



SIEDIC

METADATOS EN EL MUNDO

BIBLIOTECARIO:

**Teoría, práctica y aplicaciones
prácticas en el entorno digital
(preservación digital y linked data)**

Profesor: José Luis Bueren Gómez-Acebo

Módulo 1. Principales modelos de metadatos

ÍNDICE

| | |
|---|-----------|
| INTRODUCCIÓN | 4 |
| 1. PRINCIPALES MODELOS DE METADATOS | 6 |
| 1.1. Definición de “metadato” | 6 |
| 1.1.1. Tipología de metadatos..... | 9 |
| 1.2. El lenguaje XML | 11 |
| 1.2.1. XML..... | 15 |
| 1.3. El formato MARC. MARCXML | 19 |
| 1.3.1. MARCXML | 27 |
| 1.4. Dublin Core Metadata Initiative..... | 29 |
| 1.5. Europeana Semantic Elements | 43 |
| 1.6. Otros modelos de metadatos: METS, PREMIS, ONIX, MODS, MIX, y otros estándares de la Biblioteca del Congreso. | 55 |
| 1.6.1 Metadata Encoding and Transmission Standard (METS) | 55 |
| 1.6.2. Preservation Metadata: Implementation Strategies | 58 |
| 1.6.3. Online Information Exchange. ONIX | 60 |
| 1.6.4. Metadata Object Description Schema (MODS) | 62 |
| 1.6.5. Metadata for Images in XML Schema. MIX..... | 66 |
| 1.6.6. Otros estándares de la Biblioteca del Congreso..... | 74 |
| 2. EL PROTOCOLO OAI-PMH (OPEN ARCHIVE INITIATIVE: PROTOCOL FOR METADATA HARVESTING)..... | 76 |
| 3. CONCLUSIÓN: DE LOS METADATOS A LA WEB SEMÁNTICA | 78 |
| BIBLIOGRAFÍA | 80 |
| ÍNDICES..... | 84 |

Índice de ilustraciones 84

Índice de tablas 84

INTRODUCCIÓN

El objetivo de este curso es facilitaros un conocimiento relativamente profundo de la realidad de los metadatos en el mundo de las bibliotecas.

El curso, por su propia naturaleza es de carácter teórico pero los tres profesores hemos tenido nuestro acercamiento al tema no desde la teoría (aunque también hemos recibido varios cursos similares a éste, lógicamente) sino desde el empleo de varios modelos diferentes de metadatos en nuestro trabajo cotidiano por lo que tratamos de plantear aquello que es particularmente relevante a la hora de trabajar de hecho con modelos de metadatos.

En realidad, el concepto de metadatos esconde diferentes realidades y su definición es más compleja, y a la vez más sencilla, de lo que en muchas ocasiones se puede entender al leer bibliografía sobre ellos.

En esta curso vamos a ver qué son los metadatos, cuáles son los modelos de metadatos más relevantes en el ámbito de las bibliotecas digitales, para qué se están utilizando y qué ventajas e inconvenientes plantean.

El concepto de metadatos se aproxima en muchos aspectos a herramientas del mundo informático. En este curso mantendremos una aproximación “bibliotecaria” a los mismos, si bien, es imprescindible conocer algunas nociones más “informáticas”, especialmente en lo que se refiere al lenguaje XML.

Hoy en día, conocer los metadatos existentes y las diferentes iniciativas que se están desarrollando en torno a ellos es clave para la correcta gestión de una biblioteca (digital y no digital porque los metadatos afectan al trabajo de descripción bibliográfica). La normalización de los metadatos que empleemos en nuestra biblioteca será la manera de asegurar su interoperabilidad con otras bibliotecas e incluso la futura preservación de los objetos digitales, en el caso de las bibliotecas digitales.

Como veremos, los metadatos no representan, en esencia, nada nuevo para las bibliotecas, sin embargo, el desarrollo tecnológico y la gran cantidad de proyectos cooperativos que existen actualmente obligan al bibliotecario a estar al día sobre los cambios que se están produciendo para adaptar la práctica bibliotecaria al entorno digital.

El curso incluye tres módulos. El primero es de carácter más general y plantea la definición del concepto de “metadato” junto a un repaso de los principales modelos que se emplean en el ámbito bibliotecario. En este módulo se encuentra la introducción al lenguaje XML, que es conveniente conocer porque se ha convertido en un elemento fundamental en el tratamiento y conversión de metadatos de unos formatos a otros.

El segundo trata de la relación entre los metadatos y la Web Semántica o los datos enlazados. Es un tema muy interesante porque muchas de las iniciativas más innovadoras en bibliotecas se están centrando en la aplicación de este concepto de “datos enlazados”.

Por último, se presentan los metadatos orientados a la preservación digital a largo plazo, que es también un tema central para el desarrollo actual y futuro de las bibliotecas.

1. PRINCIPALES MODELOS DE METADATOS

1.1. Definición de “metadato”

Existen numerosas definiciones de metadatos. Vamos a ver dos definiciones y a partir de ahí comentaremos lo más significativo de ellas:

1. **Wikipedia**¹: (Del griego μετα, meta, 'después de, más allá de' y latín datum, 'lo que se da', «dato»), literalmente «sobre datos», son datos que describen otros datos. En general, un grupo de metadatos se refiere a un grupo de datos, llamado recurso. El concepto de metadatos es análogo al uso de índices para localizar objetos en vez de datos. Por ejemplo, en una biblioteca se usan fichas que especifican autores, títulos, casas editoriales y lugares para buscar libros. Así, los metadatos ayudan a ubicar datos.

Para varios campos de la informática, como la recuperación de información o la web semántica, los metadatos en etiquetas son un enfoque importante para construir un puente sobre el intervalo semántico.

2. **Dublin Core Metadata Initiative**²: En general, "datos sobre los datos"; funcionalmente "datos estructurados sobre los datos". Los metadatos abarcan tanto los datos asociados con un sistema de información, como los asociados a un objeto de información con la finalidad de descripción, administración, señalar los requisitos legales, la funcionalidad técnica, el uso y la utilización y la preservación.

Como vemos, la idea recurrente en ambas definiciones se refiere a la idea de que los metadatos son “datos sobre los datos”. De este modo, en realidad la clave es tener claro

¹ <http://es.wikipedia.org/wiki/Metadato>.

² http://www.sedic.es/glosario_DCMI.pdf

el concepto de “dato”. Dato, en este contexto es tanto el objeto que es descrito como el registro que lo describe. Dicho de otra manera, “dato” sería la obra Don Quijote de la Mancha, y “dato” sería también la anotación de que su autor es Miguel de Cervantes. En el caso de la novela, se trata de un “dato” sin más. En el caso de la indicación de la autoría se trata de un dato que informa sobre otro (sobre la obra Don Quijote), es por tanto, un dato sobre un dato; y por tanto un “metadato”.

Siguiendo la idea que exponen Eva Méndez y José A. Senso³:

“Los metadatos en sí no suponen algo completamente nuevo dentro del mundo bibliotecario (...) Atendiendo a la definición anterior, podríamos considerar la catalogación como un proceso de generación de metadatos. Teniendo en cuenta que la mayoría de los sistemas de metadatos han sido creados no sólo por profesionales de la información sino también por informáticos, diseñadores de programas, técnicos de sistemas, etc., la utilización de este término puede conllevar una carga documental a veces excesiva (con conceptos como por ejemplo, reglas de catalogación, clasificaciones de materias, etc.). "Metadato/s" se utiliza como un término neutral, que permite alejarnos de posibles prejuicios por parte de todas aquellas personas menos cercanas al mundo bibliotecario colocando, a todos los grupos profesionales implicados en su desarrollo, en una posición de igualdad, y a su propia evolución y aplicación, en una situación de multidisciplinariedad.”

Completando lo que dicen, no es que los metadatos no supongan algo completamente nuevo, es que, como ellos mismos explican, crear metadatos es lo que las bibliotecas han hecho desde siempre.

³ MÉNDEZ, Eva y SENSO, José A.: *Introducción a los metadatos: estándares y aplicación*. Disponible en: <http://www.sedic.es/autoformacion/metadatos/index.htm>.

Metadato es, por tanto, cualquier información que se refiera a otra información. Son ejemplos de metadatos: el resumen de un documento, los registros catalográficos en cualquier formato (ISBD, MARC, Dublin Core), una dirección IP, el índice de un documento, la extensión de un documento multimedia, los datos que nos ofrece el ordenador sobre cualquier documento (fecha y usuario de creación/modificación, peso del fichero), etc.

Un buen resumen de lo dicho es lo que recoge la Guía de uso de Dublin Core de la SEDIC⁴:

“Los metadatos han estado con nosotros desde que el primer bibliotecario hizo una lista de los documentos que tenía en un estante, en un pergamino escrito a mano. El término "meta" viene de una palabra griega que significa "junto a, con, después, siguiente". El uso más reciente del latín y el inglés emplearía "meta" para indicar algo trascendental, o sobrenatural. Los metadatos, pues, pueden ser definidos como datos sobre otros datos. Es el término usado en la era de Internet para la información que los bibliotecarios tradicionalmente habían puesto en los catálogos, y más comúnmente se refiere a información descriptiva sobre recursos de la Web.

Un registro de metadatos consiste en un conjunto de atributos, o elementos, necesario para describir la fuente en cuestión. Por ejemplo, un sistema de metadatos común entre los bibliotecarios --el catálogo de biblioteca -- contiene un conjunto de registros de metadatos con elementos que describen un libro u otra publicación en una biblioteca: autor, título, fecha de creación o publicación, materia, signatura topográfica...

La relación entre un registro de metadatos y el recurso al que describe puede darse de una de estas dos formas:

⁴ http://www.sedic.es/usando_dublin_core.pdf

1. los elementos pueden estar en un registro separado del documento, como en el caso del registro de un catálogo de bibliotecas; o
2. los metadatos pueden estar incluidos [embebido] en el propio recurso.”

1.1.1. Tipología de metadatos

Dado que su definición es tan amplia, la tipología de metadatos que nos podemos encontrar es casi infinita, en función de los objetos que se describan. La Wikipedia en inglés⁵, por ejemplo, menciona varios: estructurales, de guía, técnicos, de proceso, descriptivos, administrativos (y dentro de estos incluyen los de preservación). Sin embargo estas tipologías no dejan de ser relativamente arbitrarias. También podríamos diferenciar los metadatos en función de los objetos a los que se aplican: objetos físicos o digitales, imágenes, texto, audio o vídeo, etc.

En bibliotecas, los más habituales hasta ahora han sido los llamados **metadatos descriptivos**⁶, es decir, aquellos que aportan información que ayuda a buscar y encontrar un objeto con datos como el título, el autor, la materia, palabras clave, editor, etc. Estos serían los datos que tradicionalmente se consignan en los catálogos de bibliotecas (tanto manuales como automatizados).

Con la aparición de los objetos digitales se hace casi inevitable la utilización de otros tipos de metadatos como los **administrativos** que recogen información sobre los derechos que se aplican a un objeto o su forma de acceso y de reproducción. También es necesario la inclusión de datos (metadatos) **técnicos** donde se especifique el formato

⁵ http://en.wikipedia.org/wiki/Metadata#Metadata_types.

⁶ Nótese que en realidad es una redundancia, un metadato por su propia definición siempre describe algo

y la versión de un objeto digital, información sobre el hardware y software necesario para su reproducción (o empleados en su creación).

Otro tipo de metadatos que se suelen mencionar son los relativos a la **estructura** de un objeto digital. Es decir, cuántas partes lo componen y cómo está estructurado (por ejemplo, indicar que una obra está formada por tres volúmenes y cuál de estos es el primero, cuál el segundo y cuál el tercero).

Igualmente hay un conjunto de metadatos clave que son los de **preservación** donde se recoge toda la información necesaria para asegurar el acceso a largo plazo a los objetos digitales.

En cualquier caso, como se verá cuando entremos en detalle en los diferentes modelos de metadatos, estas distinciones son muchas veces arbitrarias y se solapan, dado que para preservar un documento es necesario conocer sus características técnicas y estructurales y su descripción. E igualmente que describir la estructura sólo tiene sentido si hemos descrito e identificado el objeto digital, etc.

Por este motivo, al hablar de metadatos es más útil pensar en “modelos de metadatos”. Estos modelos son conjuntos de datos (de metadatos) que pretenden describir objetos desde una perspectiva concreta y con una finalidad determinada. Así, por ejemplo, el formato MARC 21 (que a pesar de su origen en los años 60 puede ser considerado un modelo de metadatos) pretende la descripción catalográfica de los objetos. El modelo METS, veremos más adelante que se centra en la estructura, PREMIS, en la preservación a largo plazo, RDF en la conceptualización para el desarrollo de la web semántica, etc.

Más adelante veremos varios de los modelos de metadatos más relevantes para las bibliotecas. Antes, sin embargo, es necesario detenerse a explicar algunas nociones del lenguaje XML, porque es básico para entender cómo funcionan y cómo se estructuran estos diferentes modelos. En el próximo apartado vamos a explicar qué es XML. Al principio puede parecer algo complejo pero en realidad XML es más fácil de entender

que de explicar. Os aconsejo leer hasta el final y ver los ejemplos para tener una idea. Y tened en cuenta que para esta lección nos basta con entender los principios de XML, para así comprender mejor los diferentes modelos de metadatos.

1.2. El lenguaje XML

XML, eXtensible Markup Language se puede definir como un lenguaje de marcas textual pensado para representar información estructurada: documentos, datos, configuraciones, libros, transacciones, facturas y mucho más. Se deriva de un formato más antiguo llamado SGML (ISO 8879) para hacerlo más apto al entorno Web⁷.

Un lenguaje de marcas, por otra parte, es una forma de codificar un documento que, junto con el texto, incorpora etiquetas o marcas que contienen información adicional acerca de la estructura del texto o su presentación⁸. Un ejemplo de cómo funciona el lenguaje de marcado puede observarse en el dictado de viva voz de un documento a una persona que lo transcribe a máquina:

Ponga estilo de carta, ponga comillas, ponga mayúsculas, Estimado Juan, ponga dos puntos, aparte, sangría, ponga primera letra mayúscula, te escribo esta carta, ponga negrillas, de forma muy urgente, cierre negrilla, ya que no me has enviado... etc".

El lenguaje de marcas más conocido es HTML, que se utiliza fundamentalmente en la Web. Normalmente se dice que HTML es un lenguaje “de presentación” y XML “de estructura”, pero ambos son lenguajes de marcado.

⁷ <http://www.w3.org/standards/xml/core>.

⁸ http://es.wikipedia.org/wiki/Lenguaje_de_marcado.

Los lenguajes de marcado se llaman así por las etiquetas que incorporan en los documentos. Estas etiquetas se reconocen porque van encerradas entre los símbolos < >.

Así por ejemplo, en HTML, si encontramos “Te escribo esta carta, <bold>de forma muy urgente</bold>, ya que no me has enviado...” la instrucción <bold> hará que el navegador que lee el código HTML ponga lo que encuentra a continuación en negrita hasta que encuentre una etiqueta de cierre que indique que se acaba la negrita, en este caso, la etiqueta </bold>⁹.

El encargado de determinar qué etiquetas tiene y cómo deben interpretarse los navegadores para el caso del lenguaje HTML es el W3C, el World Wide Web Consortium¹⁰. Este Consorcio es una comunidad internacional donde las organizaciones participantes, personal a tiempo completo y el público en general trabajan conjuntamente para desarrollar estándares Web. Liderado por el inventor de la Web Tim Berners-Lee y el Director Ejecutivo (CEO) Jeffrey Jaffe, la misión del W3C es guiar la Web hacia su máximo potencial. Por este motivo decimos que XML es un estándar.

Una de sus características más importantes es la **simplicidad**. Un documento XML es un documento de texto plano (como los ficheros con extensión .TXT que podemos crear usando el bloc de notas de Windows), que intercala unas “etiquetas” o “marcas” especiales entre el texto, para diferenciar así su contenido informativo y los datos que contiene.

XML proviene de un lenguaje inventado por IBM en los años setenta, llamado GML (Generalized Markup Language), que surgió por la necesidad que tenía la empresa de almacenar grandes cantidades de información. Este lenguaje gustó a la ISO, por lo que

⁹ Más adelante veremos documentos XML con lo que estas definiciones teóricas se comprenderán mucho mejor

¹⁰ <http://www.w3c.es/>.

en 1986 trabajaron para normalizarlo, creando SGML (Standard Generalized Markup Language), capaz de adaptarse a un gran abanico de problemas. A partir de él se han creado otros sistemas para almacenar información.

Dentro de SGML vamos a destacar dos elementos clave que más tarde heredará XML:

El primero de ellos es el concepto de **tipo de documento**, que como veremos también está presente en XML. Un tipo de documento puede interpretarse como una abstracción de documentos que comparten un objetivo, propiedades y características comunes, y que van a ser tratados de forma similar¹¹.

La DTD es una definición que especifica restricciones en la estructura y sintaxis que se aplican a un documento o conjunto de documentos. La DTD se puede incluir dentro del archivo del documento, pero normalmente se almacena en un fichero ASCII de texto separado. La sintaxis de las DTD para SGML y XML es similar pero no idéntica.

Las DTD se emplean generalmente para determinar la estructura de un documento mediante etiquetas (en inglés tags) XML o SGML. Una DTD describe:

Elementos: indican qué etiquetas son permitidas y el contenido de dichas etiquetas.

Estructura: indica el orden en que van las etiquetas en el documento. Si los elementos son opcionales u obligatorios o si se pueden repetir o no.

Anidamiento: indica qué etiquetas van dentro de otras.

¹¹ EÍTO BRUN, Ricardo (2004). *Curso de formación Online. XML: aplicaciones libres para la edición electrónica, bibliotecas, archivos y centros de documentación. Módulo 1: Introducción y Características del lenguaje XML*. SEDIC.

De este modo, todos los documentos de un mismo tipo tendrán una DTD común en la cual se basan. Por ejemplo, una DTD para describir libro nos dirá que hay un elemento libro que contiene otros elementos como autor, título, fecha, etc. Luego el documento SGML será el que dé contenido a esos elementos definidos en la DTD.

El otro elemento de SGML fundamental para comprender XML son las **hojas de estilo**. Una hoja de estilo contiene instrucciones para convertir las marcas del documento SGML en algo diferente (o bien de presentación o bien de contenido). Es decir, que nos permite convertir un documento que tiene los elementos nombrados de una determinada forma a otro documento con los elementos nombrados de otra manera. Así, por ejemplo, será posible convertir un documento donde el título se reconoce por la etiqueta <title> a otro donde se reconozca por la etiqueta <title>.

De este modo, los conceptos clave de SGML que heredará XML son:

- Es un lenguaje de marcas.
- Es simple (no requiere software especial, basta un procesador de texto).
- Su carácter no propietario. Se trata de un estándar, luego facilita la interoperabilidad y la normalización.
- Utiliza DTDs para indicar qué elementos tiene un documento y cómo se relacionan.
- Se pueden emplear hojas de estilo para modificar los documentos.

1.2.1. XML

El lenguaje XML (eXtensible Markup Language) comenzó a desarrollarse en septiembre de 1996 auspiciado por el W3C con el objetivo de hacer más sencillo SGML y adaptarlo al entorno de la Web.

Al igual que SGML, XML es un lenguaje de marcas. Por ejemplo, una sección de un documento XML podría ser esta:

```
<nombre>
    Juan Pérez
</nombre>
```

Como vemos se declara un elemento, en este caso, <nombre>, se da su contenido, “Juan Pérez” y se cierra el elemento.

XML se caracteriza además de por ser un lenguaje de marcas, por estar orientado a expresar la estructura (más que la presentación) de un documento. Es decir, las marcas de XML en principio no nos van a decir si un determinado valor (“Juan Pérez” en el caso anterior) se escribe en negrita, en cursiva, justificado, en mayúsculas, etc.

XML permite expresar la estructura porque es posible ir añadiendo marcas según el nivel de estructura que queramos expresar:

```
<usuario>
    <nombre> Juan </nombre>
    <apellido> Pérez </apellido>
    <documento> 242452345 </documento>
</usuario>
```

A cada sección delimitada por etiquetas se le llama elemento. En este caso, tenemos cuatro elementos: usuario, nombre, apellido, y documento. Y la estructura del documento

nos permite ver que existe entre ellos una jerarquía: usuario es un elemento que contiene a los otros tres.

XML no determina qué elementos deben componer un documento. Ni siquiera determina “a priori” las relaciones entre los elementos, es decir su jerarquía. Precisamente, la ventaja de xml es que permite a cada usuario definir qué elementos compondrán un documento y cómo se relacionan estos elementos.

Al igual que ocurría con SGML, si tomamos por ejemplo un libro, xml nos permite decir que el elemento libro contendrá los elementos: autor, título, pie de imprenta, notas...” tantos como queramos. Y a su vez, podemos decir que el elemento autor contendrá los elementos: nombre, apellidos, fecha de nacimiento, etc.

Además, los elementos pueden tener atributos que matizan su significado u ofrecen información adicional necesaria para procesar el documento. Los atributos tienen un nombre y contienen un valor con un tipo de dato especial:

```
<autor fechaNacimiento = “1920”> Juan Pérez </autor>
```

Los atributos tienen: Un nombre o identificador; el tipo de dato que pueden recoger, o un valor de un conjunto cerrado de datos; carácter opcional, obligatorio o fijo.

Como vemos, XML es muy flexible porque nos permite definir cualquier conjunto de elementos y establecer la jerarquía que se quiera entre ellos. Pero, ¿cómo se hace esto? ¿Dónde está la estructura de nuestro documento? Al igual que SGML, XML permite utilizar una Definición de Tipo de Documento (DTD) para expresar la estructura del documento.

Aparte de otras limitaciones, un problema que plantean las DTDs es que en tanto documentos, no son documentos XML sino que tienen una sintaxis propia. Es decir, para escribir una DTD hay que conocer dos lenguajes: XML y el de las DTDs.

Para solucionar estos y otros problemas, XML cuenta con los llamados esquemas. Un esquema XML sí es un documento XML.

En este sentido, los esquemas XML equivalen a las DTD, son formas diferentes de representar la estructura de un tipo de documento y las reglas que se le aplican.

Los esquemas, además permiten una mayor precisión que éstas. Con las DTD, por ejemplo, no es posible indicar si un elemento o atributo debe recoger un valor de tipo "entero", "numérico", "fecha", o "una cadena de caracteres con un máximo de 20 caracteres". Los esquemas XML sí permiten especificar estas características con un gran nivel de detalle (similar al que podemos encontrar en un sistema de bases de datos).

Por otra parte, al igual que en las DTD, en un esquema XML se indicará:

- Los elementos que pueden aparecer en un documento.
- Los elementos que son repetibles, opcionales u obligatorios.
- El orden de estos elementos, y cómo pueden anidarse o contenerse unos a otros.
- Los atributos que se pueden asociar a cada elemento, y el tipo de dato que deben recoger.
- Los atributos que son opcionales u obligatorios
- Valores por defecto para los atributos

En la Wikipedia encontramos un ejemplo muy sencillo de qué es un esquema XML. Primero una muestra de cómo sería un esquema:

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <xsd:element name="Libro">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="Título" type="xsd:string"/>
        <xsd:element name="Autores" type="xsd:string" maxOccurs="10"/>
        <xsd:element name="Editorial" type="xsd:string"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
</xsd:schema>
```

```
</xsd:sequence>
  <xsd:attribute name="precio" type="xsd:double"/>
</xsd:complexType>
</xsd:element>
</xsd:schema>
```

Y a continuación explica lo que estamos viendo.

“Podemos ver como en ambos casos se inician las declaraciones indicando la versión de XML que se va a utilizar y la codificación que se usa. Estos dos campos son necesarios para poder interpretar el esquema. Además, en la siguiente línea de código podemos ver como se redirecciona al usuario a la página que ofrece las pautas de creación de XML Schema en las que se basará la descripción del esquema.

El elemento raíz se llama “Libro” y tiene tres hijos (elementos anidados) y un atributo. Los hijos son “Titulo”, “Editorial” que deben aparecer una vez y “Autores” que puede aparecer de una a diez veces. El hecho de que estén agrupados en una secuencia indica que los elementos deben aparecer en orden, es decir, primero el “Titulo”, luego los “Autores” y por último la “Editorial”. Los tres elementos son de tipo string. El atributo de libro se llama “precio” y es de tipo double.”

En cualquier documento XML que veamos encontraremos siempre una referencia a la DTD o al esquema según el cual se ha elaborado ese documento XML. Será la única forma de que los programas que leen el documento XML puedan comprobar que está “bien hecho”.

En este sentido, falta por comentar una distinción importante a la hora de leer un documento XML. Se trata de diferencia entre un **documento válido** y uno **bien formado**: un documento es válido cuando cumple el esquema o la DTD que viene en su definición. Un documento está bien formado cuando cumple la sintaxis XML. Es decir, que nos podemos encontrar errores de sintaxis (por ejemplo, que se deje sin cerrar una

etiqueta) o de “estructura” (por ejemplo, que incluya elementos que no están declarados en la hoja de estilo).

De este modo, como hemos dicho, en un documento XML encontraremos siempre una referencia a su esquema o a su DTD. Es posible, sin embargo, que también encontremos referencias a otros documentos. Un caso frecuente y muy importante para XML (como ya comentamos al hablar de SGML) es el de las **hojas de estilo**.

Las hojas de estilo son documentos que incluyen instrucciones para convertir un documento XML en un documento de otro tipo (puede ser otro documento XML o no). Por ejemplo, es frecuente utilizar una hoja de estilo para convertir un documento XML a HTML.

Existen dos tipos de hojas de estilo: css y xsl. En XML es más frecuente utilizar las hojas xsl en una versión llamada xslt.

Hasta ahora hemos visto la parte más ardua de este tema: la definición de metadatos y el lenguaje XML. A partir de ahora vamos a ir repasando diferentes modelos de metadatos y vamos a ver para qué se utilizan. Con esto, seguro que lo explicado hasta ahora va cobrando más sentido.

1.3. El formato MARC. MARCXML

Generalmente al hablar de metadatos no se hace alusión al formato MARC. Sin embargo, para las bibliotecas, el formato MARC ha sido y es el principal modelo para dar cuerpo a la descripción bibliográfica. Las bibliotecas digitales en muchos casos ofrecen y utilizan otros modelos de metadatos que también veremos, pero en el caso de las principales bibliotecas, el modelo que subyace en su base de datos son los registros MARC procedentes de los catálogos automatizados.

Aunque seguramente todos estéis familiarizados con el formato MARC, vamos a repasar lo principal:

En primer lugar hay que tener en cuenta que los distintos tipos de formatos Marc son versiones diferentes de la norma ISO 2709 para el intercambio de registros.

El formato Marc original data de 1966 como un proyecto de la Biblioteca del Congreso al que se llamó Marc 1. Fue seguido en 1967 de una versión desarrollada como un proyecto conjunto entre la Biblioteca del Congreso y la Británica. El formato Marc estaba pensado para el intercambio internacional de registros, pero desafortunadamente salieron dos versiones, LC MARC en 1967 y UK MARC en 1968. En los siguientes 15-20 años surgieron otros formatos nacionales basados en uno u otro.

El formato fue desarrollado en una época en que las cintas magnéticas eran el principal método para transmitir grandes cantidades de datos entre ordenadores. Cuando un ordenador lee una cinta tiene que empezar por el principio y seguir secuencialmente. Y no puede ir para atrás. Así, un registro típico en cinta tendría que tener ciertos tipos de datos al principio indicando al ordenador qué tipo de documento se describía en registro (libro, publicación periódica, audiovisual, etc.) de modo que el ordenador pudiera almacenar cada registro en la base de datos adecuada. En el LC MARC, por ejemplo, diferentes tipos de registros tenían que ser almacenados en diferentes bases de datos dado que las definiciones de las etiquetas variaban entre diferentes tipos de documentos (libro, publicación periódica...) Con los discos, los ordenadores pueden ir a cualquier parte del registro volver atrás y adelante.

En los registros MARC se pueden ver al comienzo muchos números que representan el tamaño de cada registro para que un ordenador antiguo pudiera copiar los datos de la cinta en el sitio adecuado¹².

¹² HOPKINSON, Alan: "MARC developments in UNITED KINGDOM". Middlesex University, 2003. Disponible en: <http://www.unilib.bg.ac.rs/en/e-sources/infotheca/1-2003/hopkinson2.php>.

La aparición de variantes nacionales del formato MARC hizo necesaria la aparición del conocido como UNIMARC que permitiera el intercambio de registros a partir de formatos MARC diferentes. La IFLA, dentro del Programa Control Bibliográfico Universal publicó en 1967 la primera edición de UNIMARC para alcanzar este objetivo.

Sin embargo, a partir de los noventa, el formato UNIMARC deja de verse como una solución y se decidió que todo el mundo adoptara el formato más utilizado en el mundo que era el USMARC, que cambió de nombre por el actual MARC 21. Para la gestión del formato, para actualizarlo y adaptarlo a las necesidades de la comunidad de usuarios, existe la Agencia de mantenimiento del MARC 21, formada por la Biblioteca del Congreso de Estados Unidos, la Biblioteca y Archivo de Canadá y la Biblioteca Británica. Para ello existe un Forum MARC (MARC@LOC.GOV) que permite la discusión de cambios y otros asuntos.

Antes de hablar de la estructura de los registros MARC y de los tipos de registros que hay, conviene aclarar **qué es el formato MARC y qué no**.

El formato MARC, en tanto que norma ISO 2709 no es un sistema de catalogación. No incluye, en principio, normas sobre cómo deben describirse los recursos. Por ejemplo, no indica si los datos para una catalogación se toman de la portada o de la cubierta. Tampoco está asociado a un sistema de visualización concreto, ni determina una puntuación.

El formato MARC es un sistema muy estructurado para que la información bibliográfica pueda ser intercambiada por ordenadores. El grado de estructuración del formato sí que viene, en cambio, determinado por las normas catalográficas a las que se quiere que dé cabida. Es decir, la profundidad de la estructura, el número de campos, de subcampos, y de indicadores, depende de qué criterios de catalogación sigamos.

Por ejemplo, en el formato MARC 21 al campo de Mención de Título (245), se le han asignado 12 subcampos para expresar todas las posibilidades que determinan las Reglas de Catalogación. Así, el campo permite indicar:

- \$a - Título (NR)
- \$b - Parte restante del título (NR)
- \$c - Mención de responsabilidad, etc. (NR)
- \$f - Fechas de inclusión (NR) Período durante el cual se creó el contenido completo de los materiales descritos.
- \$g - Fechas de la mayor parte (NR) Período durante el cual se creó la mayor parte del contenido de los materiales descritos.
- \$h - Medio físico (NR)
- \$k - Forma (R) Término descriptivo de la forma de los materiales descritos, determinada por un examen de sus características físicas, el tema de su contenido intelectual o el tipo de información que contienen.
- \$n - Número de la parte/sección de la obra (R)
- \$p - Nombre de la parte/sección de la obra (R)
- \$s - Versión (NR) Nombre, código o descripción de un ejemplar de los materiales descritos que fue generado en momentos diferentes o para audiencias diferentes.
- \$6 - Enlace (NR) Consulte los Subcampos de Control
- \$8 - Vínculo de campo y número de secuencia (R) Consulte los Subcampos de Control

Nada impide tener otro formato MARC en el que haya más o menos subcampos o los mismos subcampos pero con otro valor. De hecho, el formato MARC, por decirlo de algún modo, no les da valor. Son las realizaciones de la norma, el formato concreto, en este caso el MARC 21, el que dice que el subcampo b es para indicar el resto del título.

Igualmente, el formato no incluye opciones de visualización de los registros, sino que son los programas informáticos concretos los que determinan en cada caso el orden de presentación de los campos de un registro. La gran estructuración del formato permite tener muchas posibilidades de presentación de los registros. Si los registros se catalogaran sin la estructura del formato, los programas “no sabrían” distinguir los diferentes campos de un registro y no podrían dar una presentación diferente de la que se dio al introducir el registro en el fichero.

Esta misma estructuración del formato lo hace apto para utilizarlo como sistema de almacenamiento en las bases de datos, dado que permite búsquedas muy precisas (tan precisas, en este caso, como sea su estructura y el nivel de la catalogación empleado).

Vamos a ver ya cómo es esa estructura de los registros MARC:

Especificada en la norma ISO 2709 (ANSI Z39.2) que establece que las partes que componen un registro son:

- **Líder o cabecera:** son las primeras 24 posiciones (00-23) de cada registro y suministra información para el procesamiento del registro.
- **Directorio:** es un índice generado e interpretado por la computadora, sigue inmediatamente a la cabecera en la posición 24 y remite a la localización de los campos de datos de un registro. Es una serie de entrada de posiciones de 12 caracteres que indican la etiqueta, longitud y la posición de comienzo del primer carácter de cada campo variable.
- **Campos variables:** hay dos tipos de campos variables:
 - **Campos variables de control:** campos etiquetados 00X y no contienen indicadores ni códigos de subcampos.
 - **Campos variables de datos:** campos etiquetados 01X-8XX y pueden tener definidos los indicadores o no, todos tienen el subcampo \$a.

Designación del contenido

- **Etiquetas:** números de tres dígitos usados para identificar los campos que componen un registro y al tipo de dato que le sigue, por ejemplo el campo título se identifica con la etiqueta 245.

- **Indicadores:** dos caracteres que se usan al comienzo de los campos variable, contienen valores que interpretan la información sobre el tipo o función de los datos en el campo; identificados independientemente.
- **Códigos de subcampos:** entendemos como subcampos a cada elemento que compone un campo. Los códigos de subcampo son de dos caracteres que preceden a cada elemento para identificarlo y señalar su localización dentro de un registro. Un código de subcampo está compuesto por: un delimitador, representado por el símbolo \$ y de un identificador o carácter alfanumérico o numérico.

Solemos hablar del formato MARC pero en realidad hay que señalar que, del mismo modo, que existen variedades nacionales del formato, existen diferentes formatos MARC para distintos tipos de registros:

- **Bibliográficos:** Se emplea para la identificación y descripción de distintos tipos de materiales bibliográficos. Es el más conocido de todos los tipos de formato, y el único que se suele traducir y adaptar en todos los países
- **Autoridad:** Se emplea en registros que ofrecen información sobre las formas de autoridad (aceptadas o no) que pueden emplearse como puntos de acceso. Permite ofrecer notas aclaratorias sobre cómo y cuándo usar la autoridad, nos remite a otras formas de autoridad aceptadas.
- **Fondos:** El fin de este tipo de formato es proporcionar información sobre cada ítem (unidad circulante), identificar cada uno de los documentos que constituyen el fondo, con datos como la signatura, colección, número de copia, etc. Identifica tres tipos de ítems: simple (por ejemplo, monografía en un volumen), multiparte (por ejemplo, obra en varios volúmenes), seriado (por ejemplo, publicación periódica o seriada).
- **Clasificación:** Este formato permite codificar los datos relacionados con los números de clasificación y las notas asociadas con ellos. Permite, por tanto,

relacionar las notaciones con sus encabezamientos de materia o descriptores correspondientes.

- **Comunidad:** Describe recursos no bibliográficos que completan las necesidades de información de una comunidad.

Otro aspecto que conviene mencionar al hilo de algo que ya se ha dicho es que, como el formato no establece normas ni tiene, en principio, un conjunto determinado de campos, subcampos o indicadores, son las realizaciones concretas, como el MARC 21, los que establecen cuáles, cuántos y qué significan estos campos. La divergencia respecto a esto es lo que dio lugar en su día a las diferencias nacionales en los formatos.

El formato MARC, hemos dicho, surge como formato de intercambio de registros bibliográficos. Y no es, también hemos dicho, un código de catalogación. Sin embargo, sus enormes posibilidades de estructuración han hecho de él un producto empleado muy frecuentemente como interfaz de catalogación. Dado que la información que codificamos según el formato MARC puede presentarse de la forma que se considere más conveniente, no hay ningún inconveniente en utilizar una plantilla de campos, subcampos e indicadores para llevar a cabo la catalogación.

Así, el formato, originalmente de intercambio se ha mostrado capaz de expresar, por una parte, las complejidades de las Reglas de Catalogación y, por otra, los detalles que se consideren necesarios para la visualización.

Típicamente el formato ha recibido algunas críticas. Una de ellas es que el formato es muy complejo (crítica que comparte, por cierto, con las Reglas de Catalogación). El formato MARC no es en realidad ni más ni menos complejo que los registros que debe describir. Y, por otra parte, que el formato contemple la posibilidad de una descripción muy detallada y muy estructurada, no implica que sea obligatorio llevarla a cabo, sino que cada catalogador o centro catalogador puede decidir el nivel de detalle que quiere en cada caso.

También es frecuente que se diga que no es flexible, sin embargo, el formato MARC, en este caso, los formatos concretos, han demostrado que sí que se adaptan. El MARC 21 está permanentemente sujeto a revisión y a ampliación, de modo que pueda incluir cambios en criterios catalográficos o simplemente tecnológicos. También ha demostrado un grado de compatibilidad con otros formatos o sistemas de descripción más modernos basados en el lenguaje XML. De hecho la compatibilidad del MARC con XML es completa.

Ejemplo de registro en formato MARC:

```
Un mundo feliz [Texto impreso] Cambiar visualización
Huxley, Aldous 1894-1963
000:      : am b0n a
001:      : bimo0001846064
005:      : 20110509
008:      : 030416s2003 sp | ||| ||spa
016: 7    : bimoBNE20030284420|2SpMaBN
017:      : B 8785-2003|bOficina Depósito Legal Barcelona
020:      : 84-9759-425-8
040:      : SpMaBN|bspa|cSpMaBN|erdc
041: 1    : spa|heng
080: 0    : 821.111-31"19"
100: 10   : Huxley, Aldous|d1894-1963
240: 10   : Brave new world|lEspañol
245: 13   : Un mundo feliz|cAldous Huxley ; traducción de Ramón Hernández|h[Texto impreso]
250:      : 1ª ed. en este formato
260:      : Barcelona|bDebolsillo|c2003
300:      : 254 p.|c19 cm
490: 1    : Contemporánea|v185
594:      : Traducción de: Brave new world
830: 0    : Contemporánea (Barcelona)|v185
```

Ilustración 1. Registro MARC21. Biblioteca Nacional de España

Como hemos dicho el formato MARC en realidad es un formato para que los ordenadores procesen los registros bibliográficos. Podemos ver cómo es “internamente” para un ordenador este registro, de acuerdo a la norma ISO 2709:

```
02938nam                                a2200469                                b
45000010015000000050017000150080041000320160031000730170051001040200018
00155040002900173041001300202080001900215100003000234240003000264245008
60029425000280038026000320040830000180044049000240045859400360048283000
36005189560006005549800105005609800104006659800106007699800105008759800
10900980980011301089980011301202980009101315980009301406980009301499980
00940159298000940168698001140178098001150189498001150200998001140212498
0011502238980011502353 bimo0001846064 20110509          030416s2003  sp |
||| ||spa 7 -abimoBNE20030284420-2SpMaBN -aB 8785-2003-bOficina Depósito
Legal Barcelona -a84-9759-425-8 -aSpMaBN-bspa-cSpMaBN-erdc 1 -aspa-
heng 0-a821.111-31"19" 10-aHuxley, Aldous-d1894-1963 10-aBrave new world-
IEspañol 13-aUn mundo feliz-cAldous Huxley ; traducción de Ramón Hernández-
h[Texto impreso] -a1ª ed. en este formato -aBarcelona-bDebolsillo-c2003
-a254 p.-c19 cm 1 -aContemporánea-v185 -aTraducción de: Brave new world 0-
aContemporánea (Barcelona)-v185 -a2 -.017. -aB 5975-2008-.020. -a978-84-
9759-425-7-...
```

Como vemos, la cadena de etiquetas que habitualmente asociamos al formato MARC, internamente se corresponden con esta cadena de caracteres.

1.3.1. MARCXML

Hemos visto que xml es un lenguaje de marcado que permite determinar la estructura de un documento. Por esta razón no ha sido difícil adaptar el formato MARC al lenguaje xml. De este modo lo que hacemos es convertir el registro MARC, con sus etiquetas, indicadores y códigos de subcampo a un registro XML. Vemos, por ejemplo, unas etiquetas MARC para expresar el autor y el título (campos 100 y 245 con sus indicadores y sus códigos de subcampo):

100: 10 \$aHuxley, Aldous \$d1894-1963

245: 13 \$aUn mundo feliz \$cAldous Huxley ; traducción de Ramón Hernández \$h[Texto impreso]

Y ahora vemos estos mismos campos convertidos a MARCXML:

```
<datafield tag="100" ind1="1" ind2="0">
```

```
  <subfield code="a">Huxley, Aldous</subfield>
```

```
  <subfield code="d">1894-1963.</subfield>
```

```
</datafield>
```

```
<datafield tag="245" ind1="1" ind2="3">
```

```
  <subfield code="a">Un mundo feliz</subfield>
```

```
  <subfield code="c">Aldous Huxley ; traducción de Ramón  
  Hernández.</subfield>
```

```
  <subfield code="h">Texto impreso</subfield>
```

```
</datafield>
```

Se puede ver fácilmente que la estructura de MARC se adapta sin problemas a la sintaxis XML convirtiendo las etiquetas, indicadores y subcampos en diferentes elementos y atributos.

La ventaja esencial que tiene poder convertir MARC a MARCXML es que una vez que lo tenemos en xml hemos convertido un formato propio del ámbito bibliotecario y que sólo los sistemas de gestión bibliotecarios suelen poder interpretar, en un lenguaje más

universal. Además, como un documento xml siempre se puede convertir en otro documento xml diferente (mediante las hojas de estilo) existe la posibilidad de convertir el documento MARCXML a cualquier otro modelo de datos que se base en la sintaxis xml. Y hoy en día prácticamente todos los modelos de datos (o metadatos) utilizan xml como lenguaje fundamental.

1.4. Dublin Core Metadata Initiative

Dublin Core es el conjunto de metadatos más conocido en el ámbito bibliotecario. Su origen se remonta a 1995 cuando tuvo lugar una reunión de expertos en metadatos en Dublin (Ohio), promovida por la OCLC. La idea fundamental de Dublin Core era proporcionar una forma fácil de describir recursos, especialmente electrónicos.

Originalmente se concibió como un conjunto de etiquetas que deberían ser generadas por el autor del documento HTML con la finalidad de facilitar su identificación y posterior recuperación en Internet. No obstante, este modelo ha llamado la atención de diversas comunidades de profesionales interesados en la descripción de recursos en museos, bibliotecas, organizaciones comerciales...

Para esto se definieron quince elementos. Se puede decir que algunos de ellos están más directamente relacionados con el contenido del recurso, otros con la propiedad intelectual y otros con aspectos más formales¹³ :

Contenido:

- **Título:** el nombre dado a un recurso, habitualmente por el autor.
Etiqueta: DC.Title

¹³ <http://www.rediris.es/search/dces/>

- **Claves:** los tópicos del recurso. Típicamente, Subject expresará las claves o frases que describen el título o el contenido del recurso. Se fomentará el uso de vocabularios controlados y de sistemas de clasificación formales.
Etiqueta: DC.Subject
- **Descripción:** una descripción textual del recurso. Puede ser un resumen en el caso de un documento o una descripción del contenido en el caso de un documento visual.
Etiqueta: DC.Description
- **Fuente:** secuencia de caracteres usados para identificar unívocamente un trabajo a partir del cual proviene el recurso actual.
Etiqueta: DC.Source
- **Lengua:** lengua/s del contenido intelectual del recurso.
Etiqueta: DC.Language
- **Relación:** es un identificador de un segundo recurso y su relación con el recurso actual. Este elemento permite enlazar los recursos relacionados y las descripciones de los recursos.
Etiqueta: DC.Relation
- **Cobertura:** es la característica de cobertura espacial y/o temporal del contenido intelectual del recurso. La cobertura espacial se refiere a una región física, utilizando por ejemplo coordenadas. La cobertura temporal se refiere al contenido del recurso, no a cuándo fue creado (que ya lo encontramos en el elemento Date).
Etiqueta: DC.Coverage

Propiedad Intelectual:

- **Autor o Creador:** la persona o organización responsable de la creación del contenido intelectual del recurso. Por ejemplo, los autores en el caso de documentos escritos; artistas, fotógrafos e ilustradores en el caso de recursos visuales.

Etiqueta: DC.Creator

- **Editor:** la entidad responsable de hacer que el recurso se encuentre disponible en la red en su formato actual.

Etiqueta: DC.Publisher

- **Otros Colaboradores:** una persona u organización que haya tenido una contribución intelectual significativa, pero que esta sea secundaria en comparación con las de las personas u organizaciones especificadas en el elemento Creator. (por ejemplo: editor, ilustrador y traductor).

Etiqueta: DC.Contributor

- **Derechos:** son una referencia (por ejemplo, una URL) para una nota sobre derechos de autor, para un servicio de gestión de derechos o para un servicio que dará información sobre términos y condiciones de acceso a un recurso.

Etiqueta: DC.Rights

Aspectos formales:

- **Fecha:** una fecha en la cual el recurso se puso a disposición del usuario en su forma actual. Esta fecha no se tiene que confundir con la que pertenece al elemento Coverage, que estaría asociada con el recurso en la medida que el contenido intelectual está de alguna manera relacionado con aquella fecha.

Etiqueta: DC.Date

- **Tipo del Recurso:** la categoría del recurso. Por ejemplo, página personal, romance, poema, diccionario, etc.

Etiqueta: DC.Type

- **Formato:** es el formato de datos de un recurso, usado para identificar el software y, posiblemente, el hardware que se necesitaría para mostrar el recurso.

Etiqueta: DC.Format

- **Identificador del Recurso:** secuencia de caracteres utilizados para identificar unívocamente un recurso. Ejemplos para recursos en línea pueden ser URLs i URNs. Para otros recursos pueden ser usados otros formatos de identificadores, como por ejemplo ISBN ("International Standard Book Number").

Etiqueta: DC.Identifier

En principio la idea de DC es que estos elementos fueran sencillos y que no fuera necesaria la experiencia ni los conocimientos de un catalogador para procesar, por ejemplo, las páginas web que continuamente están apareciendo. Para contribuir a la sencillez de la descripción, en DC los quince elementos son opcionales, repetibles y pueden aparecer en cualquier orden.

Sin embargo DC no concluyó con la definición de estos elementos. Pronto se vio que era necesario precisar algo más el contenido de estos quince campos y apareció el llamado **Dublin Core cualificado**. Con los calificadores lo que se pretende es que el significado de un elemento sea más específico. Un elemento refinado comparte el significado del elemento no cualificado, pero con un alcance más restrictivo.

Metadatos en mundo bibliotecario

Abril 2014

| DCMES Element | Element Refinement(s) | Element Encoding Scheme(s) |
|----------------------|--|--|
| <u>Title</u> | <u>Alternative</u> | - |
| <u>Creator</u> | - | - |
| <u>Subject</u> | - | <u>LCSH</u> <u>MeSH</u> <u>DDC</u> <u>LCC</u> <u>UDC</u> |
| <u>Description</u> | <u>Table Of Contents</u> <u>Abstract</u> | - |
| <u>Publisher</u> | - | - |
| <u>Contributor</u> | - | - |
| <u>Date</u> | <u>Created</u> <u>Valid</u> <u>Available</u> <u>Issued</u> <u>Modified</u> <u>Date Accepted</u> <u>Date Copyrighted</u> <u>Date Submitted</u> | <u>DCMI Period</u> <u>W3C-DTF</u> |
| <u>Type</u> | - | <u>DCMI Type Vocabulary</u> |
| <u>Format</u> | - | <u>IMI</u> |
| | <u>Extent</u> | - |
| <u>Identifier</u> | <u>Medium</u> | - |
| | - | <u>URI</u> |
| <u>Source</u> | <u>Bibliographic Citation</u> | - |
| | - | <u>URI</u> |
| <u>Language</u> | - | <u>ISO 639-2RFC 3066</u> |
| <u>Relation</u> | <u>Is Version Of</u> <u>Has Version</u> <u>Is Replaced By</u> <u>Replaces</u> <u>Is Required By</u> <u>Requires</u> <u>Is Part Of</u> <u>Has Part</u> <u>Is Referenced By</u> <u>References</u> <u>Is Format Of</u> <u>Has Format</u> <u>Conforms To</u> | <u>URI</u> |

| | | |
|-----------------------------|---|---|
| <u>Coverage</u> | <u>Spatial</u> | <u>DCMI Point</u> <u>ISO 3166</u> <u>DCMI Box</u> <u>TGN</u> |
| | <u>Temporal</u> | <u>DCMI Period</u> <u>W3C-DTF</u> |
| <u>Rights</u> | <u>Access Rights</u> | - |
| | <u>License</u> | <u>URI</u> |
| <u>Audience</u> | <u>Mediator</u> <u>Education Level</u> | - |
| <u>Provenance</u> | - | - |
| <u>Rights Holder</u> | - | - |
| <u>Instructional Method</u> | - | - |
| <u>Accrual Method</u> | - | - |
| <u>Accrual Periodicity</u> | - | - |
| <u>Accrual Policy</u> | - | - |

Ilustración 2. Elementos de Dublin Core Cualificado. DCMI¹⁴.

El resultado de la ampliación de los 15 elementos ha sido la creación de un amplio conjunto de elementos definidos por la Dublin Core Metadata Initiative, que se recogen en el documento DCMI Metadata Terms¹⁵.

Un siguiente paso fue la definición de “perfiles de aplicación”¹⁶. Un perfil de aplicación es un conjunto de metadatos, políticas y recomendaciones definidas para un uso concreto.

¹⁴ <http://dublincore.org/documents/usageguide/qualifiers.shtml>

¹⁵ <http://dublincore.org/documents/dcmi-terms/>

¹⁶ HEERY, Rachel y PATEL, Manjula: “Application profiles: mixing and matching metadata schemas” <http://www.ariadne.ac.uk/issue25/app-profiles/>

Los perfiles de aplicación se pueden definir como esquemas que consisten en elementos de datos tomados de uno o más namespaces, espacios de nombre.

Conviene aclarar también el concepto de “espacio de nombre” o namespace. Según la Wikipedia¹⁷ se define:

Un espacio de nombres XML es una recomendación W3C para proporcionar elementos y atributos con nombre único en una instancia XML. Una instancia XML puede contener nombres de elementos o atributos procedentes de más de un vocabulario XML. Si a cada uno de estos vocabularios se le da un espacio de nombres, se resuelve la ambigüedad existente entre elementos o atributos que se llamen igual. Los nombres de elementos dentro de un espacio de nombres deben ser únicos.

Por tanto, un espacio de nombre es el conjunto de elementos que una entidad o agencia ha definido para un propósito determinado. Y un perfil de aplicación permite combinar diferentes espacios de nombre bajo un único esquema.

Es decir, los espacios de nombre se utilizan para, en un mismo documento xml, poder utilizar elementos que están definidos ya en algún otro vocabulario. Parte de los elementos de Europeana utilizan elementos de Dublin Core, por eso decimos que utilizan dos espacios de nombres: el de Dublin Core y el de Europeana.

Volviendo de nuevo a la Wikipedia:

El alcance de la declaración de un prefijo de espacio de nombres comprende desde la etiqueta de inicio de un elemento XML, en la que se declara, hasta la etiqueta final de dicho elemento XML. En las etiquetas vacías, correspondientes a elementos sin "hijos", el alcance es la propia etiqueta.

¹⁷ http://es.wikipedia.org/wiki/Espacio_de_nombres_XML

Es decir, que cuando veamos un documento xml en el que hay varios espacios de nombres, cuando se quiera declarar uno encontraremos expresiones del tipo: `dc:title>Eight weeks</dc:title>`. Este “dc:” implica que ese elemento está definido de acuerdo al vocabulario de Dublin Core.

En Dublin Core existe el DC-Lib, que es el perfil de aplicación para bibliotecas . En este caso, DC-Lib toma sus elementos de los espacios de nombres de la DCMI metadata terms y de los elementos MODS definidos por la Biblioteca del Congreso de los Estados Unidos, Library of Congress.

Los elementos que define el perfil de aplicación DC-Lib para bibliotecas son los siguientes:

- Title
- Alternative
- Creator
- Contributor
- Publisher
- Subject
- Description
- Abstract
- Date
- Created
- Valid
- Available
- Issued
- Modified
- DateCopyrighted
- DateSubmitted
- DateAccepted
- DateCaptured
- Type
- Format
- Extent
- Medium
- Identifier
- BibliographicCitation
- Source
- Language
- Relation
- IsVersionOf
- IsFormatOf
- HasFormat
- IsReplacedBy
- Replaces
- IsPartOf
- HasPart

- Requires
- IsReferencedBy
- References
- Coverage
- Spatial
- Temporal
- Rights
- Audience
- Edition
- Location

En total, más de cuarenta elementos. Como vemos, los sencillos quince elementos se han ido complicando al sumarse los calificadores y los perfiles de aplicación. Aunque Dublin Core pretende ser sencillo todos estos cambios han ido haciéndolo más complejo, a la vez que más preciso. Por otra parte, todos los elementos, calificadores y perfiles de aplicación requieren definiciones para precisar su significado y así garantizar la uniformidad de su uso y evitar ambigüedades.

Por las características que hemos estado viendo hasta ahora, no sorprende el hecho de que tampoco sea difícil adaptar Dublin Core al lenguaje XML, si bien, originalmente se pensó para que fuera fácil de incorporar al código HTML de las páginas web.

Efectivamente, el conjunto de elementos de Dublin Core es perfectamente adaptable a la estructura de xml. Aquí vemos el esquema XML para los elementos Dublin Core:

```
<?xml version="1.0"?>

<metadata
  xmlns="http://example.org/myapp/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://example.org/myapp/ http://example.org/myapp/schema."
  xmlns:dc="http://purl.org/dc/elements/1.1/"
  xmlns:dcterms="http://purl.org/dc/terms/">

  <dc:title>
    UKOLN
  </dc:title>
  <dcterms:alternative>
    UK Office for Library and Information Networking
  </dcterms:alternative>
  <dc:subject>
    national centre, network information support, library
    community, awareness, research, information services,public
    library networking, bibliographic management, distributed
    library systems, metadata, resource discovery,
    conferences,lectures, workshops
  </dc:subject>
  <dc:subject xsi:type="dcterms:DDC">
    062
  </dc:subject>
  <dc:subject xsi:type="dcterms:UDC">
    061(410)
  </dc:subject>
```

```
<dc:description>
  UKOLN is a national focus of expertise in digital information
  management. It provides policy, research and awareness services
  to the UK library, information and cultural heritage communities.
  UKOLN is based at the University of Bath.
</dc:description>
<dc:description xml:lang="fr">
  UKOLN est un centre national d'expertise dans la gestion de l'information
  digitale.
</dc:description>
<dc:publisher>
  UKOLN, University of Bath
</dc:publisher>
<dcterms:isPartOf xsi:type="dcterms:URI">
  http://www.bath.ac.uk/
</dcterms:isPartOf>
<dc:identifier xsi:type="dcterms:URI">
  http://www.ukoln.ac.uk/
</dc:identifier>
<dcterms:modified xsi:type="dcterms:W3CDTF">
  2001-07-18
</dcterms:modified>
<dc:format xsi:type="dcterms:IMT">
  text/html
</dc:format>
<dcterms:extent>
  14 Kbytes
</dcterms:extent>
</metadata>
```

Ilustración 3. Ejemplo DC Cualificado. DCMI¹⁸.

Esta posibilidad de utilizar los elementos DC con el lenguaje XML es, sin duda, una ventaja para DC porque lo hace compatible con cualquier otro conjunto de metadatos que se codifiquen en XML. De este modo, y como el formato MARC, según veíamos antes, también se puede estructurar en XML, es posible e incluso sencillo convertir un registro MARC en uno DC y viceversa.

¹⁸ <http://dublincore.org/documents/dc-xml-guidelines/>

Registro original en formato MARC

```
=LDR 01087nkmka2200253 b 45 0
=001 a\4756903
=005 20100521\
=007 kd*aob
=008 051017s16uu\ita\k\
=040 \ $aSpMaBN
=100 0\ $aAnónimo italiano$ds. XVII
=245 14$a[La Inmaculada Concepciónn]$h[Material gráfico]
=260 \ $c[16--]
=300 \ $a1 dibujo sobre papel agarbanzado oscuro verjurado$bpluma, pincel, tinta china y aguada$c271 x 181 mm
=500 \ $aAnotación manuscrita a lápiz en borde inferior: "4"
=510 4\ $aBarcia Catáogo de dibujos de la BN. 1906$cp. 614 n. 8059
=520 8\ $aLa figura de la Virgen, aislada en la parte derecha del papel, está sobre nubes, doblada una rodilla y cruzadas las manos en el pecho. A su izquierda, algo más arriba, parte de un coro de Ángeles
=593 \ $aBarcia lo cataloga como anónimo italiano del s. XVII
=594 \ $aTít. tomado de Barcia
=600 0\ $aVirgen María
=655 \7$aDibujos religiosos$zItalia$ys.XVII
=655 \7$aDibujos a pluma$zItalia$ys.XVII
=655 \7$aDibujos a la aguada$zItalia$ys. XVII
=956 \ $a1
```

Dublin Core

```
<?xml version="1.0" encoding="utf-8" ?>
```

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:dc="http://purl.org/dc/elements/1.1/">
```

```
<rdf:Description>
```

```
<dc:title>[La Inmaculada Concepción] [Material gráfico</dc:title>
<dc:creator>Anónimo italianos. XVII</dc:creator>
<dc:type>still image</dc:type>
<dc:type>Dibujos religiosos Italia</dc:type>
<dc:type>Dibujos a pluma Italia</dc:type>
<dc:type>Dibujos a la aguada Italia</dc:type>
<dc:publisher />
<dc:date>[16-</dc:date>
<dc:language>k</dc:language>
<dc:description>La figura de la Virgen, aislada en la parte derecha del papel, está sobre
nubes, doblada una rodil la y cruzadas las manos en el pecho. A su izquierda, algo
más arriba, parte de un coro de á</dc:description>
<dc:description>Anotación manuscrita a lápiz en borde inferior: "4"</dc:description>
<dc:description>Barcia Catálogo de dibujos de la BN. 1906</dc:description>
<dc:description>La figura de la Virgen, aislada en la parte derecha del papel, está sobre
nubes, doblada una rodil la y cruzadas las manos en el pecho. A su izquierda, algo
más arriba, parte de un coro de ángeles</dc:description>
<dc:description>Barcia lo cataloga como anonimo italiano del s. XVII</dc:description>
<dc:description>Tít. tomado de Barcia</dc:description>
<dc:subject>Virgen María</dc:subject>
</rdf:Description>
</rdf:RDF>
```

Como vemos la conversión es posible y sencilla (hay varias herramientas que permiten llevarla a cabo. Sin embargo, hay que señalar que por el camino se han perdido algunas de las precisiones que el detallado formato MARC aportaba. Donde había notas: 500, 510, 520, 593, y 594, en DC encontramos sólo el campo “Descripción”.

1.5. Europeana Semantic Elements

Ahora vamos a conocer el modelo de metadatos conocido como Europeana Semantic Elements (ESE). El objetivo principal es que tengáis una visión de en qué consisten estos elementos, por qué surgen y que conozcáis los documentos en los que se encuentran definidos en detalle.

Para comprender este esquema hay que tener en cuenta que surge para dar respuesta a las necesidades que aparecen durante la creación de Europeana.

Europeana (www.europeana.eu) es un portal que provee acceso integrado a objetos digitales de organizaciones relacionadas con el patrimonio cultural de toda la Unión Europea. Incorpora materiales de museos, bibliotecas, archivos y archivos audiovisuales con el objetivo de que la riqueza multicultural y multilingüística de Europa pueda encontrarse en único entorno en línea.

Europeana, a su vez, tiene en su origen una estrecha relación con otro proyecto europeo, *The European Library* (TEL, www.theeuropeanlibrary.org). En este caso se trata de un portal que da acceso a los catálogos y a las bibliotecas digitales de las bibliotecas nacionales europeas.

Ambos proyectos se diferencian por tanto en que Europeana da acceso sólo a objetos digitales e incluye todo tipo de instituciones culturales mientras que en TEL hasta el momento sólo participan bibliotecas nacionales y da acceso no sólo a objetos digitales sino también a los catálogos.

Para hacer esto, Europeana recolecta e indexa los metadatos descriptivos asociados a los objetos digitales. Como no hay un ningún estándar de metadatos que se pueda

aplicar a los diferentes tipos de instituciones participantes, se ha desarrollado un conjunto de metadatos que permitirá proveer toda la información necesaria para soportar las funcionalidades requeridas por los usuarios e imprescindible para el funcionamiento del sistema.

Los Europeana Semantic Elements V3.41 (ESE) son una actualización del conjunto de metadatos que se utilizaron en el prototipo de noviembre de 2008. Se trata de una aplicación basada en el Perfil de Aplicación Dublin Core y proporciona un conjunto genérico de términos que pueden ser aplicados a diferentes materiales (los que proceden de bibliotecas, archivos, museos, etc.) y permite a los que aportan dichos materiales aprovechar la riqueza de sus descripciones.

Para proporcionar los metadatos en el formato ESE, es necesario que las instituciones que aportan materiales *mapeen* (establezcan correspondencias) entre sus propios metadatos y el formato ESE. Para facilitar su trabajo se ha creado un esquema XML.

En definitiva, al igual que en su origen ocurrió con Dublin Core, los ESE surgen con la intención de ser un número de campos relativamente reducido, fáciles de utilizar y suficientemente amplios para poder dar cabida a información procedente de fuentes tan diversas como bibliotecas, archivos, museos, etc. que no comparten normas descriptivas.

Como acabamos de mencionar, los ESE surgen al hilo de la creación de Europeana. Esto hace que nos encontremos ante un conjunto de metadatos muy prácticos, dado que se nutren de la experiencia directa y de los problemas concretos que se estaban encontrando a la hora de indexar los millones de documentos que en la oficina central del proyecto estaban recibiendo.

Este carácter práctico queda de manifiesto en el hecho de que ya nos encontremos en la versión 2.4 de una Especificación que surge por primera vez hace poco más de tres años, en 2008.

Los ESE se han convertido en el conjunto de elementos obligatorios para todos aquellos que quieran que sus registros bibliográficos se encuentren en Europeana. Esto los

convierte en un estándar europeo de facto porque Europeana es, sin duda, un lugar en el que la mayoría de las bibliotecas digitales quieren estar.

Para conocer cuáles son estos elementos que componen los elementos semánticos de Europeana contamos con una página donde Europeana recoge toda la documentación disponible¹⁹. De entre ella nos vamos a guiar por el documento *Europeana Semantic Elements Specifications and Guidelines*²⁰.

En primer lugar cabe señalar que los ESE son fundamentalmente una extensión de Dublin Core, de manera que vamos a encontrar un gran número de elementos comunes. De hecho, en el propio documento que recoge las especificaciones se señala que el esquema con el que se valida el documento reconoce como válidos todos los términos de Dublin Core, aunque luego no los incorpore al portal.

Vamos a ver ya qué elementos componen la especificación ESE:

| Elementos obligatorios | Elementos adicionales | Elementos recomendados | Elementos facilitados por Europeana |
|--------------------------------------|-----------------------|------------------------|-------------------------------------|
| dc:title or dc:description | dc:format | dcterms:alternative | europa:country |
| dc:language for text objects | dcterms:extent | dc:creator | europa:language |
| europa:dataProvider | dcterms:medium | dc:contributor | europa:uri |
| europa:isShownAt or europa:isShownBy | dc:identifier | dc:date | europa:userTag |
| | dc:rights | dcterms:created | europa:year |

¹⁹ <http://pro.europeana.eu/ese-documentation/>

²⁰ <http://pro.europeana.eu/documents/900548/2eee7beb-b9d8-4532-a089-8e8d6df38ce7>

| | | | |
|---|------------------------------|-------------------------------|--|
| europiana:provider | dcterms:provenance | dcterms:issued | |
| dc:subject or dc:type or dc:coverage or dcterms:spatial | dc:relation | dcterms:temporal | |
| | dcterms:conformsTo | dc:publisher | |
| europiana:rights | dcterms:hasFormat | dc:source | |
| europiana:type | dcterms:isFormatOf | dcterms:isPartOf | |
| | dcterms:hasVersion | europiana:object | |
| | dcterms:isVersionOf | | |
| | dcterms:hasPart | | |
| | dcterms:isReferencedBy | | |
| Elementos obligatorios | Elementos adicionales | Elementos recomendados | Elementos facilitados por Europeana |
| | dcterms:references | | |
| | dcterms:isReplacedBy | | |
| | dcterms:replaces | | |
| | dcterms:isRequiredBy | | |
| | dcterms:requires | | |
| | dcterms:tableOfContents | | |
| | europiana:unstored | | |

Tabla 1. Elementos de ESE. Europeana Semantic Elements Specification...

Otra forma de presentar estos mismos elementos que igual ayuda a ver las relaciones entre ellos con más claridad es la siguiente:

| Source | Element | Element Refinement(s) |
|--------|-------------|---|
| DC | title | alternative |
| DC | creator | |
| DC | subject | |
| DC | description | tableOfContents |
| DC | publisher | |
| DC | contributor | |
| DC | date | created; issued |
| DC | type* | |
| Source | Element | Element Refinement(s) |
| DC | format | extent; medium |
| DC | identifier | |
| DC | source | |
| DC | language* | |
| DC | relation | isVersionOf; hasVersion; isReplacedBy; replaces; isRequiredBy; requires; isPartOf; hasPart; isReferencedBy; references; isFormatOf; hasFormat; conformsTo |

| | | |
|---------------|----------------|------------------------------|
| Europeana | | isShownBy; isShownAt |
| DC | coverage | spatial; temporal |
| DC | rights* | |
| DC terms | provenance | |
| Europeana | country | |
| Europeana | dataProvider | |
| Europeana | language* | |
| Europeana | object | |
| Source | Element | Element Refinement(s) |
| Europeana | provider | |
| Europeana | rights* | |
| Europeana | type* | |
| Europeana | unstored | |
| Europeana | uri | |
| Europeana | userTag | |
| Europeana | year | |

Tabla 2. Elementos de ESE. European Semantic Elements Specification...

Para comprender cómo se comporta la Especificación de los ESE conviene tener claros dos conceptos que están presentes en su definición:

El primero de ellos es el concepto de “**espacio de nombre**” o *namespace*, que definimos antes.

Es importante tener este concepto claro porque si nos fijamos en los elementos que componen los ESE, veremos que hay algunos que aparecen, en principio, repetidos. Concretamente, los elementos “type”, “language” y “rights” figuran dos veces. Una vez se definen de acuerdo al espacio de nombres de Dublin Core y otra de acuerdo al de Europea.

Por ejemplo, para Dublin Core, el elemento type se define como: la naturaleza o género del recurso. “Tipo” incluye términos que describen categorías generales, funciones, géneros o niveles de agregación del contenido. Se recomienda utilizar un vocabulario controlado (por ejemplo, el vocabulario “DC Type” disponible en <http://dublincore.org/documents/dcmi-type-vocabulary/>).

Para Europea, el elemento type es simplemente el tipo de material del recurso.

El otro concepto que conviene tener claro para comprender la estructura de los ESE es el de “**Refinamiento**” (*Refinement*). Entre la documentación de la Dublin Core Metadata Initiative encontramos la siguiente definición²¹:

Además de aportar una definición y un identificador para cada una de las propiedades que declara, DCMI también describe las relaciones entre esas propiedades. Si las definiciones de dos propiedades son tales que siempre que dos recursos estén relacionados por la primera propiedad están también relacionados por la segunda, decimos que la primera “refina” o es un “refinamiento” de la segunda propiedad.

²¹ <http://dublincore.org/documents/dc-elem-refine/>

Así, por ejemplo, la definición de la propiedad `dcterms:created` es: “Fecha de creación del recurso”; y la definición de la propiedad `dc:date` es: “una fecha asociada con un evento en el ciclo vital de un recurso”. La fecha de creación de un recurso siempre es “una fecha asociada con un evento en el ciclo vital del recurso”, luego la propiedad `dcterms:created` refina la propiedad `dc:date`.

En este punto no vamos a repasar exhaustivamente cada uno de los elementos en primer lugar porque muchos de ellos son Dublin Core y ya lo hemos visto en otro apartado del curso; por otra parte, toda la información necesaria está en el documento que recoge las especificaciones. Vamos a limitarnos a explicar los elementos que son específicos de ESE.

Para cada elemento se da la siguiente información:

- **Nombre del elemento.**
- **Espacio de nombre (namespace).**
- **URI.**
- **Etiqueta.**
- **Definición.**
- **Nota de Europeana** (una nota que informa el uso que se hace del elemento dentro del portal).
- **Obligatoriedad y ocurrencia** (si es obligatorio y si es repetible).
- **Opciones de visualización y búsqueda en Europeana:** Indica si los elementos se utilizan para la búsqueda sencilla (todos lo hacen), como facetas, para la línea de tiempo, en la búsqueda avanzada, y en la visualización completa de los registros.
- **Ejemplo.**

Veamos cómo define el elemento *country*:

| Element name: country | |
|---------------------------------------|--|
| Namespace | europeana |
| URI | http://www.europeana.eu/schemas/ese/country |
| Label | Country |
| Definition | The name of the country of the data provider or “Europe” in the case of Europe-wide projects. |
| Europeana note | Europeana enters the name of the country of the data provider as part of the ingest process but can only do this accurately if Providers (aggregators) supply datasets by country. |
| Obligation & Occurrence | Mandatory (Minimum: 1, Maximum: 1) |
| Europeana search and display features | Simple search X Facet X (Country) Timeline Advanced search Full search result display X (Country Line) |
| Example | <europeana:country>lithuania</europeana:country> |

Tabla 3. Ejemplo de un elemento de ESE. European Semantic Elements Specification...

Como vemos, para cada elemento se recoge la información que determina cómo se debe utilizar dicho elemento en el portal. Centrándonos en los elementos que son específicos de los ESE (es decir, excluyendo los que son elementos de Dublin Core), nos encontramos con los siguientes:

Country (país): País del proveedor de los datos (registros bibliográficos):

dataProvider (proveedor de datos): Nombre o identificador de la organización que aporta datos a Europeana. Este elemento se incluye para distinguir aquellos casos en que una institución aporta sus datos a través de otra que hace de “agregadora” de

diferentes instituciones. La institución que realiza efectivamente el envío de datos se consigna en el elemento `europiana:provider`, y en el caso de que una organización aporte sus propios datos el valor de ambos elementos `dataProvider` y `provider` será el mismo²².

isShownAt (mostrado en): Contiene una URL donde se muestra el objeto digital junto a su información contextual en el entorno del proveedor de datos. Por ejemplo, si es necesario mostrar un esquema jerárquico que los ESE no recogen se enlazará al sitio donde el proveedor de datos contiene esa información.

isShownBy (mostrado por): Contiene una URL donde se muestra directamente el objeto digital. Es complementario del anterior y siempre es obligatorio que los registros incluyan uno de los dos con un enlace válido.

Language (lengua): En este caso, Europeana también reconoce el elemento `dc:language`, que recoge la información de la lengua del recurso. El elemento `europiana:language` se emplea a partir de la información del `dc:language` y tiene como objetivo normalizar las lenguas de este elemento según el código ISO 639-2 con el fin de establecer facetas en las búsquedas.

Object (objeto): Contiene una URL donde se muestre una imagen en la mejor resolución posible a partir de la cual se puedan generar pequeñas imágenes para su uso en el portal. Muchas veces contendrá el mismo valor que `europiana:isShownBy`

Provider (proveedor): Nombre de la organización que envía los datos a Europeana.

Rights (derechos): También se encuentra el valor `dc:rights`. Para este elemento el valor es el de la información sobre los derechos relativos al objeto. En el caso del elemento `europiana:rights`, el valor es el uso que Europeana puede hacer de estas obras. Para

²² En España, por ejemplo, existe Hispana, que gestionado por el Ministerio de Cultura recolecta multitud de bibliotecas digitales y las “agrega” a Europeana. Más información en: <http://roai.mcu.es/es/inicio/inicio.cmd>.

ello se debe aportar una URL construida de forma controlada según la información disponible en:

<http://version1.europeana.eu/web/guest/technical-requirements/>

Type (tipo): En este caso también hay un valor aceptado como dc:type. La diferencia es que mientras éste entiende tipo de forma amplia, como tipo de documento, género, categoría..., europeana:type se entiende sólo como tipo de material y sólo comprende cuatro valores: TEXT, IMAGE, SOUND o VIDEO.

Unstored (sin almacenar): Es un elemento contenedor que incluye toda la información relevante que de otro modo no podría incluirse en los elementos de ESE.

URI: Genera una URI para identificar unívocamente el registro dentro de Europeana. Lo crea Europeana a partir de los identificadores únicos que se aportan en los metadatos.

userTag (etiqueta de usuario): Incluye las etiquetas que los usuarios registrados en Europeana pueden incorporar a los registros.

Year (año): Un punto del tiempo asociado con un evento de original analógico digitalizado o de un objeto nacido digital. Se obtiene del elemento dc:date y se emplea para la funcionalidad de la línea del tiempo y para la faceta de fecha.

La conclusión principal al estudiar los Europeana Semantic Elements es que se trata de un conjunto de datos que toman lo que les interesa de un lenguaje previamente existente y definido (el Dublín Core) y añade aquellos otros elementos que en función de las necesidades concretas de los desarrolladores del portal Europeana van considerándose imprescindibles. Es interesante, en todo caso, destacar que si bien Dublin Core funciona como un estándar por su amplia aceptación y por se referencia, incluso para la construcción de los ESE, estos son de estándar *de facto*, al menos en Europa, ya que la mayor parte de los proyectos de bibliotecas, archivos o museos digitales deben implementarlos para participar en Europeana.

Metadatos en mundo bibliotecario

Abril 2014

Un ejemplo de un registro de la BNE en Europeana (y por tanto en Europeana Semantic Elements):

Title: España [Material cartográfico] : Dedicada Al Rey N.S.D. Carlos III Que Dios Guarde Por Antonio Sanz Año de 1767

Creator: López, Tomás1730-1802

Creator: Sanz, Antoniofl. 1728-1770

Coverage: España

Date: 1767

Type: cartographic | Material cartográfico

Description: El mapa se publicó en "Kalendario Manual y guía de Forasteros en Madrid para el año de MDCCCLXVII"

En el ángulo inferior derecho, cartela del título en cornucopia adornada con motivos vegetales

Es una nueva edición con ligeras variantes de la plancha original (cat. nº 70)

Líter, C. La obra de Tomás López, 2002

Orientado. - Márgenes graduados. - Relieve por montes perfil y sombreado. - Red hidrográfica. -

Divisiones administrativas diferenciadas por color. - Costas sombreadas con un fino rayado

Tomás López grabó para la Guía de Forasteros, numerosos mapas de España con ligeras variantes. Esta colaboración abarcó prácticamente toda su vida profesional desde el año 1759, continuando su publicación incluso después de su muerte

Provider: Biblioteca Nacional de España| Spain

View item at: [Biblioteca Nacional de España](#)

Identifier: 43786.jpg oai:bibliotecadigitalhispanica.bne.es:1860267

Language: es

Publisher:[Madrid Antonio Sanz]

1.6. Otros modelos de metadatos: METS, PREMIS, ONIX, MODS, MIX, y otros estándares de la Biblioteca del Congreso.

Hasta ahora hemos visto modelos de metadatos que responden casi completamente a lo que es normal en una biblioteca, esto es, describir documentos: dar datos sobre su autoría, título, materia, edición, etc. Ahora vamos a hablar de otros modelos que es importante conocer.

Los más importantes de estos modelos son METS y PREMIS. Precisamente por eso y por su estrecha relación con la preservación de los objetos digitales (PREMIS está específicamente pensado para eso), se verán con más detalle en el Módulo 3. Aquí simplemente veremos lo más básico.

1.6.1 Metadata Encoding and Transmission Standard (METS)

METS es una norma que permite expresar metadatos descriptivos, administrativos y estructurales relativos a objetos en una biblioteca digital. Se basa en los esquemas del lenguaje XML. Esta norma la mantiene la Biblioteca del Congreso y está siendo desarrollada como iniciativa de la Digital Library Federation.

Sobre METS contamos con tutorial muy bueno de la Biblioteca del Congreso que os recomendamos consultar²³:

Lo primero que se afirma en este tutorial es que, como hemos dicho, la gestión de una biblioteca de objetos digitales requiere la gestión de metadatos sobre esos objetos. A esto añade una precisión importante:

Los metadatos necesarios para gestionar y usar con éxito objetos digitales son más complejos que los que se emplean para gestionar colecciones de documentos impresos y materiales con soporte físico. Una biblioteca puede registrar metadatos descriptivos sobre un libro de su colección, pero el libro nunca se disolverá en una serie de páginas independientes, desconectadas, si la biblioteca no registra los metadatos estructurales relativos a la organización del libro; tampoco los usuarios se verán incapacitados para valorar la obra si la biblioteca no registra que el libro se produjo usando una prensa offset de un tipo determinado. Sin embargo, esto mismo no podría afirmarse para la versión digital de ese mismo libro.

Sin metadatos estructurales, las imágenes y los archivos de texto que conforman el objeto digital tienen poca utilidad, y sin los metadatos técnicos relativos a proceso de digitalización los usuarios no pueden evaluar en qué medida la obra digital es un fiel reflejo del original impreso. Para la gestión interna, la biblioteca debe conocer los metadatos técnicos para poder refrescar y migrar regularmente los contenidos y asegurar la preservación de estos valiosos recursos.

Lo que se afirma es cierto, el objeto digital, como nueva realidad frente al objeto físico, requiere de otros elementos descriptivos para poder gestionarlos adecuadamente. Estos nuevos elementos se suman a los que ya existían para los objetos físicos. Por este motivo, modelos como Dublin Core o MARCXML veremos que se pueden incorporar a la parte descriptiva de METS.

²³ http://www.loc.gov/standards/mets/METSOverview_spa.html#MHead

Para dar respuesta a estas necesidades, el proyecto *Making of America II* (MOA2) propuso un formato de codificación de metadatos descriptivos, administrativos y estructurales para obras textuales y basadas en imágenes. METS, una iniciativa de la *Digital Library Federation*, se desarrolló a partir del trabajo de MOA2, y ofrece un formato basado en XML para codificar los metadatos necesarios para la gestión de objetos digitales y para su intercambio entre repositorios (o entre repositorios y sus usuarios).

Un documento METS consta de siete secciones que ahora simplemente enumeramos:

1. Cabecera METS
2. Metadatos Descriptivos:
3. Metadatos administrativos
4. Sección Archivo
5. Mapa estructural
6. Enlaces estructurales
7. Sección Comportamiento

En general, la mayor ventaja de METS desde el punto de vista bibliotecario es que ofrece la posibilidad de incluir datos sobre la estructura de los documentos. De este modo al ver un registro bibliográfico, por ejemplo de una revista, METS permite describir la estructura del registro diferenciando los objetos digitales que pertenecen a cada fascículo o número de la revista.

Es evidente que para que esto sea posible hay que codificar esa información en el registro. Este proceso es más laborioso que la simple catalogación de un registro bibliográfico pero sin esta información estructural no es posible ni presentar ni preservar adecuadamente los objetos digitales.

Como METS está codificado también en XML permite que dentro del registro se reutilice la información descriptiva que esté disponible en cualquier otro documento XML, de ahí que sea compatible tanto con MARCXML como con Dublin Core.

Además de su utilidad para expresar la estructura de los documentos, METS tiene muchas posibilidades para almacenar información relevante de cara a la preservación a largo plazo de los objetos digitales (fechas de creación y modificación de los registros, relaciones entre copias maestras y documentos derivados, etc.) que lo hacen apto para utilizarlo también como conjunto de metadatos en proyectos de preservación digital. De hecho, METS podría utilizarse dentro del modelo de preservación OAIS (Open Archival Information System) y en combinación con PREMIS.

1.6.2. Preservation Metadata: Implementation Strategies

Hasta ahora hemos visto cuatro modelos de metadatos. Los tres primeros, MARCXML, DC y ESE los podemos asociar a la descripción bibliográfica tradicional. El cuarto, METS, contiene elementos mucho más específicos del ámbito digital y muy especialmente nos permite expresar la estructura de los diferentes ficheros que componen un objeto digital.

PREMIS por su parte está orientado a la descripción de objetos digitales con vistas a su preservación digital a largo plazo. Al igual que en el caso de METS contábamos con un documento introductorio de la Biblioteca del Congreso, en este caso contamos con una guía muy útil escrita por Priscilla Caplan, también de la LC, y que encontramos traducida al español²⁴.

En primer lugar, Caplan rastrea algo de la historia de PREMIS:

PREMIS son las siglas de "Preservation Metadata: Implementation Strategies" (Metadatos de preservación: estrategias de implementación) que es el nombre de un grupo de trabajo internacional patrocinado por OCLC y RLG desde 2003-

²⁴ CAPLAN, Priscilla: *Entender Premis*. Traducido por María Luis Martínez Conde. 2009. <http://www.mcu.es/bibliotecas/docs/MC/PREMIS/Contenido.pdf>.

2005. Este grupo de trabajo elaboró un informe denominado PREMIS Data Dictionary for Preservation Metadata (Diccionario de datos PREMIS de metadatos de preservación) que incluye un diccionario de datos e información sobre los metadatos de preservación. En marzo de 2008 se publicó la segunda versión actualizada. La Library of Congress mantiene un esquema de representación de PREMIS en XML”.

El mantenimiento de PREMIS se realiza a través de la *PREMIS Maintenance Activity*, que patrocina la Biblioteca del Congreso²⁵.

PREMIS como tal es, por tanto el nombre de un Grupo de Trabajo. Sin embargo, cuando hablamos de “PREMIS” en general nos referimos al Diccionario de datos²⁶ que es el fruto del trabajo de este Grupo. Y aunque este Diccionario no es en sí mismo un esquema XML, lo cierto es que se hizo teniendo el lenguaje XML en mente y de hecho se ha desarrollado un esquema XML para PREMIS²⁷.

No avanzaremos nada más sobre PREMIS porque, al igual que METS, se verá con más detalle en el Módulo 3.

Un asunto interesante para reflexionar es la compatibilidad entre METS y PREMIS, dado que ambos presentan elementos concurrentes. Su integración no es complicada pero requiere tomar una serie de decisiones sobre cómo estructurar el documento XML que incluya los datos recogidos en ambos modelos²⁸. Sobre este asunto también tendréis más información en el Módulo 3.

²⁵ <http://www.loc.gov/standards/premis/>.

²⁶ www.loc.gov/standards/premis/v2/premis-2-0.pdf. Disponible también en español gracias a la traducción de Bárbara Muñoz de Solano y Palacios y Lorea Elduayen Pereda en http://www.loc.gov/standards/premis/PREMIS_es.pdf.

²⁷ <http://www.loc.gov/standards/premis/schemas.html>.

²⁸ En <http://www.loc.gov/standards/premis/premis-mets.html> se puede encontrar bibliografía sobre este tema.

1.6.3. Online Information Exchange. ONIX

ONIX es un modelo de metadatos desarrollado dentro del mundo de los editores para su intercambio de información. Aunque su objeto son, al igual que en las bibliotecas, obras bibliográficas, sus necesidades varían sensiblemente. Datos como proveedores, precios, costes, transporte, etc. no son relevantes en la misma medida para editores o bibliotecarios. De este modo, el conjunto de elementos definidos por ONIX se asemeja en algunos aspectos a los de otros modelos como DC, ESE o MARCXML pero también difiere notablemente en otros.

La versión 1.0 del modelo ONIX apareció en enero de 2000 tras los trabajos de la Asociación de Editores Americanos junto a los principales actores del comercio editorial. Actualmente ONIX lo publica y mantiene EDItEUR²⁹ en colaboración con BISG³⁰ en Estados Unidos y con BIC³¹ en el Reino Unido. Es curioso señalar el hecho de que al hablar de las causas que llevaron a crear ONIX se mencionan dos principales problemas: la necesidad de aportar un contenido más rico para los libros en línea; y la gran variedad de formatos que existían para el envío de esa información³². Como vemos, la necesidad de normalización se da en muchos más ámbitos que el bibliotecario.

ONIX, por tanto, no es más que la definición de alrededor de 200 elementos necesarios para la descripción de documentos bibliográficos en el ámbito del comercio de libros. No vamos a entretenernos en verlos todos. Como muestra del tipo de elementos definidos en ONIX presentamos los que aparecen recogidos en el documento *Product Metadata: Best practice for data senders*, elaborado por el BISG, Book Industry Study Group³³. Este trabajo, que pretende ofrecer los principales elementos de ONIX de cara a

²⁹ <http://www.editeur.org/>.

³⁰ <http://www.bisg.org/>.

³¹ <http://www.bic.org.uk/>.

³² <http://www.bisg.org/what-we-do-21-15-onix-for-books.php#FAQ>.

³³ http://www.bisg.org/docs/Best_Practices_Document.pdf.

simplificar su utilización por parte de los usuarios, reduce a 31 estos elementos fundamentales:

1. ISBN-10/ISBN-13/EAN.UCC-13
2. Title
3. Contributor(s)
4. Publisher/Imprint/Brand Name
5. Price
6. Publisher's Proprietary Discount Code
7. Publisher Status Code
8. Product Availability Code
9. Product Form (Format/Binding/Packaging)
10. Publication Date
11. On Sale Date
12. BISAC Subject
13. Language of Product Content
14. Series
15. Edition Number
16. Edition Type/Description
17. Volume Number
18. ONIX Audience Code
19. Age Range of Target Audience
20. Case Pack/Carton Quantity
21. Replaces/Replaced By
22. Territorial Rights
23. Bar Code Indicator
24. Weight and Dimensions
25. Return Code
26. Page Count, Running Time, and Extent
27. Distributor/Vendor of Record
28. Number of Pieces
29. Textual Description of Product
30. Illustration Details
31. Digital Image of Product

Se puede ver que, aunque similares a los que encontramos en los modelos que ya conocemos, ONIX está planteado con elementos propios del ámbito comercial.

Hay que señalar que ONIX en realidad es una familia de estándares del cual quizá *ONIX for Books* sea el más conocido pero no el único. Existen también *ONIX for Serials* y *ONIX for Publications Licenses* así como otros modelos más especializados para metadatos asociados con el registro de identificadores (DOIs, ISTCs, etc.) y para la comunicación de derechos.

Como se puede ver en la página web del BISG³⁴, ONIX tiene una importante comunidad de usuarios. Las principales librerías en línea como Amazon, BN.com o Borders al igual que grandes editoriales como Ingram o RR. Bowker han implementado ONIX por lo que su mantenimiento parece de momento asegurado.

1.6.4. Metadata Object Description Schema (MODS)

MODS es un modelo de metadatos para la descripción de objetos que fue desarrollado por la Biblioteca del Congreso de EE.UU. a partir de 2002. Es un esquema XML que pretende utilizar datos de registros MARC 21 ya existentes así como permitir la creación de registros nuevos. Incluye un subconjunto de campos MARC y utiliza etiquetas lingüísticas en lugar de numéricas. Lo mantiene el MODS Editorial Committee³⁵ con el apoyo del Network Development and MARC Standards Office³⁶ de la Biblioteca del Congreso.

- Los posibles usos de MODS no difieren de los que podrían tener otros modelos de metadatos:

³⁴ <http://www.bisg.org/directory/onix-users.php?type=sender>.

³⁵ <http://www.loc.gov/standards/mods/editorial-committee.html>.

³⁶ <http://www.loc.gov/marc/ndmso.html>.

- Como un formato de SRU (es un protocolo de comunicación para el envío de registros).
- Como una extensión del esquema METS.
- Para representar metadatos que vayan a ser recolectados (en este caso mediante el protocolo OAI-PMH, que mencionaremos brevemente más adelante).
- Como modelo de descripción en XML.
- Para representar un registro MARC simplificado en XML.

Las ventajas³⁷ que plantean los promotores de MODS se centran fundamentalmente en instituciones que partan de registros MARC. Algunas ventajas que señalan son:

- El conjunto de elementos es más rico que el de Dublin Core.
- El conjunto de elementos es más compatible con datos bibliotecarios que ONIX.
- El esquema está más orientado al usuario que el esquema MARCXML.
- El conjunto de elementos es más sencillo que formato MARC completo.

En cuanto a sus características principales, cabe señalar que los elementos de MODS en general heredan la semántica de MARC, aunque en ocasiones varios campos MARC se convierten en un único elemento de MODS. Es decir, aunque MODS pretende ser un modelo de metadatos basado en MARC, lleva a cabo una síntesis de algunos campos MARC por lo que su correspondencia no es exacta. Si convertimos un registro MARC en MODS y luego tratamos de volver a MODS no tendremos exactamente el mismo registro, habrá campos que hayan sido convertidos a otros más generales con lo que se habrá perdido cierta especificidad.

³⁷ <http://www.loc.gov/standards/mods/mods-overview.html>.

Existe una hoja de estilo ya creada para convertir MARCXML a MODS³⁸. Igualmente es posible consultar las correspondencias que se establecen en esa hoja de estilo³⁹. A continuación vemos un ejemplo de cómo se han establecido para el título. A la izquierda están los campos y subcampos MARC (incluso en algún caso se tienen en cuenta los indicadores) y a la derecha el elemento correspondiente en MODS:

3. Mapping for MARC 21 bibliographic records

| | |
|---|--|
| titleInfo | <titleInfo> |
| 245 \$a\$f\$g\$sk | <title> with no <titleInfo> type attribute and |
| 245 \$b | <subTitle> |
| 245 \$n (and \$f\$g\$sk following \$n) | <partNumber> |
| 245 \$p (and \$f\$g\$sk following \$p) | <partName> |
| 245 ind2 is not 0 | <nonSort> around characters excluded from sort as indicated in indicator value |
| <i>[If \$f\$g\$sk follow \$b they go with <subTitle>. If they follow \$a they go with <title>.]</i> | |
| 210 \$a | <title> with <titleInfo> type="abbreviated" and |
| 210 \$b | <subTitle> |
| 210 \$2 | add attribute authority="content of subfield" |
| 242 \$a | <title> with <titleInfo> type="translated" and |
| 242 \$b | <subTitle> |
| 242 \$n | <partNumber> |
| 242 \$p | <partName> |
| 242 \$y | add attribute lang="content of subfield" |
| 246 \$a with ind2=1 | <title> with <titleInfo> type="translated" and |
| 246 \$b | <subTitle> |
| 246 \$i | displayLabel="text of \$i" |
| 246 \$n | <partNumber> |
| 246 \$p | <partName> |

Ilustración 4. Mapeo Marc21-MODS. Biblioteca del Congreso

³⁸ Disponible en <http://www.loc.gov/standards/marcxml/>. La última versión es MARCXML to MODS Stylesheet (Version 3.4): <http://www.loc.gov/standards/mods/v3/MARC21slim2MODS3-4.xsl>.

³⁹ <http://www.loc.gov/standards/mods/mods-mapping.html>.

Un último recurso que conviene citar al hablar de MODS es la guía de uso ofrecida por la Biblioteca del Congreso⁴⁰. En ella ofrece toda la información sobre MODS. Para ver todos los elementos que componen este vocabulario contamos con el punto 2 de la Guía *MODS elements and attributes*⁴¹. Desde ahí es posible acceder a la definición de cada elemento. En esta definición se incluye, la definición, los atributos del elemento, y los subelementos que lo componen, si los hay. Por ejemplo:

Top-level Element: <titleInfo>

| | |
|--------------------|---|
| Element | <titleInfo> |
| Definition | A word, phrase, character, or group of characters, normally appearing in a resource, that names it or the work contained in it. |
| Attributes | type; authority; authorityURI, valueURI; usage; displayLabel; supplied; nameTitleGroup; altRepGroup; xlink; ID; lang; xml:lang; script; transliteration |
| Subelements | <title> <subTitle> <partNumber> <partName> <nonSort> |

Ilustración 5. Ejemplo elemento MODS. Biblioteca del Congreso

A continuación se explican las directrices de uso, se definen los atributos y cada uno de los subelementos.

También es posible acceder a la página donde se recogen ordenados alfabéticamente todos los elementos que componen MODS⁴².

Al igual que MODS pretende ser una adaptación del formato MARC de registros bibliográficos, la Biblioteca del Congreso quiso hacer lo mismo para los registros de autoridades. Esto dio lugar a **MADS, Metadata Authority Description Standard**⁴³.

⁴⁰ <http://www.loc.gov/standards/mods/userguide/>.

⁴¹ <http://www.loc.gov/standards/mods/userguide/generalapp.html>.

⁴² <http://www.loc.gov/standards/mods/userguide/userguide-index.html>.

⁴³ <http://www.loc.gov/standards/mads/>.

1.6.5. Metadata for Images in XML Schema. MIX.

El modelo de metadatos que vamos a ver a continuación, MIX, está pensado para la definición de imágenes digitales ráster, con el fin de permitir a los usuarios desarrollar, intercambiar e interpretar archivos de imágenes digitales. Está hecho con el fin de facilitar la interoperabilidad entre sistemas, servicios y software así como para apoyar la preservación a largo plazo y el acceso permanente a las colecciones de imágenes digitales. Este modelo se centra muy específicamente en la imagen digital y deja otros aspectos de la preservación a otros modelos como PREMIS.

Al igual que ocurría con PREMIS, MIX no es originalmente un esquema XML sino un Diccionario de datos⁴⁴ donde se definen cuáles es necesario consignar para la gestión de imágenes digitales. El propio Diccionario, por otra parte, está concebido y recomienda codificar estos datos en XML. El Diccionario es, además, la norma ANSI/NISO Z39.87-2006. La información sobre el estándar se puede encontrar en la página de la Biblioteca del Congreso⁴⁵, donde también se puede acceder al esquema MIX en XML⁴⁶.

En el Diccionario se emplean dos tipos de elementos: contenedores de datos y elementos de datos. Los contenedores de datos son agrupaciones semánticas de dos o más elementos de datos relacionados, contenedores o sub-contenedores. Los elementos de datos son la parte componente del Diccionario de datos y se utilizan para registrar los valores específicos de los datos.

El Diccionario describe elementos para definir cinco tipos de informaciones: las relativas al objeto digital; a la imagen digital; a la captura de la imagen; al aseguramiento de la calidad de la imagen; y la relativa a la historia de los cambios que se producen. A

⁴⁴ http://www.niso.org/kst/reports/standards/kfile_download?id%3Austring%3Aiso-8859-1=Z39-87-2006.pdf&pt=RkGKiXzW643YeUaYUqZ1BFwDhIG4-24RJbcZBWg8uE4vWdpZsJDs4RjLz0t90_d5_ymGsj_IKVaGZww13HuDIISn6cvwjex0ejiIKSaTYIErPbfamndQa6zKS6rLL3olr.

⁴⁵ <http://www.loc.gov/standards/mix/>.

⁴⁶ <http://www.loc.gov/standards/mix/mix20/mix20.xsd>.

continuación voy a mencionar brevemente qué recoge cada uno de estos bloques y he pegado los metadatos que se incluyen en cada uno pero os recomiendo que leáis con detalle la definición de cada uno de los elementos que encontraréis en el Diccionario.

Metadatos relativos al objeto digital:

Aquí se agrupan una serie de elementos que se refieren a cualquier objeto digital, no sólo a imágenes digitales. Por lo tanto son datos de carácter más general referidos a la preservación, no tan específicamente técnicos como los referidos a la imagen. Los datos de esta sección se han armonizado para que sean compatibles con PREMIS.

Los elementos que se incluyen en esta sección son los siguientes:

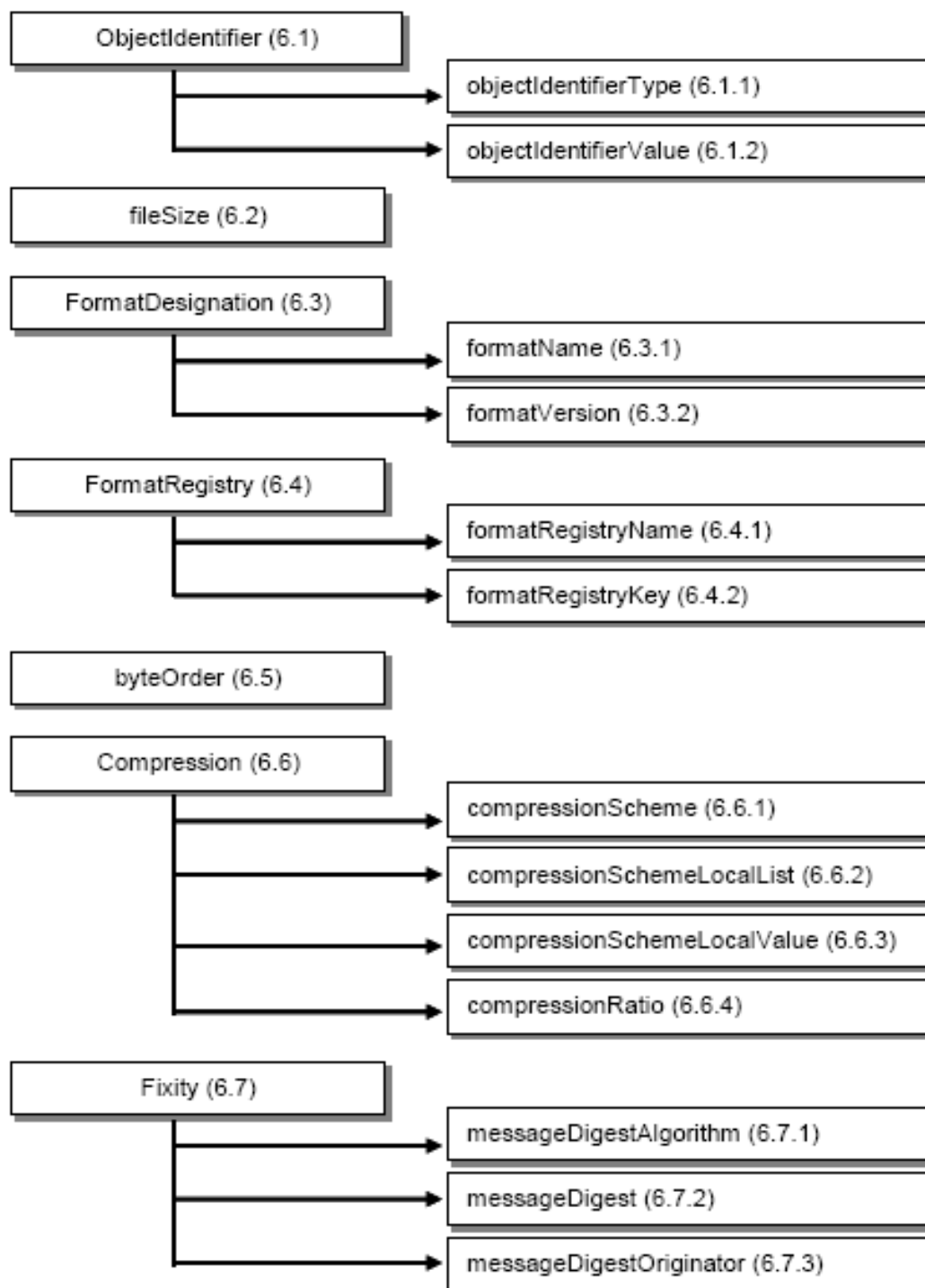


Ilustración 6. Metadatos MIX relativos al objeto digital. Biblioteca del Congreso

Metadatos relativos a la imagen digital:

Aquí se incluyen los metadatos que se consideran imprescindibles para la reconstrucción de la imagen digital, es decir, que se pueda visualizar en un interfaz. El Diccionario describe detalladamente cada uno de los elementos. Aquí tenéis la visión general de cuáles se incluyen:

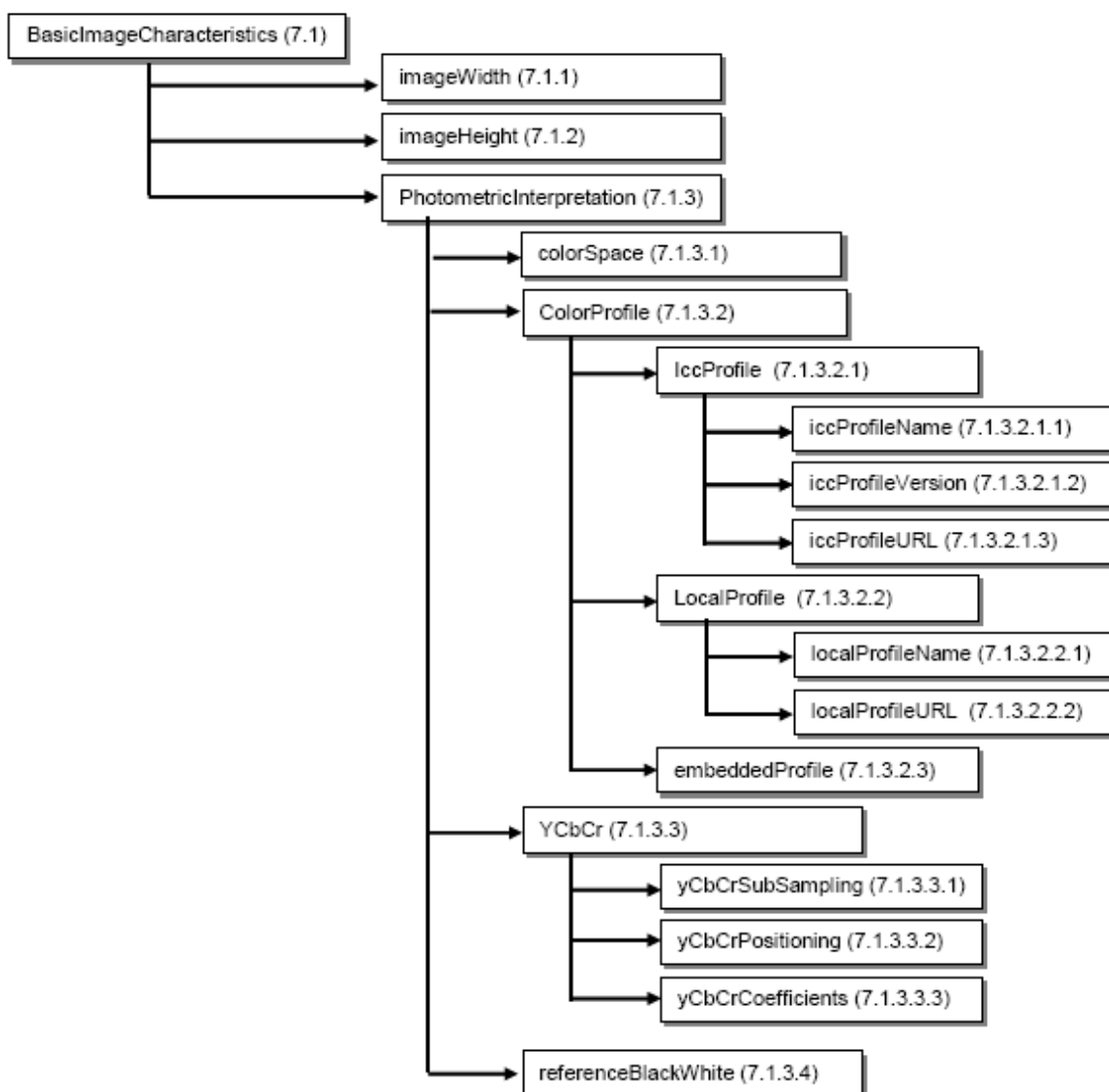


Figure 2: Logical structure of BasicImageCharacteristics

Ilustración 7. Metadatos MIX relativos al objeto digital. Biblioteca del Congreso

Metadatos relativos a la captura de la imagen:

Esta sección podría describirse como la de los metadatos técnico-descriptivos o administrativos. Parte de la información se puede obtener del propio archivo de la imagen y otra parte deberá ser aportada por la propia institución responsable de la captura de imagen. Estos metadatos sirven para documentar los atributos necesarios del proceso de transformación de analógico a digital para asegurar en el futuro la calidad de la imagen. Por definición, la captura de la imagen sólo se da una vez por lo que aquí se pueden aportar datos clave sobre las condiciones en que dicha captura tuvo lugar.

Los elementos incluidos son:

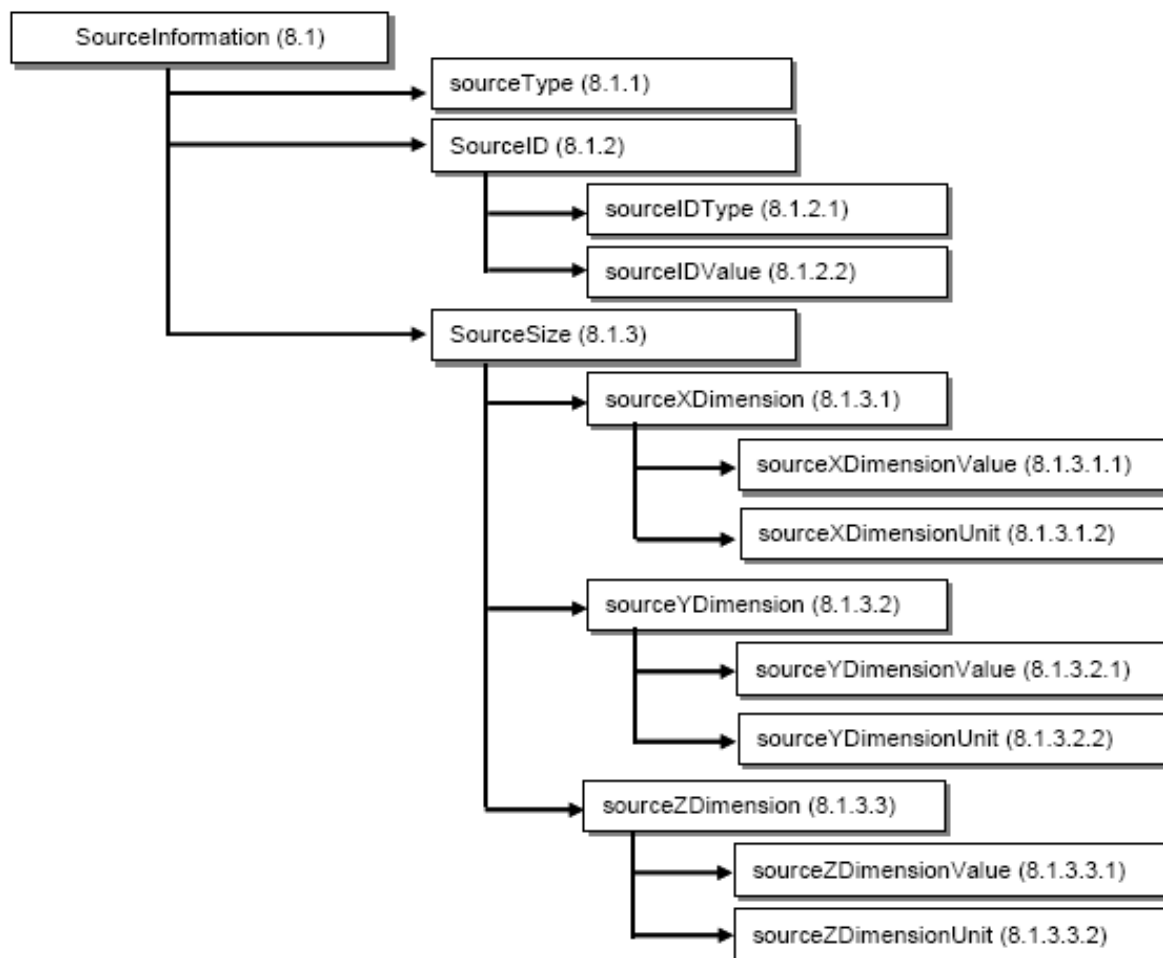


Figure 4: Logical structure of SourceInformation

Ilustración 8. Metadatos MIX relativos a la captura de la imagen. Biblioteca del Congreso

Metadatos relativos al aseguramiento de la calidad:

El principal objetivo de esta sección es mantener los atributos de la imagen inherentes a su calidad. Estos datos tienen un uso en el presente y en el futuro. En el presente permiten medir la exactitud del resultado de digitalización y en el futuro, el de las técnicas de preservación, particularmente la migración.

La figura que resumen estos datos es la siguiente:

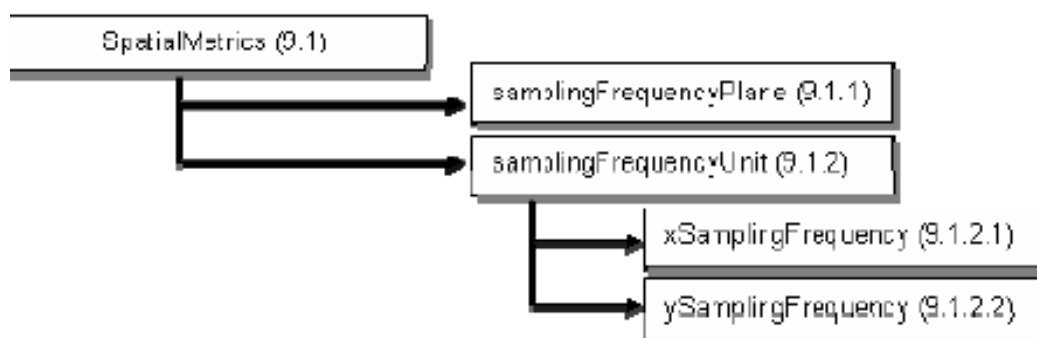


Ilustración 9. Metadatos MIX. relativos al aseguramiento de la calidad.

Metadatos relativos a la historia de los cambios:

Este conjunto de datos están diseñados para documentar los procesos que se aplican sobre la imagen. Estos procesos pueden ser edición o transformación (migración) de la imagen. A continuación, el esquema de aplicación de estos metadatos y el conjunto de ellos:

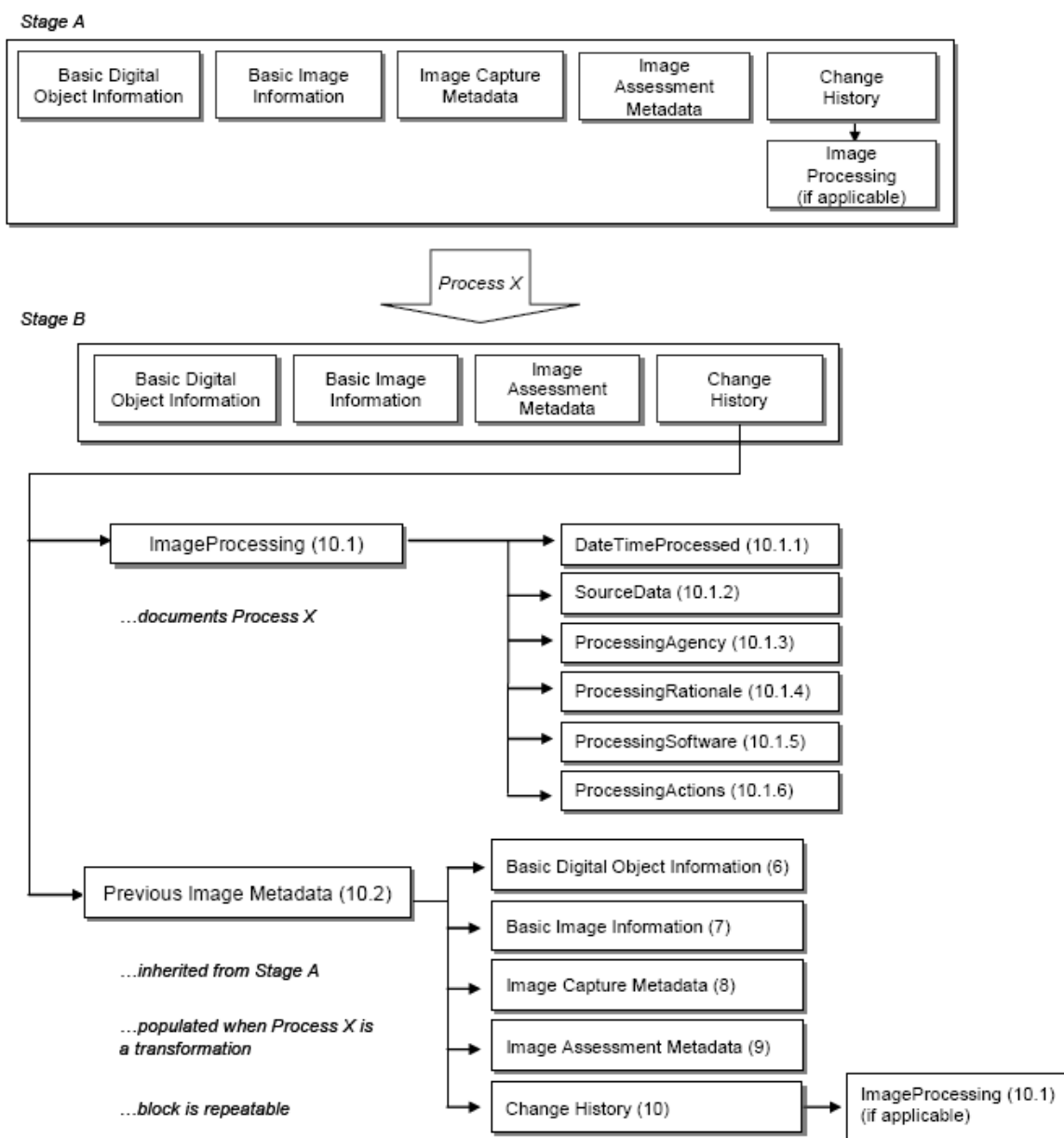


Figure 13: Logical structure of change history for transformed image file

Ilustración 10. Metadatos relativos a la historia de los cambios. Biblioteca del Congreso

Como habéis podido comprobar, MIX, es un detallado conjunto de elementos pensado para recoger toda la información necesaria para la correcta gestión de las imágenes digitalizadas. Al ser un modelo pensado para expresarse en XML y tener en cuenta otros esquemas como PREMIS, es fácilmente compatible con el modelo de metadatos que decidamos aplicar a nuestra biblioteca digital.

1.6.6. Otros estándares de la Biblioteca del Congreso.

Como habéis podido ver hasta ahora, la Biblioteca del Congreso de Estados Unidos tiene un papel principal en el desarrollo y mantenimiento de diferentes modelos de metadatos. En este curso no vamos a entrar a ver con detalle más ningún otro modelo. El objetivo es que con lo que habéis visto estéis en disposición de conocer, evaluar y decidir qué metadatos emplearíais en vuestra biblioteca. Esta decisión, el fin último de todo este esfuerzo, dependerá en todo caso de las condiciones particulares (colección, recursos económicos, personal...) para la que trabajéis.

La Biblioteca del Congreso mantiene la página www.loc.gov/standards/ en la que da acceso a la información de todas las normas que mantiene. Aparte de las que hemos visto, MARC, MODS, METS, MIX, PREMIS, hay otra serie de estándares que mantiene y sería conveniente que conocierais:

EAD (Encoded Archival Description)⁴⁷: Es una norma, también basada en XML, para la descripción de materiales de archivo. El desarrollo de la DTD de EAD comenzó en 1993 en un proyecto de la Universidad de Berkeley, California. El objetivo era crear un formato de codificación no propietario para la descripción de materiales como inventarios, registros, índices y otros documentos creados por archivos, bibliotecas, museos, etc.

VRA Core⁴⁸: Esta norma tiene como fin la descripción de obras de la cultura visual así como de las imágenes que la documentan.

TextMD (Technical Metadata for Text)⁴⁹: Se trata de un esquema XML que detalla los metadatos técnicos para la descripción de objetos digitales textuales.

⁴⁷ <http://www.loc.gov/ead/>.

⁴⁸ <http://www.loc.gov/standards/vracore/>.

⁴⁹ <http://www.loc.gov/standards/textMD/>.

ALTO (Analyzed Layout and Text Objec)⁵⁰: Este esquema XML está pensado para ofrecer metadatos que describan la presentación y el contenido de recursos físicos textuales, como las páginas de un libro o un periódico. Generalmente se utiliza como una extensión de METS aunque las instancias de ALTO pueden utilizarse independientemente.

audioMD y **videoMD**⁵¹: Estos son dos esquemas XML donde se ofrecen metadatos técnicos para la descripción de objetos digitales basados en vídeo y en audio. También se utilizan frecuentemente como una extensión de METS y de PREMIS, aunque igualmente pueden utilizarse de forma independiente.

⁵⁰ <http://www.loc.gov/standards/alto/>.

⁵¹ <http://www.loc.gov/standards/amdvmd/index.html>.

2. EL PROTOCOLO OAI-PMH (OPEN ARCHIVE INITIATIVE: PROTOCOL FOR METADATA HARVESTING)

Aunque este es un curso sobre metadatos, incluimos este pequeño apartado sobre OAI-PMH porque se trata de un protocolo de comunicación muy extendido para compartir esos metadatos. Un protocolo en términos informático, como posiblemente conocéis, es un conjunto de reglas usadas por computadoras para comunicarse unas con otras a través de una red por medio de intercambio de mensajes⁵². En el ámbito bibliotecario, el protocolo más conocido hasta hace unos años era z39.50, que permite la consulta de bases de datos.

(En el módulo 3 vais a oír hablar de OAIS, que es un modelo de preservación y no tiene nada que ver con el protocolo OAI).

Como tal protocolo, OAI establece una serie de reglas para que dos sistemas puedan intercambiar mensajes. En este caso, aparte de otras informaciones, lo más relevante es que permite el intercambio de registros bibliográficos.

Fundamentalmente, el protocolo está pensado para facilitar la “recolección” (traducción de *harvesting*) de metadatos. OAI-PMH permite a dos máquinas intercambiar una serie de mensajes muy sencillos. El primero de ellos es:

Identify: Permite recoger información sobre el servidor: nombre, versión del protocolo, administrador, etc.

ListMetadataFormats: Informa de qué metadatos se emplean para codificar los registros contenidos en el servidor.

⁵² http://es.wikipedia.org/wiki/Protocolo_%28inform%C3%A1tica%29.

ListSets: El servidor destino presenta una lista de los *sets*, colecciones, en las que están organizados los registros (puede no haber más que una colección que contenga todos los registros).

ListIdentifiers: Presenta un listado de los identificadores de todos los registros contenidos en el servidor.

ListRecords: Presenta el conjunto de registros que se solicitan.

GetRecord: Permite recuperar un registro concreto. Para ello es necesario incorporar a la consulta el identificador del registro que se quiere capturar y el formato de metadatos en que se quiere que se presente.

La información técnica sobre el protocolo se puede encontrar en la de la organización que lo mantiene, la Open Archives Initiative⁵³. En la página *Travesía* del Ministerio de Cultura de España hay un tutorial para explicar qué es y cómo funciona el protocolo en el que podréis encontrar más información⁵⁴.

El protocolo OAI-PMH está pensado para funcionar dentro de la Web. Por eso, es posible, utilizar un navegador para hacer consultas OAI-PMH. A continuación tenéis un ejemplo de consulta para cada uno de las acciones que hemos visto que soporta el protocolo. Si seguís los enlaces veréis lo que el servidor OAI-PMH de la BNE responde para cada argumento:

Identify:

<http://bibliotecadigitalhispanica.bne.es/OAI-PUB?verb=Identify>

⁵³ <http://www.openarchives.org/>. Una matización importante respecto al nombre de Iniciativa de Archivos Abiertos, es que se habla de Archivos Abiertos, que no significa que sean necesariamente gratuitos. Es decir, el protocolo OAI-PMH no especifica si el intercambio de información será gratuito o no, simplemente establece un marco tecnológico que permite ese intercambio de forma abierta.

⁵⁴ <http://travesia.mcu.es/portalnbb/jspui/html/10421/1823/intro.htm>.

List metadata Formats:

<http://bibliotecadigitalhispanica.bne.es/OAI-PUB?verb=ListMetadataFormats>

List Sets

<http://bibliotecadigitalhispanica.bne.es/OAI-PUB?verb=ListSets>

List Identifiers:

<http://bibliotecadigitalhispanica.bne.es/OAI-PUB?verb=ListIdentifiers&set=bdh&metadataPrefix=marc21>.

List Records:

<http://bibliotecadigitalhispanica.bne.es/OAI-PUB?verb=ListRecords&set=bdh&metadataPrefix=marc21>

Get Record

<http://bibliotecadigitalhispanica.bne.es/OAI-PUB?verb=GetRecord&identifier=oai:bibliotecadigitalhispanica.bne.es:1865867&metadataPrefix=marc21>

3. CONCLUSIÓN: DE LOS METADATOS A LA WEB SEMÁNTICA

Ya hemos visto un buen número de modelos de metadatos que pueden ser útiles a la hora de gestionar una biblioteca digital. Como decíamos al principio, los metadatos no son, en su esencia nada diferente de lo que las bibliotecas han hecho desde siempre. Sin embargo, es indudable que el mundo digital ha supuesto cambios en la forma en la que describimos los objetos y las perspectivas desde las que lo hacemos, desarrollándose así múltiples modelos de metadatos.

Desde hace pocos años se ha dado un paso más en el empleo de estos modelos de descripción. Este avance añade a la tradicional tarea de describir recursos (empleando el modelo de metadatos que sea) la de crear enlaces entre ellos para poder incrementar la información que se ofrece. La forma en que estos enlaces se crean y los lenguajes que se emplean los vais a ver en el siguiente Módulo.

Este enlazado de datos estructurados (en modelos de metadatos) permite ya el desarrollo de nuevas formas de acceder a la información mucho más “semánticas”, en la medida en que los ordenadores “entienden” mejor la información que contienen. (Decimos que el ordenador “entiende” la información para referirnos al hecho de que se les da la información mucho más estructurada y con nuevas relaciones entre ella). En la medida en que hay mayor comprensión de la información, hablamos de Web Semántica. Entender qué es y cómo funciona la Web Semántica, el enlazado de datos, es relativamente complejo pero es fundamental para comprender uno de los aspectos más innovadores en el terreno bibliotecario. Innovación que, como hemos visto y veréis, está basada en trabajos, conceptos y teorías que están presentes en las bibliotecas desde hace años.

BIBLIOGRAFÍA

BIBLIOTECA DEL CONGRESO. “ALTO” [en línea]. Disponible en <http://www.loc.gov/standards/alto/>. (Consultado: marzo 2014).

BIBLIOTECA DEL CONGRESO. “EAD” [en línea]. Disponible en <http://www.loc.gov/ead/> (Consultado: marzo 2014).

BIBLIOTECA DEL CONGRESO. “MADS” [en línea]. Disponible en <http://www.loc.gov/standards/mads/> (Consultado: marzo 2014).

BIBLIOTECA DEL CONGRESO. “METS: Introducción y tutorial” [en línea]. Disponible en. http://www.loc.gov/standards/mets/METSOverview_spa.html. (Consultado: marzo 2014).

BIBLIOTECA DEL CONGRESO. “MIX” [en línea]. Disponible en <http://www.loc.gov/standards/mix/> (Consultado: marzo 2014).

BIBLIOTECA DEL CONGRESO. “MODS” [en línea]. Disponible en <http://www.loc.gov/standards/mods/mods-overview.html>. (Consultado: marzo 2014).

BIBLIOTECA DEL CONGRESO. “PREMIS” [en línea]. Disponible en. <http://www.loc.gov/standards/premis/>. (Consultado: marzo 2014).

BIBLIOTECA DEL CONGRESO. “PREMIS Data Dictionary for Preservation Metadata” [en línea]. Disponible en www.loc.gov/standards/premis/v2/premis-2-0.pdf.. (Consultado: marzo 2014). Disponible también en español gracias a la traducción de Bárbara Muñoz de Solano y Palacios y Lorea Elduayen Pereda en http://www.loc.gov/standards/premis/PREMIS_es.pdf

BIBLIOTECA DEL CONGRESO. “PREMIS Schemas” [en línea]. Disponible en. <http://www.loc.gov/standards/premis/schemas.html>. (Consultado: marzo 2014).

BIBLIOTECA DEL CONGRESO. “Standars at the Library of Congress” [en línea].

Disponible en www.loc.gov/standards/. (Consultado: marzo 2014).

BIBLIOTECA DEL CONGRESO. “TextMD” [en línea]. Disponible en

<http://www.loc.gov/standards/textMD/>. (Consultado: marzo 2014).

BIBLIOTECA DEL CONGRESO. “VRACORE” [en línea]. Disponible en

<http://www.loc.gov/standards/vracore/>. (Consultado: marzo 2014).

CAPLAN, Priscilla (2009): *Entender Premis*. Traducido por María Luis Martínez Conde

[en línea]. Disponible en <http://www.mcu.es/bibliotecas/docs/MC/PREMIS/Contenido.pdf>

DCMI. “DCMI Metadata Terms” [en línea]. Disponible en

<http://dublincore.org/documents/dcmi-terms/>. (Consultado: marzo 2014).

DCMI. “Library Application Profile” [en línea]. Disponible en

<http://dublincore.org/documents/library-application-profile/>. (Consultado: marzo 2014).

EDITEUR. “ONIX: practical guidance and best practice” en línea]. Disponible en

https://www.bisg.org/docs/BISG_ONIXtraining_handout_final.pdf. (Consultado: marzo 2014).

EÍTO BRUN, Ricardo (2004). *Curso de formación Online. XML: aplicaciones libres para la edición electrónica, bibliotecas, archivos y centros de documentación. Módulo 1: Introducción y Características del lenguaje XML*. SEDIC.

EUROPEANA. “ESE Documentation” [en línea]. Disponible en

<http://pro.europeana.eu/ese-documentation>. (Consultado: marzo 2014).

EUROPEANA. “Europeana Semantic Elements Specification and Guidelines” [en línea].

Disponible en <http://pro.europeana.eu/documents/900548/2eee7beb-b9d8-4532-a089-8e8d6df38ce7>. (Consultado: marzo 2014).

HEERY, Rachel y PATEL, Manjula (2000): "Application profiles: mixing and matching metadata schemas" Disponible en <http://www.ariadne.ac.uk/issue25/app-profiles/> (Consultado: marzo 2014).

MÉNDEZ, Eva y SENSO, José A.: "Introducción a los metadatos: estándares y aplicación" [en línea]. Disponible en: <http://www.sedic.es/autoformacion/metadatos/index.htm>.

MINISTERIO DE EDUCACIÓN, CULTURA Y DEPORTE. [en línea]. Disponible en <http://travesia.mcu.es/portaln/jspui/html/10421/1823/intro.htm>. (Consultado: marzo 2014).

REDIRIS. "Dublin Core en castellano" [en línea]. Disponible en <http://www.rediris.es/search/dces/> (Consultado: marzo 2014).

SEDIC. "Glosario DCMI" [en línea]. Disponible en http://www.sedic.es/glosario_DCMI.pdf. (Consultado: marzo 2014).

SEDIC. "Guía de uso de Dublin Core" [en línea]. Disponible en http://www.sedic.es/usando_dublin_core.pdf. (Consultado: marzo 2014).

W3C. "XML essential" [en línea]. Disponible en <http://www.w3.org/standards/xml/core>. (Consultado: marzo 2014).

WIKIPEDIA. "Espacios de nombre XML" [en línea]. Disponible en http://es.wikipedia.org/wiki/Espacio_de_nombres_XML (Consultado: marzo 2014).

WIKIPEDIA. "Lenguaje de marcado" [en línea]. Disponible en http://es.wikipedia.org/wiki/Lenguaje_de_marcado. (Consultado: marzo 2014).

WIKIPEDIA. "Metadata Types" [en línea]. Disponible en http://en.wikipedia.org/wiki/Metadata#Metadata_types. (Consultado: marzo 2014).

Metadatos en mundo bibliotecario

Abril 2014

WIKIPEDIA. “Metadato” [en línea]. Disponible en <http://es.wikipedia.org/wiki/Metadato>. (Consultado: marzo 2014).

WIKIPEDIA. “Protocolo Informático” [en línea]. Disponible en http://es.wikipedia.org/wiki/Protocolo_%28inform%C3%A1tica%29. (Consultado: marzo 2014).

ÍNDICES

Índice de ilustraciones

| | |
|---|----|
| Ilustración 1. Registro MARC21. Biblioteca Nacional de España..... | 26 |
| Ilustración 2. Elementos de Dublin Core Cualificado. DCMI. | 34 |
| Ilustración 3. Ejemplo DC Cualificado. DCMI..... | 40 |
| Ilustración 4. Mapeo Marc21-MODS. Biblioteca del Congreso | 64 |
| Ilustración 5. Ejemplo elemento MODS. Biblioteca del Congreso | 65 |
| Ilustración 6. Metadatos MIX relativos al objeto digital. Biblioteca del Congreso | 68 |
| Ilustración 7. Metadatos MIX relativos al objeto digital. Biblioteca del Congreso | 69 |
| Ilustración 8. Metadatos MIX relativos a la captura de la imagen. Biblioteca del Congreso | 71 |
| Ilustración 9. Metadatos MIX. relativos al aseguramiento de la calidad. | 72 |
| Ilustración 10. Metadatos relativos a la historia de los cambios. Biblioteca del Congreso | 73 |

Índice de tablas

| | |
|--|----|
| Tabla 1. Elementos de ESE. Europea Semantic Elements Specification... .. | 46 |
| Tabla 2. Elementos de ESE. European Semantic Elements Specification... .. | 48 |
| Tabla 3. Ejemplo de un elemento de ESE. European Semantic Elements Specification... .. | 51 |

La titularidad de los materiales del curso corresponde a su autor:

José Luis Bueren Gómez-Acebo.

Han sido creados para su uso exclusivo de la actividad formativa organizada por SEDIC y, por tanto, su reproducción y difusión sin permiso de los autores y SEDIC vulneraría los derechos de autor.