y la URL de la página Web.
TBLOBJECT	Almacena la información básica de los objetos de aprendizaje (SCOs, Assets y Content Aggregations) del repositorio, como es el identificador único global (GUID), la URL donde se almacenó en el sistema de archivos, el archivo de inicio para su visualización (por ejemplo para múltiples archivos html e imágenes en una carpeta), y la huella del Objeto con MD5 para evitar el registro de Objetos de Aprendizaje repetidos.
TBLACCESSSTATISTICS	Almacena estadísticas de acceso y manejo de los objetos de aprendizaje por parte de los usuarios del sistema, incluido el usuario anónimo/público.
TBQUALIFICATIONSTATISTICS	Almacena registros de evaluación cualitativa (comentarios) y cuantitativa (de cero a cinco) de los objetos de aprendizaje, con el objeto de ir día a día estableciendo un ranking de cada uno de ellos cuando son consultados por otros usuarios.
TBLQUESTION	Almacena las preguntas que se han definido para preguntar a un usuario, que visualice un objeto de aprendizaje, sobre la calidad del mismo para establecer el ranking de los objetos de aprendizaje almacenados en el repositorio de datos.
TBLCOUNTRY	Almacena los diferentes países del mundo de acuerdo con el estándar ISO 3166. Esta tabla tiene una relación auto reflexiva que permite almacenar las relaciones de contención que existen entre los diferentes países con sus respectivas provincias, estados o departamentos.
TBLUSERTYPE	Almacena los registros de los diferentes tipos de usuario que pueden acceder al repositorio de datos, los cuales pueden ser de género masculino o femenino cuando el usuario es una persona, o bien una aplicación cuando el usuario es un LMS (Learning Management System – Sistema de Gestión de Aprendizaje).

En la **Figura 10** y en la **Tabla 5** se muestra el modelo relacional físico de las tablas y las relaciones que permiten establecer los diferentes elementos que son requeridos para las diferentes especificaciones y para el repositorio

de datos de SPAR. Con estas tablas se establece una relación entre campos de las especificaciones que tienen un significado fundamental para el uso del repositorio

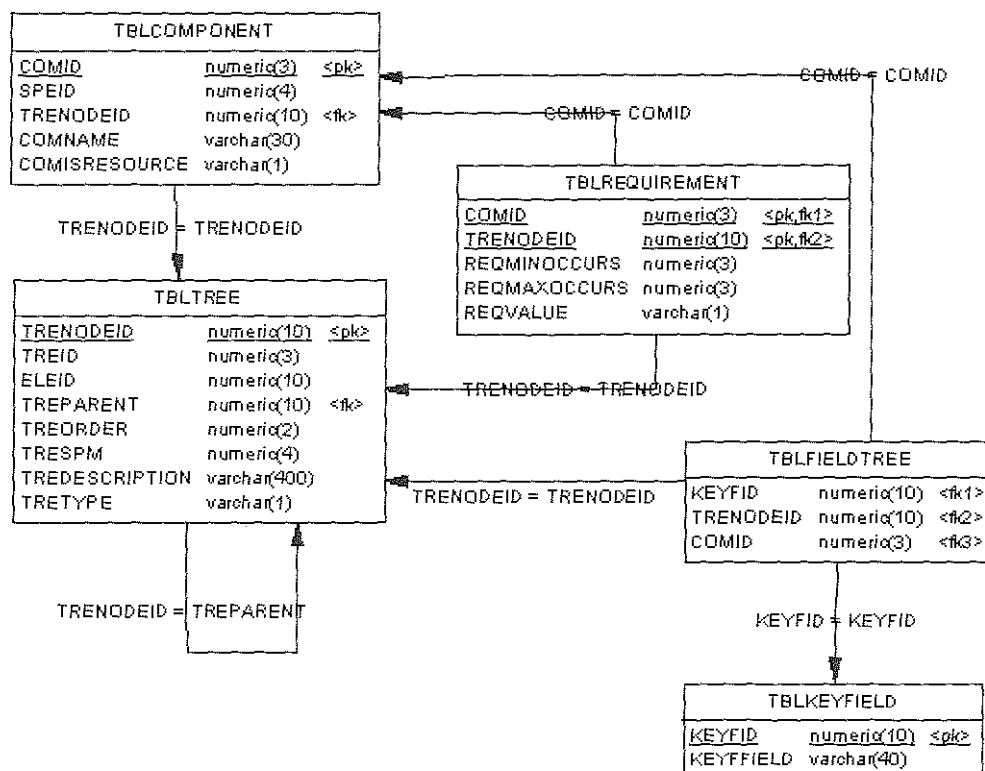


Figura 10. Persistencia de los elementos requeridos por las especificaciones

Tabla 5. Descripción de las tablas relacionadas con la persistencia de los elementos requeridos por las especificaciones


Nombre de la Tabla	Descripción
TBLTREE	Esta tabla almacena la estructura jerárquica de las especificaciones. En ella se registran las relaciones de contención (con una relación auto reflexiva) que existe entre los diferentes elementos contemplados dentro de las especificaciones almacenadas en la tabla TBLSPECIFICATION.
TBLCOMPONENT	Almacena los diferentes componentes de una especificación, por ejemplo para la especificación SCORM 1.2 almacena Sco, Asset y Content Aggregation (curso).
TBLREQUIREMENT	Almacena la información de los elementos de la jerarquía de la especificación que son obligatorios para cada componente de la especificación, por ejemplo el elemento CATALOG ENTRY es obligatorio para SCOs y CONTENT AGGREGATION, pero es opcional para los ASSETS dentro de la especificación SCORM 1.2.
TBLKEYFIELD	Almacena los registros de los elementos de la jerarquía de la especificación que son campos claves para el repositorio de datos y por tal razón obligatorios en el momento de insertar los meta-datos de los diferentes objetos de aprendizaje almacenados en el repositorio de datos.
TBLFIELDTREE	Establece la relación entre los campos claves para el repositorio, los componentes de las especificaciones y los nodos de los árboles de especificaciones para un componente, de forma que se pueda manejar adecuadamente la obligatoriedad del registro de estos campos.

## HERRAMIENTA SOFTWARE SPAR 1.0

A finales del mes de Febrero de 2006 se realizó la liberación/entrega de la primera versión operativa y estable del repositorio de datos SPAR [12][13][14] con una interfaz Web (ver **Figura 11**) que permite a los usuarios reales y potenciales de SPAR acceder al repositorio y usar los servicios Web implementados dentro de la aplicación, los cuales acceden a los registros de las diferentes tablas del modelo físico de la base de datos descrito anteriormente.

Todas las páginas desplegadas dentro de la aplicación presentan un encabezado en el cual se muestra el nombre de la aplicación (Repositorio de acceso público basado en SCORM para el idioma español y SCORM Public-Access Repository para inglés) con un menú de navegación que permite que los diferentes usuarios de SPAR utilizar los servicios que se prestan dentro del sitio Web (ver **Figura 11**), entre ellos, Iniciar sesión, Cambiar de Idioma (Inglés, Español), Subir un recurso, Modificar mis recursos, Buscar recursos, Referenciar un recurso externo y Descargar el software del repositorio para montar un repositorio propio. Aunque no se pretende describir todos los servicios que presta SPAR se presentará a continuación algunas interfaces y su descripción.

Bienvenido a SPAR Entrar



### Repositorio de acceso público basado en SCORM

Escuela Alumnos y Compañía Objetos de Aprendizaje

[Inicio](#) [Idioma](#) [Subir recurso](#) [Mis recursos](#) [Buscar recursos](#) [Referenciar recurso](#) [Descargas](#)

---

**Ingresar al sistema**

Nombre:

Clave:

[¿Olvidó su clave?](#)

**Los mejor calificados**

[DoS y DDoS mas](#)  
☆☆☆☆☆

**Los más accedidos**

[DoS y DDoS\(?\)](#)  
mas

**Historia de la Informatica 500 A.C. hasta 1949 D.C. (5)**  
[mas](#)

**Clasificación**

<a href="#">Artes (0)</a>	<a href="#">Negocios (0)</a>	<a href="#">Educación (0)</a>
<a href="#">Humanidades (0)</a>	<a href="#">Matemáticas y Estadísticas (0)</a>	<a href="#">Ciencia y Tecnología (3)</a>
<a href="#">Ciencias sociales (0)</a>		

**Recursos encontrados:**

**DoS y DDoS**  
Presentación que habla acerca de los diferentes tipos de ataques de denegación de servicios realizados actualmente. Tambien presenta algunas de las herramientas mas utilizadas en este tipo de ataques  
es  
Asset  
22/02/2006 18:21:26  
☆☆☆☆☆  
[Descargar recurso mas](#)

**Historia de la informatica 500 A.C. hasta 1949 D.C.**  
Presentacion interactiva de la historia de la informatica desde 500 a.c hasta 1949  
es  
Sc0  
24/02/2006 18:21:15  
☆☆☆☆☆  
[Descargar recurso mas](#)

**Figura 11.** Interfaz de Inicio de SPAR

Cuando un usuario inicia sesión dentro del repositorio, además de poder consultar los objetos mejor calificados, los más accedidos, o la totalidad de objetos almacenados por la clasificación taxonómica, puede subir sus propios recursos, cargar y modificar los meta-datos (que deben estar muy bien descritos para hacer que las búsquedas obtengan mejores resultados), descargar los contenidos a su máquina local, cambiar la contraseña y los datos propios de su perfil, entre otras cosas.

En la página correspondiente a la opción **Subir recurso** del menú de navegación se presentan los campos necesarios para subir un objeto de aprendizaje al repositorio de datos, una vez que se llenan los campos que aparecen en la interfaz, se sube el recurso al repositorio y

posteriormente se genera la jerarquía de los meta-datos según la especificación SCORM 1.2 que pueden ser editados en cualquier momento por el usuario que subió el recurso (ver **Figura 12**).

En la página correspondiente a la opción **Mis recursos** del menú de navegación se presentan los diferentes objetos de aprendizaje que el usuario ha subido al repositorio. Estos objetos

Bienvenido a SPAR: [trunksdtz](#) [Salir](#)

**Repositorio de acceso público basado en SCORM**  
Usado, Afiliación y Campaña Objeto de Aprendizaje

Inicio Idioma Subir recurso Mis recursos Buscar recursos Referenciar recurso Descargas

		Modifica Metadatos			
		Elemento	Descripción	Insertados	Máximo
<ul style="list-style-type: none"> <li>SCORM 1.2</li> <li>General               <ul style="list-style-type: none"> <li>Título</li> <li>Datos</li> <li>Catalogo de Entrada                   <ul style="list-style-type: none"> <li>Catalogo</li> <li>Entrada                       <ul style="list-style-type: none"> <li>Datos</li> </ul> </li> <li>Lenguaje</li> <li>Descripción</li> <li>Datos</li> <li>Palabras Claves</li> </ul> </li> <li>Ciclo de Vida                   <ul style="list-style-type: none"> <li>Versión</li> <li>Estado</li> </ul> </li> <li>Meta-Metadata</li> <li>Información Técnica</li> <li>Derechos</li> <li>Relación</li> </ul> </li> </ul>	Insertar	Información Educativa	Esta categoría describe las características claves educacionales o pedagogicas de este recurso	0	1
	Insertar	Relación	Esta categoría define la relacion entre este recurso y otros recursos	1	100
	Insertar	Anotación	Esta categoría provee comentarios sobre el uso educacional de este recurso, quien creo esta anotacion y cuando, cuando multiples anotaciones son necesitadas multiples instancias de esta categoria	0	30

**Descripción**

Raiz de la especificacion SCORM 1.2

**Figura 12.** Interfaz para edición de los meta-datos de un objeto de aprendizaje según la especificación SCORM 1.2.

pueden ser vistos, editados, reemplazados o descargados por el usuario.

En la página correspondiente a la opción **Buscar recursos** del menú de navegación se presenta la opción de que un usuario del repositorio pueda buscar algún recurso mediante una búsqueda por **frase exacta**, en donde se presentarán al usuario los objetos de aprendizaje que coincidan con la frase exacta de búsqueda ingresada en el campo correspondiente para la búsqueda.

En la página correspondiente a la opción **Referenciar recurso** del menú de navegación se presentan los campos requeridos para que un usuario inserte en el repositorio una referencia a una página Web que quiera vincular dentro del repositorio para ser accedida por otros usuarios del repositorio.

En la página correspondiente a la opción **Descargas** del menú de navegación se presentan las descargas disponibles de SPAR diferentes a los objetos de aprendizaje que se encuentran almacenados en él. Dentro de las descargas disponibles se encuentran: Guía de instalación y configuración, Base de datos y Aplicación; la primera descarga se puede realizar sin necesidad de aceptar la licencia Microsoft de fuente compartida, mientras que las otras dos descargas deben realizarse una vez se haya leído la licencia y aceptado los términos de la misma.

## ESTADO ACTUAL DEL PROYECTO Y TRABAJO FUTURO

La versión actual de SPAR presenta una interfaz Web que ofrece los siguientes servicios:

- \* Servicio de registro en el repositorio de datos SPAR.
- \* Para los usuarios no registrados: búsqueda y consulta de objetos de contenido de aprendizaje almacenados en el repositorio, calificación de los objetos de aprendizaje.
- \* Para los usuarios registrados: búsqueda y consulta de objetos de contenido de aprendizaje almacenados en el repositorio, carga de nuevos objetos de aprendizaje en el repositorio, edición de los meta-datos de los objetos de aprendizaje, referencia a otros recursos mediante su dirección URL, descarga de objetos de aprendizaje, descarga del código fuente de la herramienta software y calificación de los diferentes objetos de aprendizaje del repositorio.

Como trabajo futuro se tiene programado dos versiones, la primera corresponde a una mejora no sustancial y la corrección de errores detectados en los meses de Marzo-Mayo de 2006, esta corresponde a la versión 1.1 y otra versión que incluye el desarrollo de un sistema de toma de decisiones basado en Bodegas de Datos y Procesamiento Analítico En Línea, con Bases de Datos Distribuidas que corresponde a la versión 2.0. Las mejoras no sustanciales que se contemplan en la versión 1.1 son:

- \* **Mejoras en la interfaz (navegación, usabilidad, colores, fuentes y otros).** Por ejemplo, en la edición de los meta-datos que es la parte más difícil de comprender para los usuarios, por su novedad y tecnicismo.
- \* **Nueva funcionalidad:** Crear una nueva especificación de forma que SPAR pueda soportar múltiples estándares.
- \* **Nueva funcionalidad:** Soporte Multilinguaje para la aplicación (inglés en su totalidad).
- \* **Nueva funcionalidad:** Búsquedas avanzadas utilizando operadores lógicos y el manejo de subconjuntos.
- \* **Nueva funcionalidad:** Subir los meta-datos desde un archivo XML.
- \* **Nueva funcionalidad:** Validar que los meta-datos requeridos para un recurso esté totalmente diligenciados.
- \* **Reorganizar los servicios Web públicos** para que los usuarios puedan subir un recurso mediante estos

servicios, desde sistemas desarrollados por terceros, como por ejemplo Sistemas de Gestión de Aprendizaje u otros repositorios.

- \* **Desarrollar una interfaz** de consulta de recursos para dispositivos Pocket PC.

## CONCLUSIONES

Al momento de implementar o desarrollar una solución informática en el campo educativo y con miras a potenciar la industrialización de esta área del desarrollo humano, se debe tener presente los diferentes estándares y especificaciones que existen a nivel mundial y nacional en torno al tema. En el caso específico de los repositorios digitales, además de tener en cuenta la estandarización, se debe tener en cuenta que son el mecanismo ideal y natural para compartir el trabajo de profesores y estudiantes en todo el mundo, ya que los LMS, como por ejemplo Moodle, a pesar de potenciar la colaboración y la reutilización de recursos en los cursos, no son la herramienta idónea para compartir y reutilizar recursos entre distintos cursos, asignaturas e instituciones.

Los objetos de contenido de aprendizaje requieren ciertas características (accesibilidad, reusabilidad, interoperabilidad y durabilidad) para que puedan ser accedidos y consultados desde diferentes herramientas de consulta y acceso a objetos de aprendizaje y repositorios de datos, por este motivo el uso de una aplicación web es ideal, el visor web permite asociar los visores (Word, adobe, flash, otros) que están asociados a cada uno de los recursos/contenidos.

La base de datos diseñada e implementada se encuentra estable después de una serie de pruebas realizadas durante el ciclo de desarrollo del modelado de la base de datos que permite soportar la inserción y vinculación de cualquier especificación y estándar que se base en LOM 1.0 [2]. El modelo físico de la base de datos definido es más complejo en su diseño y uso frente al desarrollo de esta misma tarea en un sistema gestor de bases de datos con soporte para XML en forma nativa, pero este modelo se puede implantar sobre cualquier motor de base de datos, inclusive es útil para generar Editores LOM/SCORM en aplicaciones cliente independientes "standalone" que cumplan totalmente con el estándar o la especificación seleccionada (o varias al mismo tiempo).

La utilización de servicios Web permite que el repositorio de datos pueda ser accedido desde diferentes repositorios y LMS (Sistemas de Gestión de aprendizaje) que se



encuentran actualmente disponibles en el mercado libre y privado. Con lo anterior se facilita la adopción de la herramienta desarrollada en cualquier institución de educación o entrenamiento.

El modelo propuesto, y en especial SPAR cuenta con la capacidad de integrarse a otras iniciativas mundiales como GLOBE o CORDRA a través del esquema de una federación de repositorios. El proceso de integración exige el desarrollo de algunos servicios de registro en los servidores centrales de CORDRA (por ejemplo), pero con la arquitectura de servicios web desarrollada y el soporte en comunicación estándar (SOAP/XML) este trabajo será fácil de desarrollar.

Es preciso adelantar acciones con el objeto de que el modelo y la herramienta sean apropiadas en las distintas instituciones del país, empezando por el Ministerio de Educación Nacional y las Universidades (teniendo en cuenta el actual uso de Internet que se tiene en dichas instituciones). Además, proporcionar a la comunidad de desarrollo de Internet, el código fuente de la aplicación en un esquema de “control de código fuente distribuido” como por ejemplo CVS o VSS, para que este desarrollo se potencie con la participación de analistas, diseñadores, desarrolladores y probadores de todo el mundo; En este sentido se espera registrar el proyecto en <http://sourceforge.net>.

## AGRADECIMIENTOS

A la Universidad del Cauca (Colombia) y Microsoft Research en el marco del “2004 Microsoft Research eScience and SciData RFP Awards” por su apoyo económico y logístico.

## BIBLIOGRAFÍA

- [1] Advanced Distributed Learning. Sharable Content Object Reference Model (SCORMTM) Version 1.2; <http://www.adlnet.org/>, (Febrero 27 de 2006).
- [2] IEEE (Institute of Electrical and Electronics Engineers). Learning Technology Standards Committee (LTSC); <http://ieeeltsc.org/>, (Febrero 27 de 2006).
- [3] Looms, T., Christensen, C. Advanced Distributed Learning. Emerging and Enabling Technologies for the Design of Learning Object Repositories Report. Version 1.0. November 26, 2002.
- [4] IMS Digital Repositories Interoperability Specification Version 1.0; <http://www.imsglobal.org/digitalrepositories>, (Febrero 27 de 2006).
- [5] Rehak, D., Dodds P. and Lannom L. A Model and Infrastructure for Federated Learning Content Repositories. Proceedings, Interoperability of Web-Based Educational Systems Workshop at the 14th International World Wide Web Conference (WWW2005) May 10, 2005, Chiba, Japan. <http://www.lsal.cmu.edu/lsal/expertise/projects/cordra>.
- [6] CORDRA. Content Object Repository Discovery and Registration / Resolution Architecture; <http://cordra.lsal.cmu.edu/>, (Febrero 27 de 2006).
- [7] MERLOT. Multimedia Educational Resource for Learning and Online Teaching. <http://www.merlot.org>, (Febrero 27 de 2006).
- [8] EdNA. Educational Network Australia. <http://www.edna.edu.au/edna/page1.html>, (Febrero 27 de 2006).
- [9] ARIADNE. Alliance of Remote Instructional Authoring and Distribution Networks for Europe. Ariadne foundation for the European knowledge Pool. <http://www.ariadne-eu.org/index.html>, (Febrero 27 de 2006).
- [10] EduSource Canada. Candian Network of Learning Repositories. [http://www.edusource.ca/english/home\\_eng.html](http://www.edusource.ca/english/home_eng.html).
- [11] NIME. National Institute of Multimedia Education. <http://www.nime.ac.jp/index-e.html>.
- [12] SCORM Public-access repository. Descripción del proyecto SPAR; <http://prometeo.unicauca.edu.co/scorm/>, (Febrero 27 de 2006).
- [13] SPAR. Repositorio de acceso público basado en SCORM (Primera liberación); <http://spar.unicauca.edu.co/spar/>, (Febrero 27 de 2006).
- [14] COBOS, Carlos, MEZA, Erwin, MENDOZA, Martha, MORENO, Jorge, MUÑOZ, Norman. A Distributed Digital Content Repositories Model For Learning Objects in Latin-America. EATIS 2006 – Euro American

Conference on Telematics and Information Systems. Santa Marta, Colombia, Febrero 7 al 10 de 2006. ISBN: 958-8166-38-1 (Papel), 958-8166-36-5 (CD-ROM).

2004. CS497 Report. Report Id: 420. Department of Computer Science and Engineering, Indian Institute of Technology, Kanpur.

- [15] Learning Systems Architecture Lab. Carnegie Mellon University. <http://www.lsal.cmu.edu>, (Febrero 27 de 2006).
- [16] DCMI. Dublin Core Meta-data Initiative. <http://dublincore.org/>, (Febrero 27 de 2006).
- [17] Vcard Specification 1.0; <http://www.imc.org/pdi/>, (Febrero 27 de 2006).
- [18] Google Web APIs Reference. <http://www.google.com/apis/reference.html>
- [19] Dr. Thelma Looms, Clark Christensen. Advanced Distributed Learning. Emerging and Enabling Technologies for the Design of Learning Object Repositories Report. Version 1.0. November 26, 2002.
- [20] Qu, C., and Nejdil, W. (2004). Integrating XQuery-enabled SCORM XML Meta-data Repositories into an RDF-based E-Learning P2P Network. Educational Technology & Society, 7 (2), 51-60.
- [21] Sampson, D., Papaioannou, V. and Karadimitriou, P. EM<sup>2</sup>: An Environment for Editing and Management of Educational Meta-data. Educational Technology & Society 5 (4) 2002. ISSN 1436-4522. [http://ifets.ieee.org/periodical/vol\\_4\\_2002/sampson\\_papaioannou.html](http://ifets.ieee.org/periodical/vol_4_2002/sampson_papaioannou.html).
- [22] Meta-data Tool. Learning Systems Architecture Lab. Carnegie Mellon University. [http://www.lsal.cmu.edu/lsal/expertise/projects/compio/compio2001finalreport/scorm\\_tools/meta-data\\_tool.html](http://www.lsal.cmu.edu/lsal/expertise/projects/compio/compio2001finalreport/scorm_tools/meta-data_tool.html)
- [23] Zhen Yin, Zhengfang Xu and Abdulmoteleb El Saddik. Study of meta-data for advanced multimedia learning objects. CCGEI 2003 Proceedings.
- [24] Learning Technology. Volume 5 Issue 1. Publication of IEEE Computer Society Learning Technology Task Force (LTF), January 2003. ISSN 1438-0625. [http://ltf.ieee.org/learn\\_tech/issues/january2003/#9](http://ltf.ieee.org/learn_tech/issues/january2003/#9)
- [25] Anand Vivek Srivastava. Comparison and Benchmarking of Native XML Databases. November,