
6. Interoperabilidad de los Repositorios de Objetos de Aprendizaje a través de IMS

Como puede observarse en el listado de iniciativas del apartado 4.2.4, a nivel mundial se están llevando a cabo proyectos para la construcción de ROA, es de resaltar que gran número de las iniciativas son para formar redes con estos repositorios. El interés general de la comunidad está siendo enfocado a conectar y utilizar recursos distribuidos en repositorios heterogéneos (Hatala, et al., 2004), se está trabajando en corregir las divergencias de los repositorios para canalizarse en el uso de los mismos estándares o al menos el uso de los que sean compatibles con otros. También se está trabajando en la creación de repositorios federados (Massart & Dung, 2004), que se basan en búsquedas propagadas en metadatos distribuidos en distintos servidores. El objetivo final es la interoperabilidad, es decir, que los sistemas tengan la capacidad para trabajar fácilmente con otro u otros. Entre estos sistemas se busca, a través de una normalización tecnológica o procedimental, poder interconectarse para realizar diversas transacciones o actividades, siendo la principal actividad el intercambio de contenidos.

El grupo de especificaciones más ambicioso en *e-learning* y que está teniendo trabajos específicos para la estandarización de los repositorios es IMS, que dentro de sus propuestas (descritas en el apartado 5.4) maneja *IMS Digital Repositories Specification* y otras especificaciones que se relacionan con ésta, haciendo posible, con su implementación, la interoperabilidad de los repositorios en los ambientes *e-learning* y fuera de éstos.

A continuación, como contenido de este capítulo, se describe con detalle esta especificación, además de *IMS Content Packaging*, *IMS Learning Resource Metadata* e *IMS Resource List Interoperability*, que se considera son las especificaciones IMS relevantes para la construcción e interoperabilidad de los ROA. En cada especificación se hace hincapié en los requisitos de conformidad (*conformance*) que son necesarios para asegurar la interoperabilidad de la implementación del estándar. Por conformidad se entienden las formalidades que hacen que un sistema debe cumplir para apegarse

íntegramente a lo dictado por un estándar o, en este caso, una especificación. En el último apartado se presenta un modelo interoperable de repositorios que ejemplifica cómo se integran estas especificaciones para facilitar la comunicación de contenidos y datos entre los sistemas de un entorno *e-learning*.

6.1. *IMS Learning Resource Metadata*

Esta especificación (Mckell & Thropp, 2001) se considera importante para la construcción de los repositorios ya que se ha mencionado que los OA requieren de los metadatos para poder ser localizados y potencialmente reutilizados, si no se cuenta con metadatos es imposible la generación de catálogos organizados de forma homogénea y hasta cierto nivel también proveen un aporte semántico del contenido del recurso, lo cual le dará capacidades para aplicaciones futuras.

Esta especificación se desarrolla estrechamente ligada a LOM y describe los nombres, las definiciones, organización y restricciones de los elementos del esquema de metadatos propuesto por IMS, que tiene cambios mínimos respecto a lo propuesto por LOM. En el modelo de datos se mantiene la estructura jerárquica de árbol (Figura 12), en donde hay un solo elemento raíz (lom) del que se derivan ramas que son las categorías y sus elementos. Las hojas son las instancias (datos o valores) de cada elemento.

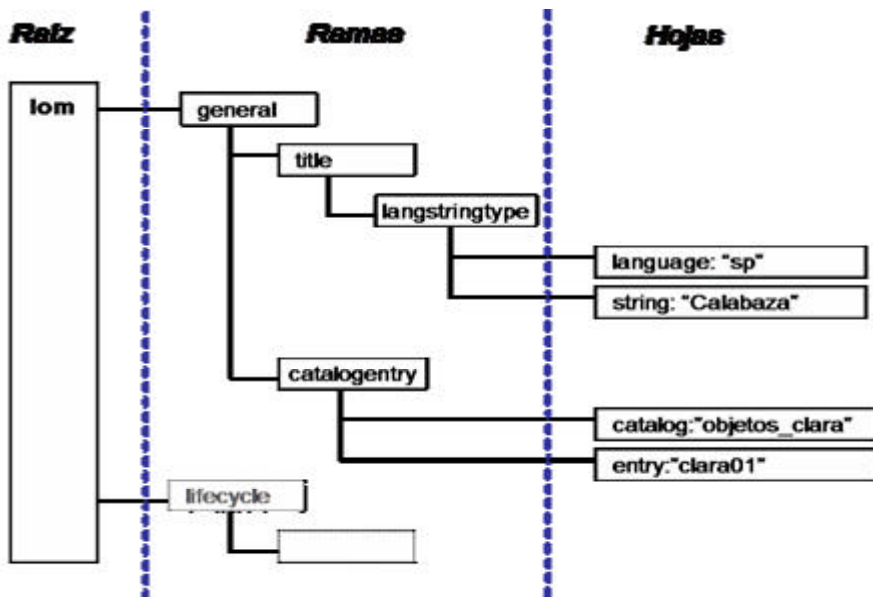


Figura 12. Esquema conceptual de la jerarquía de árbol de los metadatos

Dentro del Modelo de Información, cada elemento es descrito con:

Nombre: cómo el elemento debe ser escrito.

Explicación: la definición del elemento.

Multiplicidad: cómo se registra el contenido de los elementos y su orden de significancia.

Dominio: limitantes del vocabulario que puede incluirse.

Tipo: puede ser textual, numérico o fecha, así como sus restricciones de tamaño y formato.

Extensibilidad: posibilidad de extenderse o no.

Nota: por qué el elemento fue incluido, guía para su uso, etc.

Ejemplo: cuando sea apropiado, un ejemplo de cómo se usa el elemento.

Para asegurar la interoperabilidad entre sistemas de metadatos que aplican esta especificación se deben cumplir los requisitos para su conformidad (Thropp & Mckell, 2001a):

1. La instancia debe contener uno o más elementos LOM.
2. Todos los elementos LOM de la instancia de metadatos son utilizados para describir las características especificadas por LOM.
3. Los valores de los elementos LOM en la instancia estarán estructurados y definidos como lo especifica LOM, es decir, en las mismas categorías y subelementos.
4. Si la instancia contiene extensiones, éstas no deberán reemplazar a ningún elemento de la estructura LOM.

Una aplicación, como un ROA, tendrá conformidad con la especificación IMS Metadata si:

1. Es capaz de procesar todos los elementos LOM.
2. Cuando reciba una instancia de metadatos LOM, la almacena, la retransmite y conserva la instancia original durante su retransmisión.

En la Tabla 7 se da un ejemplo del código XML de un archivo de metadatos IMS. Para facilitar el entendimiento de los valores dados en el extremo superior izquierdo se muestra la imagen que se ha descrito (esto no es así en el archivo real). Se han insertado unas flechas del lado izquierdo para indicar el inicio `<lom>` y el fin `</lom>` del elemento raíz, después de éste vienen los elementos por categorías. El ejemplo muestra sólo las etiquetas

de la categoría general (también se indica con una flecha), pero de forma análoga se representan los elementos de las otras categorías. Si se lee detalladamente cada una de las líneas es relativamente sencillo para los humanos interpretar el código y dado que es un documento estructurado y en texto plano, para los ordenadores es simple procesarlo.


<pre><?xml version="1.0" encoding="UTF-8" ?> <!-- This is a Reload version 1.3 Metadata document --> <!-- Spawned from the Reload Metadata Generator - http://www.reload.ac.uk --></pre>	
<pre><lom xmlns="http://www.imsglobal.org/xsd/imsmd_v1p2" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://www.imsglobal.org/xsd/imsmd_v1p2 imsmd_v1p2p2.xsd"></pre>	<p>Inicio de etiqueta</p>
<pre> <general></pre>	<p>Inicio de etiqueta</p>
<pre> <identifier>01</identifier> <title> <langstring xml:lang="sp">Calabaza</langstring> </title> <catalogentry> <catalog>objetos_clara</catalog> <entry> <langstring xml:lang="es">clara01</langstring> </entry> </catalogentry> <language>es</language> <description> <langstring xml:lang="en">Imagen de calabaza con calabaza pequeña en platón de cerámica. Calabaza color naranja.</langstring> </description> <keyword> <langstringxml:lang="en">hornamento, vegetales</langstring> </keyword> <structure> <source> <langstring xml:lang="en">LOMv1.0</langstring> </source> <value> <langstring xml:lang="x-none">Atomic</langstring> </value> </structure> <aggregationlevel> <source> <langstring xml:lang="en">LOMv1.0</langstring> </source> <value> <langstring xml:lang="x-none">1</langstring> </value> </aggregationlevel></pre>	<p>Cierre de etiqueta</p>
<pre> </general></pre>	<p>Cierre de etiqueta</p>
<pre></lom></pre>	

Tabla 7. Código XML de un archivo de metadatos de IMS

La especificación también provee la información para hacer el mapeo de *Dublin Core* hacia LOM. Aunque este mapeo es posible, debe ponerse especial atención en que no implica una equivalencia estructural o semántica, esto es, que los elementos pueden coincidir pero los datos contenidos en cada metadato pueden tener un contexto semántico distinto. Esto se discutirá a fondo en el caso de estudio del siguiente capítulo.

6.2. IMS Content packaging

Como puede observarse en el Anexo B, ésta es una de las primeras especificaciones y la más intensamente revisada. Su importancia para los ROA está en que provee la función para formar paquetes de contenidos que el ROA podrá exportar para que los LMS y los LCMS puedan utilizar el OA. Asimismo, el ROA debe tener la capacidad de importar y extraer el contenido de los paquetes que reciba con esta especificación.

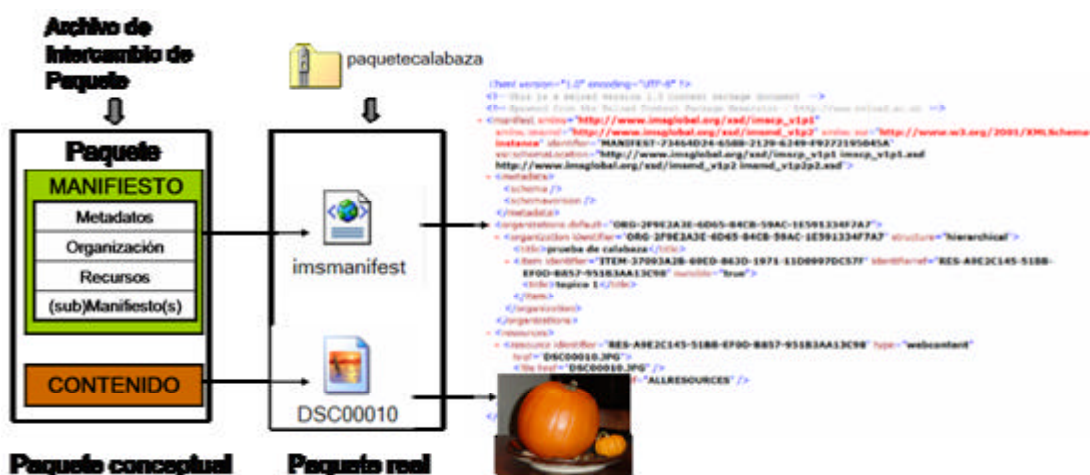


Figura 13. Representación de un paquete conceptual y un paquete real de IMS

Para IMS CP (Smythe & Jackl, 2004b), un paquete de contenido (*content package*) está compuesto por dos elementos (Figura 13): manifiesto y contenido. El manifiesto tiene secciones, que son elementos XML que describen al propio manifiesto con metadatos, la organización del contenido, las referencias a los recursos que componen el contenido incluyendo sus metadatos y las referencias a archivos externos, otro elemento pueden ser otros (sub)manifiestos. El contenido es propiamente el fichero del recurso (en SCORM se llama *asset*¹⁷). Con estos dos elementos se crea un archivo comprimido (usualmente *.zip*) y

¹⁷ Curiosamente el mismo nombre con el que se identifica a los elementos de software reutilizables

se llama Archivo de Intercambio de Paquete o PIF (*Package Interchange File*).

Un paquete representa una unidad de contenido utilizable y reutilizable que debe ser posible agregarlo o disgregarlo con otros paquetes, además debe contener toda la información necesaria para ser utilizado de forma autónoma, esto es, como puede deducirse, un objeto de aprendizaje. En SCORM los paquetes tienen una composición similar ya que están basados en esta especificación, pero como se mencionó anteriormente se les llaman SCO.

Los principales interesados en esta especificación son los desarrolladores de contenidos, los proveedores de LMS, los vendedores de sistemas informáticos (herramientas de autor, software de presentaciones, etcétera) y los proveedores de servicios de aprendizaje.

Para asegurar la interoperabilidad de paquetes de contenidos se deben cumplir los requisitos de conformidad (nivel 0) que la especificación detalla como (Smythe & Jackl, 2004a):

1. El paquete debe contener un archivo llamado `imsmanifest.xml` ubicado en el directorio raíz.
2. El paquete debe contener en el directorio raíz los archivos de control directamente referenciados.
3. El archivo `imsmanifest.xml` debe seguir los lineamientos descritos en la sección 3 del documento *XML Binding* de esta especificación.
4. Si el archivo `imsmanifest.xml` contiene IMS metadata, entonces debe contener la extensión para incluir los metadatos de la especificación v1.2.1.
5. El archivo `imsmanifest.xml` no debe referenciar elementos utilizando `XInclude`.
6. Todos los archivos en los que los recursos estén contenidos en el paquete deben identificarse por los elementos `<file>` en la sección `<resources>` del archivo `imsmanifest.xml` y debe estar incluido dentro del directorio o subdirectorios que contienen al `imsmanifest.xml`.

Cuando hay extensiones, es decir cuando en la implementación XML se agregan o se modifican elementos, para la conformidad (nivel 1) se requiere:

1. Cumplir con todos los requisitos del nivel 0, excepto el 5.
2. El `imsmanifest.xml` puede contener extensiones adicionales de espacios de nombres. Si las extensiones adicionales son descritas y controladas utilizando un

(como se explicó en la definición de los repositorios en el apartado 4.2.1).

esquema o un DTD modificado, entonces cualquier archivo de control referenciado debe ser incluido en el paquete.

La especificación observa que las extensiones pueden causar problemas cuando se requiere interoperabilidad con contenido de otros proveedores, ya que se requiere tener un acuerdo entre las dos partes b que complica la interoperabilidad global. También se presentan problemas de interoperabilidad cuando un desarrollador agrega extensiones o modifica el esquema de validación de documentos, ya que se requerirá de éste en particular para hacer la validación.

Para las herramientas y sistemas que importan, exportan, crean y manejan paquetes de contenido, los requisitos de conformidad (nivel 0) son:

1. Un sistema o herramienta con conformidad debe reconocer y procesar cualquier Paquete de Contenido IMS con conformidad nivel 0 ó 1.
2. Todos los elementos XML y los metadatos IMS o LOM presentes en el `imsmanifest.xml` deben mantenerse en su retransmisión.
3. Las extensiones de espacios de nombres diferentes a los de IMS Metadata o LOM y los del Enlace XML deben ser ignorados y no retransmitidos.

Para la conformidad nivel 1, es decir, cuando se conservan las extensiones:

1. Cumplir con los requisitos 1 y 2 del nivel 0.
2. Todos las extensiones de espacios de nombres deben ser conservados en su retransmisión.

La especificación se enfoca sólo a la descripción y la estructura de los contenidos y es importante mencionar que no se involucra con los modelos pedagógicos, dejando con esto la libertad de que el profesor utilice el que a su consideración, sea más conveniente en cada aplicación.

6.3. IMS Digital Repositories Interoperability

IMS DRI (Riley & Mckell, 2003b) tiene como objetivo facilitar el acceso a los contenidos en los repositorios para contextos educativos, con los LMS y los LCMS, pero también con otros sistemas como los portales de búsquedas. Esta especificación se propone para la interoperabilidad entre servicios o aplicaciones que tienen las funciones más comunes de un repositorio: buscar, exponer, coleccionar, enviar, almacenar, pedir, entregar y alertar. Entre

estas funciones, se reconocen cinco combinaciones como actividades principales: Buscar/Exponer, Colectar/Exponer, Enviar/Almacenar, Pedir/Entregar y Alertar/Exponer, que pueden verse con detalle en la Tabla 8.

Estas funciones se encuentran resaltadas en la Figura 14, en la que se muestra la arquitectura funcional entre un sistema *e-learning*, los repositorios digitales y los servicios de información. En la misma figura se muestran los roles, los componentes funcionales y los servicios que definen el espacio de interacción completo pero IMS DRI no incluye a todas éstas, sólo se enfoca a la interoperabilidad de las funciones que se han mencionado.

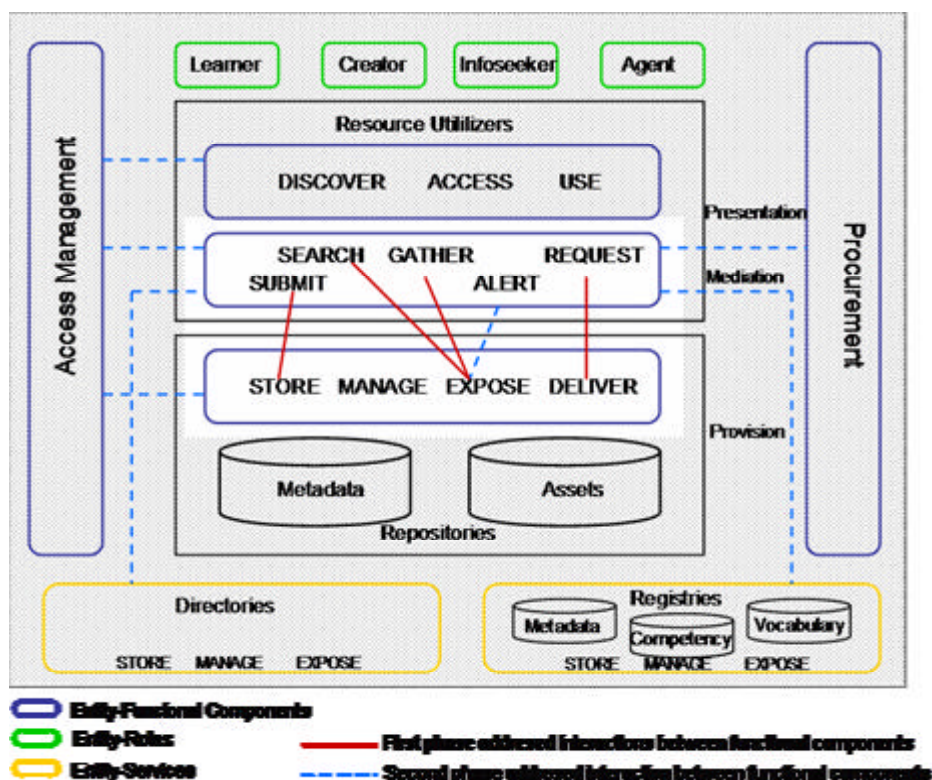


Figura 14. Arquitectura funcional. Reproducida de IMS DRI *Information Model* (Riley & Mckell, 2003b)

La especificación considera que los repositorios digitales tienen diversidad de formatos en sus contenidos, diferentes sistemas, tecnologías y objetivos, por lo que no tiene requisitos para la operación interna de los repositorios y sus recomendaciones. Para la implementación de las funciones principales, están orientadas a dos tipos de repositorios: los que ya tienen establecido un protocolo para la interoperabilidad, como Z39.50 (NISO, 2002) (comúnmente utilizado en las bibliotecas digitales), y aquellos repositorios que tienen la capacidad de implementar XQuery (Boag, Chamberlin, Fernández, Florescu,

Robie & Siméon, 2005) (para búsquedas de metadatos en XML) y como protocolo de acceso a SOAP¹⁸ (Gudgin, Hadley, Mendelsohn, Moreau & Nielsen, 2003), recomendados por la especificación para los ROA. En la Tabla 8 se resumen las recomendaciones tecnológicas que la especificación hace para cada una de las funciones principales.

<i>Función</i>	<i>Descripción</i>	<i>Recomendación tecnológica</i>
Buscar/Exponer (<i>Search/Expose</i>)	Ejecuta la búsqueda de metadatos asociados con los contenidos que el repositorio expone.	XQuery para colecciones con metadatos en XML y Z39.50 para búsquedas en sistemas bibliotecarios.
Colectar/Exponer (<i>Gather/Expose</i>)	Define la solicitud de metadatos que el repositorio expone, la agregación de los metadatos para utilizarse en búsquedas subsecuentes y la agregación de metadatos para crear nuevos repositorios.	No hay recomendación específica, pero IMS DRI sugiere que OAI puede proveer una funcionalidad adecuada.
Enviar/Almacenar (<i>Submit/Store</i>)	Se enfoca a la manera en la que un objeto se mueve a un repositorio desde un sitio accesible por red y cómo el objeto será representado en el repositorio para su acceso.	Se recomienda el uso de paquetes IMS a través de SOAP.
Pedir/Entregar (<i>Request/Deliver</i>)	Permite, que una vez que el usuario a localizado los metadatos en la función Buscar, pueda solicitar al repositorio el acceso al recurso.	Recomendación general para utilizar HTTP ¹⁹ y FTP ²⁰ para diferentes tipos de recursos. También IMS CP.
Alertar/Exponer (<i>Alert/Expose</i>)	Función clave, en la que a través de correo electrónico se notifica a los usuarios sobre eventos en el repositorio, pero no está contemplada todavía en esta versión de la especificación.	No hay recomendación específica ya que esta función sale del alcance de esta especificación.

Tabla 8. Descripción y recomendaciones tecnológicas para las funciones de un repositorio

¹⁸ SOAP (*Simple Object Access Protocol*) es un protocolo basado en XML para el intercambio de información electrónica estructurada entre pares en un entorno distribuido descentralizado.

¹⁹ HTTP. *HyperText Transfer Protocol*

²⁰ FTP. *File Transfer Protocol*

En cuanto a los requisitos de conformidad para la interoperabilidad en esta especificación (Riley & Mckell, 2003a) se apunta que no los hay, ya que las soluciones para el intercambio de información entre repositorios se pueden realizar de distintas maneras y no es posible determinar de forma estricta estos requisitos. En su lugar, se da un listado de las especificaciones y estándares que se recomienda seguir para elegir las aplicaciones y llevar a cabo una implementación que fomente la interoperabilidad (consideradas en para las recomendaciones de la Tabla 8). Para asegurar la interoperabilidad global la especificación requiere de más desarrollo.

Para la interoperabilidad de las funciones principales IMS DRI utiliza esquemas definidos en otras especificaciones como IMS LRM e IMS CP, así como de IMS RLI que se describe a continuación.

6.4. IMS Resource List Interoperability

IMS Resource List Interoperability (Jackl, 2004b) se refiere a cómo organizar, describir e intercambiar listas de recursos de un curso, como lo son las bibliografías. El nacimiento de esta especificación se justifica por la carencia de un método para importar y exportar los metadatos necesarios para agregar recursos dentro de listas de recursos o RL (*Resource List*), teniéndose hasta la fecha soluciones que son difíciles de compartir.

La especificación define como destinatarios de esta especificación a estudiantes, diseñadores instruccionales, bibliotecarios y repositorios de contenidos, entre otros, que tendrán la posibilidad de construir, seleccionar, utilizar y transmitir listas de recursos.

IMS RLI e IMS DRI tienen grandes similitudes, ambos se enfocan a manipular recursos y metadatos, tanto de colecciones de objetos como de información, así como de paquetes de objetos de aprendizaje.

La especificación se basa en un servicio abstracto de comportamiento y en un modelo de datos que describe en términos generales un recurso a nivel de un ítem, una colección de estos recursos (una lista), y los comportamientos asociados con un servicio de Administración de RL o RLM (*Resource List Manager*).

En la Figura 15 se muestra el modelo de la arquitectura del servicio de Administración de Listas de Recursos, que hace uso de un protocolo de mensajes para el intercambio de datos entre los sistemas administradores de listas. El alcance de la especificación se resalta en el

centro, puede observarse que se refiere a la interoperabilidad entre los datos y el modelo de comportamientos en el que se incluyen las operaciones que el RLM puede ejecutar. Estas operaciones están basadas en un modelo conocido como CRUD²¹ (*Create/Read/Update/Delete*) y se detallan en la Tabla 9, también pueden ubicarse en la Figura 15 como las operaciones centrales entre los servicios de IMS RLI.

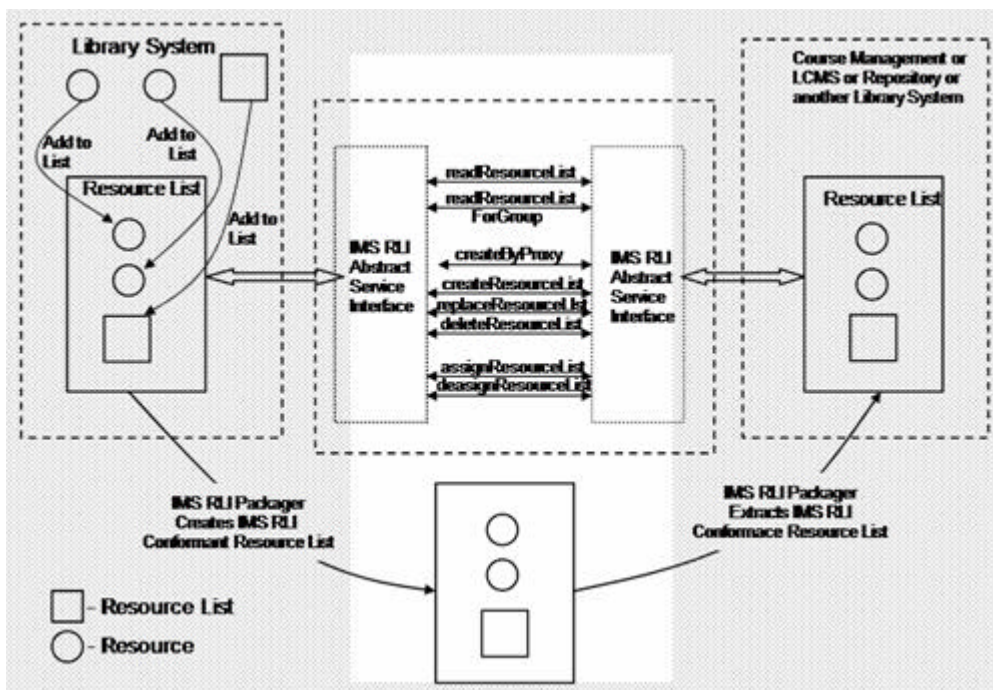


Figura 15. Arquitectura del Administrador de Servicios del RLI. Reproducción de *IMS Resource List Interoperability Information Model* (Jackl, 2004b)

El modelo de datos comprende un conjunto mínimo de elementos para citar publicaciones impresas. Se apoya en los estándares existentes para especificar e intercambiar dichos metadatos:

- Para metadatos utiliza LOM, a través de IMS LRM, combinado con ISO 690-2²² (ISO, 1997) para referencias bibliográficas a documentos electrónicos.

²¹ CRUD son operaciones primitivas conocidas en la comunidad de las bases de datos y de los sistemas de administración de información.

²² ISO 690-2 la segunda parte de la norma 690 emitida por la Organización Internacional de Normalización. Especifica los elementos que deben ser incluidos en una referencia bibliográfica de documentos electrónicos. Establece un orden prescrito para los elementos de la referencia e instauro convenciones para la transcripción y presentación de información derivada de la fuente del documento electrónico.

- Como esquemas de localización considera a OpenURL²³ (NISO, 2005), DOI²⁴ y PURL²⁵.
- Para el empaquetamiento de las listas y su transferencia entre sistemas se utiliza IMS CP.

<i>Operación</i>	<i>Descripción</i>
createResourceList	Pide la creación de una lista de recursos en el sistema destino, donde el sistema origen es responsable de alojar el identificador de la lista de recursos.
createByProxyResourceList	Solicita la creación de una lista de recursos en el sistema destino, donde el sistema origen es responsable de alojar el identificador de la lista de recursos.
readResourceList	Lee todo el contenido de la lista de recursos identificada. El destino debe regresar todo el contenido que tenga de la lista identificada.
readResourceListforGroup	Lee todo el contenido de un conjunto de listas de recursos asociadas en un grupo.
replaceResourceList	Escribe nuevo contenido en el registro de una lista. La información contenida anteriormente se destruye.
deleteResourceList	Solicita el borrado de una lista. La lista y cualquier relación con grupos de listas quedan eliminadas.
assignResourceList	Solicita asociar una lista a un grupo de listas.
deassignResourceList	Solicita eliminar la asociación de una lista en un grupo.

Tabla 9. Operaciones de un Administrador de Listas de Recursos

Cabe resaltar que la especificación no interviene en cómo se almacenan los recursos, sólo

²³ OpenURL es un protocolo para la interoperabilidad entre un componente de servicio y un recurso de información. Facilita al usuario la recuperación de la copia más apropiada del recurso que desea consultar.

²⁴ DOI (*Document Object Identifier*). Es un sistema para la identificación permanente de objetos de contenido en un ambiente digital, <http://www.doi.org/>.

²⁵ PURL (*Persistent Uniform Resource Locator*). Operan como un servicio intermediario de resolución de direcciones web. Este servicio asocia al PURL (dirección permanente) de un recurso con su dirección web actual (dirección que puede cambiar), ubicando al recurso con a través de un direccionamiento, <http://www.purl.org/>.

interviene en la interoperabilidad entre sistemas con paquetes de datos. Por otra parte, considerando que nunca se llegará a un sistema de descripción de recursos bibliográficos único, se propone el mapeo de LOM hacia los sistemas de citas más comunes entre bibliotecas y publicaciones.

Los requisitos de conformidad (Jackl, 2004a) para esta especificación son muchos y complejos, así que hay un documento específicos para ellos y se dividen en los necesarios para el Modelo de Información, éste a su vez en: datos, procesadores-receptores de datos y procesadores-transmisores de datos; y para el XML *Binding* en: generales, instancias, procesadores-receptores de datos, procesadores-creadores de datos y procesadores-transmisores de datos. No se detallará cada uno de ellos ya que definición es demasiado técnica y sale fuera del alcance de este trabajo.

6.5. Modelo interoperable basado en especificaciones IMS

Hasta ahora se ha explicado IMS y las especificaciones que están relacionadas con los ROA como son los metadatos, el empaquetamiento de contenidos y la transferencia de listas de recursos. Se han resaltado algunas de las limitantes de cada una de estas especificaciones y se han dado también, de forma general, los requisitos necesarios para lograr la interoperabilidad de cada una de ellas.

También se ha visto que las especificaciones están relacionadas entre sí y que se utilizan unas a otras para poder cumplir con sus funciones.

En IMS DRI se observó que no hay limitantes para la construcción de un repositorio y que sólo se limita a la interoperabilidad entre los mismos. Por ello, no es posible a partir de la especificación tener los elementos para construir un repositorio en sí, esto queda a elección de la organización que emprenda el proyecto.

Con estos antecedentes se propone modelo conceptual de cómo se relacionan las especificaciones para lograr la interoperabilidad entre los repositorios y los otros elementos que se mencionaron como componentes de un entorno integral *e-learning*, además de otros sistemas que pueden ser también interoperables y que aportan contenidos a los ROA.

El modelo se presenta en la Figura 16 y está dividido en cuatro partes: recursos, estándares IMS, protocolos y aplicaciones. Los recursos están almacenados y administrados en un ROA que puede ser centralizado o distribuido, se ha separado el contenedor de los

metadatos del de los recursos sólo para dejar más claros los componentes pero es irrelevante si éstos están o no en el mismo servidor. Todo lo que entra y sale del repositorio estará gestionado a través de IMS LRM o de IMS CP, es decir todo movimiento se hace a través de metadatos (que tienen referencia al objeto) o de paquetes de contenidos que en su interior incluyen los metadatos (e incluyen también el objeto). El gestor de toda la interoperabilidad es el IMS DRI, que aunque su función es la interoperabilidad entre repositorios se propone como intermediario para el intercambio de contenidos con los LMS, los LCMS y los agentes que son principalmente aplicaciones usuarios de los repositorios. Las funciones del IMS DRI pueden cubrir los requisitos que estas aplicaciones tienen de los ROA, propiciando así un entorno estandarizado. El IMS RLI funge como intermediario para poder crear las listas de recursos como objetos de aprendizaje y puedan ser incluidas en los repositorios y después ser entregadas a cualquier aplicación. Los protocolos se encargarán de la transferencia de contenidos entre el ROA y las distintas fuentes de recursos como son las bibliotecas, los servicios de información, bibliotecas digitales y otros repositorios. Siguiendo las recomendaciones de IMS DRI se propone utilizar Z39.50 para las bibliotecas que lo manejen y XQuery como otra opción para el resto de los casos y poder formar redes federadas o facilitar las búsquedas en repositorios distribuidos. OAI se utiliza para coleccionar metadatos de otros repositorios que lo permitan y poder agregarlos al repositorio principal. SOAP se propone para la entrega de paquetes y de contenidos a las peticiones de las aplicaciones.

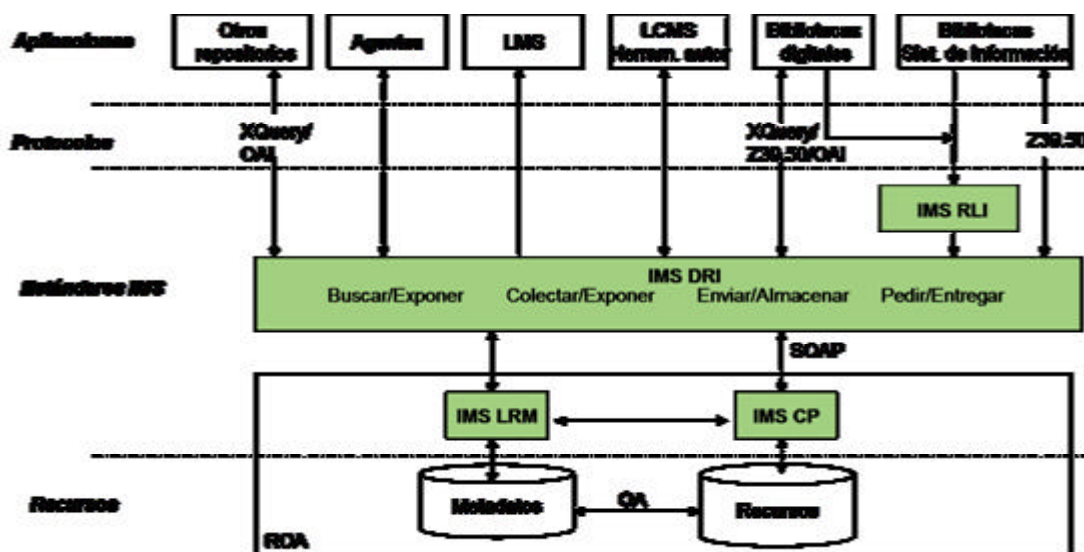


Figura 16. Modelo conceptual de la interoperabilidad de un ROA

Las flechas representan el flujo de contenidos, en general se plantea que todas las aplicaciones importan y exportan datos a los repositorios, excepto los LMS, que sólo hacen uso de ellos y no se contemplan como proveedores, al contrario de los LCMS que, como se ha mencionado, pueden crear e incluso modificar OA y devolverlos al mismo repositorio.

La intención de este modelo es ejemplificar de forma gráfica la interacción entre las especificaciones, las aplicaciones y los recursos, para mostrar que los ROA pueden ser una infraestructura que soporte todos los procesos en los que los contenidos están inmersos en un entorno integral *e-learning*.

Después de haber hecho la revisión a cada una de las especificaciones involucradas con el desarrollo de repositorios interoperables y de haber planteado este modelo, se puede concluir que es posible una implementación basada en las especificaciones de IMS para formar repositorios o redes de repositorios interoperables. En dicha implementación, además de la interoperabilidad entre repositorios, también es posible la comunicación con otras aplicaciones *e-learning*. Al seguirse un modelo basado en IMS se asegura que las especificaciones que se emitan o se actualicen seguirán siendo compatibles, lo cual dará durabilidad y robustez a todo el entorno.

Para continuar con un caso práctico sobre el uso de estándares, y presentar los beneficios y posibles debilidades, en el siguiente capítulo se aborda un caso de estudio para ver la reutilización de objetos y entender la importancia del uso de esquemas estándares de metadatos.
