

XML en Java

Emmanuel Cruz Hernández
emmanuel_cruzh@ciencias.unam.mx

17 de noviembre de 2020

Contenido

- 1 Introducción a XML
- 2 Estructura
 - Prólogo
 - Cuerpo
- 3 Árbol asociado a XML
- 4 Document Type Definition (DTD)
- 5 Bibliografía

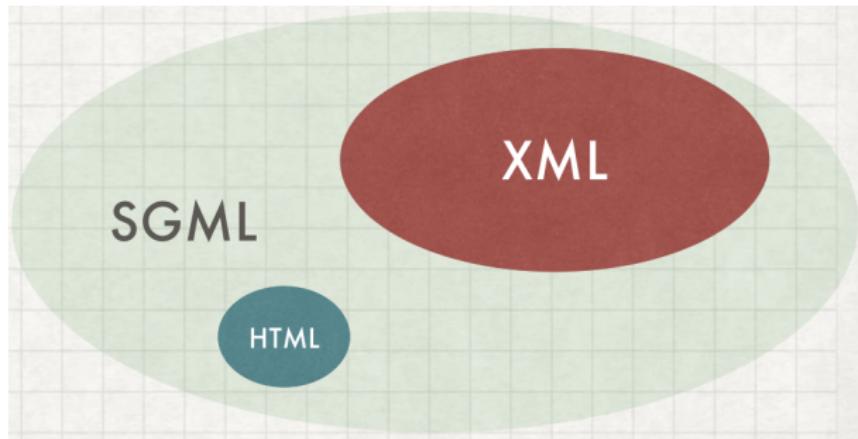
Introducción

XML es un lenguaje de marcado (no es un lenguaje de programación) el cual por medio de etiquetas nos permite marcar, organizar y categorizar información de manera específica.

La sigla XML significa *eXtensible Markup Language*

SGML

Formalmente XML es subconjunto de SGML (Standard Generalized Markup Language) el cuál es un estándar para definir lenguajes de marcados generalizados.



Diferencia entre HTML y XML

A diferencia de HTML, XML no tiene etiquetas predefinidas, siendo ésta una característica importante, pues permite definir etiquetas propias de acuerdo a la estructura que necesitemos modelar.

Ejemplo de archivo XML

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE Biblioteca SYSTEM "biblioteca.dtd" >

<Biblioteca nombre="La astilla de madera">
    <Libro titulo="La Odisea" autor="Homero">
        <Seccion numero="1">Canto 1</Seccion>
        <Seccion numero="2">Canto 2</Seccion>
        <Seccion numero="3">Canto 3</Seccion>
    </Libro>
    <Libro titulo="La Tregua" autor="Mario Benedetti">
        <Seccion numero="1">Lunes 11 de febrero</Seccion>
        <Seccion numero="2">Viernes 15 de febrero</Seccion>
        <Seccion numero="3">Lunes 18 de febrero</Seccion>
        <Seccion numero="4">Martes 19 de febrero</Seccion>
    </Libro>
</Biblioteca>
```

A pesar de que XML no tiene etiquetas predefinidas, tiene una serie de reglas llamadas estándares establecidos por la W3C (<https://www.w3.org/>), algunas de ellas son:

- Todo elemento debe de tener su correspondiente etiqueta de inicio y de cierre o en su defecto una etiqueta vacía.
- Todos los documentos deben tener un elemento al que le llamaremos raíz que contenga a todos los demás.
- Todos los elementos deben estar correctamente anidados.
- Todos los valores de los atributos deben ir entre comillas.

Además, como en cada lenguaje, existen convenciones. Algunas convenciones en XML son las siguientes:

- La primera letra de los nombres es mayúscula.
- Los nombres compuestos están separados por un guión bajo.
- Los elementos deben comenzar por un carácter, o por un espacio, no por números.

Estructura

Un archivo XML se divide en dos partes principales: **prólogo** y **cuerpo**.

Prólogo

Principalmente aquí se especifica la versión xml a usar, la codificación y las declaraciones de tipo la cual es información del documento.

Un ejemplo de prólogo es el siguiente:

```
<?xml version="1.0' encoding="UTF-8' ?>
<!DOCTYPE Biblioteca SYSTEM "biblioteca.dtd' >
```

Prólogo I

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

La primera línea define que el archivo tendrá una estructura XML, define la versión y la codificación del texto en el cuerpo del archivo.

Esta linea define a qué versión de XML se está apagando el archivo que estamos generando. Además de la codificación que usaremos en el documento, evitando cosas como; øæå

Prólogo II

```
<!DOCTYPE Biblioteca SYSTEM "biblioteca.dtd" >
```

La segunda línea corresponde a un archivo que contiene las especificaciones de la estructura de las etiquetas en el cuerpo del archivo. Estás consisten en definir el tipo de archivo que estamos generando. En el archivo con extensión *dtd* se define formalmente los elementos, atributos, entidades y notaciones que podemos utilizar.

Cuerpo

Está compuesto por elementos, los cuales son cadenas de texto (datos) encerradas entre las etiquetas. Los elementos pueden contener atributos. En el cuerpo del archivo xml se representa la información de uno o varios objetos o elementos.

Un elemento es todo (exceptuando el prólogo) lo que está definido dentro del archivo XML (incluyendo las etiquetas que abren y cierran al elemento) un elemento puede contener:

- Texto
- Atributos
- Elementos anidados
- Una mezcla de elementos listados anteriormente

Etiquetas

Las etiquetas, como se ha mencionado con anterioridad, permiten generar la estructura que solucione nuestro problema. Así su función delimita el modelo en la estructura. Siempre que se abra una etiqueta tiene que cerrarse de las siguientes formas:

- `<etiqueta_1> Texto </etiqueta_1>`
- `<etiqueta_1 Texto />`

Etiquetas I

Las etiquetas a su vez pueden contener atributos los cuales son propiedades intrínsecas de qué describe a la etiqueta a la cual está asociada.

Una etiqueta puede tener cero o más atributos designados. Las etiquetas también suelen restringir texto plano el cuál es descrito por nuestra etiqueta.

Atributos

En función de que los atributos son características naturales de cada uno de los elementos, estos atributos no pueden ser multivaluados, es decir, un atributo no puede tener más de un valor.

Atributo vs Texto plano

Ciertamente tenemos más de una manera de representar la información en XML.

The image displays two side-by-side code snippets representing an XML message. Both snippets use the same structure but differ in how they handle date and time information.

Left Snippet:

```
<nota>
  <fecha> 31 de Agosto de 2017 </fecha>
  <de> Manuel </de>
  <para> Chepina </para>
  <mensaje>
    China ¿cómo vas? Hoy hay buenas nuevas, figura que...
  </mensaje>
</nota>
```

Right Snippet:

```
<nota dia="31" mes = "Agosto" anio="2017">
  <de> Manuel </de>
  <para> Chepina </para>
  <mensaje>
    China ¿cómo vas? Hoy hay buenas nuevas, figura que...
  </mensaje>
</nota>
```

Ambas formas transmiten el mismo mensaje, pero tomemos en cuenta que existen restricciones a los atributos.

Comparación

Consideremos al momento de hacer un modelo en XML con la representación de información con atributos y en texto plano:

- No hay restricciones en el texto plano para evitar elementos repetidos.
- Los atributos no pueden anidar más estructuras
- Considerando cambios futuros los atributos no son (fácilmente) extensibles.

Árbol asociado a XML

El archivo XML tiene una estructura de árbol con respecto a sus elementos, aunque sintácticamente tenga estructura de caja anidada.



Document Type Definition

Anteriormente se comentó que un archivo con extensión dtd sirve para especificar las reglas o la estructura que tomará el archivo xml que representa nuestra información.

Declaraciones

Para un DTD tenemos las siguientes declaraciones:

- !ELEMENT nos permite realizar restricciones con respecto a los elementos por medio de las etiquetas
- !ATTLIST Sirve para definir un atributo de una etiqueta

Tipo de atributos

Además, estos atributos tienen etiquetas para saber cómo es el tipo de atributo que pertenece a alguna etiqueta.

- #REQUIRED – Atributos requeridos
- #IMPLIED – Atributos opcionales
- #FIXED – Atributos fijos

Bibliografía

-  M. ZERÓN, C., *Notas de Estructuras de Datos 2018-1*
-  TUTORIAL POINT, *XML, Simply Easy Learning*, 2018