XML

López, Juan.

jl091505@ingenieria.sanmartin.edu.co

Fundación Universitaria San Martin

*Resumen*— Es un Lenguaje de Etiquetado Extensible muy simple, pero estricto que juega un papel fundamental en el intercambio de una gran variedad de datos. Es un lenguaje muy similar a HTML pero su función principal es describir datos y no mostrarlos como es el caso de permite HTML. XML es un formato que permite la lectura de datos a través de diferentes aplicaciones.

*Índice de Términos—XML*

# XML

Las tecnologías XML son un conjunto de módulos que ofrecen servicios útiles a las demandas más frecuentes por parte de los usuarios. XML sirve para estructurar, almacenar e intercambiar información.

## XSL

Lenguaje Extensible de Hojas de Estilo, cuyo objetivo principal es mostrar cómo debería estar estructurado el contenido, cómo debería ser diseñado el contenido de origen y cómo debería ser paginado en un medio de presentación como puede ser una ventana de un navegador Web o un dispositivo móvil, o un conjunto de páginas de un catálogo, informe o libro.

## XPath

Lenguaje de Rutas XML, es un lenguaje para acceder a partes de un documento XML.

*XPath* identifica partes de un documento XML concreto, como pueden ser sus atributos, elementos, etc

## XLink

Lenguaje de Enlace XML, es un lenguaje que permite insertar elementos en documentos XML para crear enlaces entre recursos XML.

Por su lado, describe un camino estándar para añadir hiperenlaces en un archivo XML. Es decir, es un mecanismo de vinculación a otros documentos XML. Funciona de forma similar a un enlace en una página Web, es decir, funciona como lo haría <a href="">, sólo que a href es un enlace unidireccional. Sin embargo, XLink permite crear vínculos bidireccionales, lo que implica la posibilidad de moverse en dos direcciones. Esto facilita la obtención de información remota como recursos en lugar de simplemente como páginas Web.

## Xpointer

Lenguaje de Direccionamiento XML, es un lenguaje que permite el acceso a la estructura interna de un documento XML, esto es, a sus elementos, atributos y contenido.

Funciona como una sintaxis que apunta a ciertas partes de un documento XML, es como una extensión de XPath. Se utiliza para llegar a ciertas partes de un documento XML. Primero, XLink establece el enlace con el recurso XML y luego es XPointer el que va a un punto específico del documento. Su funcionamiento es muy similar al de los identificadores de fragmentos en un documento HTML ya que se añade al final de una URI y después lo que hace es encontrar el lugar especificado en el documento XML. Al ser XPointer una extensión de XPath, XPointer tiene todas las ventajas de XPath y además permite establecer un rango en un documento XML, es decir, con XPointer es posible establecer un punto final y un punto de inicio, lo que incluye todos los elementos XML dentro de esos dos puntos.

## XQL

## Lenguaje de Consulta XML, es un lenguaje que facilita la extracción de datos desde documentos XML. Ofrece la posibilidad de realizar consultas flexibles para extraer datos de documentos XML en la Web.

## Lenguaje de consultas, se basa en operadores de búsqueda de un modelo de datos para documentos XML que puede realizar consultas en infinidad de tipos de documentos como son documentos estructurados, colecciones de documentos, bases de datos, estructuras DOM, catálogos, etc.

# 2. XML SHEMA

# El propósito del estándar XML Schema es definir la estructura de los documentos XML que estén asignados a tal esquema y los tipos de datos válidos para cada elemento y atributo. En este sentido las posibilidades de control sobre la estructura y los tipos de datos son muy amplias.

Al restringir el contenido de los ficheros XML es posible intercambiar información entre aplicaciones con gran seguridad. Disminuye el trabajo de comprobar la estructura de los ficheros y el tipo de los datos.

XML Schema tiene un enfoque modular que recuerda a la programación orientada a objetos y que facilita la reutilización de código.

Los tipos de datos tienen en XML Schema la función de las clases en la POO. El usuario puede construir tipos de datos a partir de tipos predefinidos, agrupando elementos y atributos de una determinada forma y con mecanismos de extensión parecidos a la herencia. Los tipos de datos se clasifican en función de los elementos y atributos que contienen.

1. Tipos Simples

Tipos simples son aquellos que no tienen ni elementos hijos ni atributos.

Son tipos simples:

* Tipos predefinidos de XML: string, double, boolean, etc.
* List (lista de datos separados por espacios).
* Union (tipo de dato derivado de la unión de tipos predefinidos).

1. Tipos complejos

Son tipos complejos aquellos que tienen elementos hijos y/o atributos.

Pueden tener nombre o ser anónimos. Si tienen nombre pueden ser reutilizados dentro del mismo XML Schema o por otros XML Schemas.

Es posible "mezclar" o combinar elementos y texto.

3. DTD

Una DTD es un documento que define la estructura de un documento XML: los elementos, atributos, entidades, notaciones, etc, que pueden aparecer, el orden y el número de veces que pueden aparecer, cuáles pueden ser hijos de cuáles, etc. El procesador XML utiliza la DTD para verificar si un documento es válido, es decir, si el documento cumple las reglas del DTD.

La DTD que debe utilizar el procesador XML para validar el documento XML se indica mediante la etiqueta DOCTYPE. La DTD puede estar incluida en el propio documento, ser un documento externo o combinarse ambas.

La DTD puede incluirse en el propio documento, con la siguiente sintaxis:

**<!DOCTYPE nombre [**

**... declaraciones ...**

**]>**

La DTD puede estar en un documento externo y, si sólo va a ser utilizada por una única aplicación, la sintaxis es la siguiente:

**<!DOCTYPE nombre SYSTEM "uri">**

Se puede combinar una DTD externa con una DTD interna, con la siguiente sintaxis:

**<!DOCTYPE nombre SYSTEM "uri" [**

**... declaraciones ...**

**]>**

La DTD puede estar en un documento externo y, si va a ser utilizada por varias aplicaciones, la sintaxis es la siguiente:

**<!DOCTYPE nombre PUBLIC "fpi" "uri">**

Se puede combinar una DTD externa con una DTD interna, con la siguiente sintaxis:

**<!DOCTYPE nombre PUBLIC "fpi" "uri" [**

**... declaraciones ...**

**]>**

# referencias

1. <http://www.w3c.es/Divulgacion/GuiasBreves/TecnologiasXML>
2. <http://www.mclibre.org/consultar/xml/lecciones/xml_dtd.html>