

Lenguajes de Marcas y sistemas de Gestión de la Información

Características de los lenguajes de marcas

ÍNDICE

PRESENTACIÓN

1. **Lenguaje de marcas**
2. Clasificación
3. Agrupación por su ámbito de aplicación
4. Evolución de los sistemas de marcas
 1. GML
 2. SGML
 3. HTML
 4. XML
5. Etiquetas
6. Herramientas de Edición
7. Estructura y sintaxis
8. Documentos XML bien formados
9. Utilización del espacio de nombres

1.- Lenguaje de marcas

- Un Lenguaje de Marcas es un modo de codificar un documento, donde junto con el texto se incorporan: etiquetas, marcas o anotaciones con información adicional relativa a la estructura del texto o su formato de presentación.
- Todo lenguaje de marcas está definido en un documento denominado DTD (Document Type Definition).
- En él se establecen las marcas, los elementos utilizados por dicho lenguaje y sus correspondientes etiquetas y atributos, su sintaxis y normas de uso.

ÍNDICE

PRESENTACIÓN

1. Lenguaje de marcas
2. **Clasificación**
3. Agrupación por su ámbito de aplicación
4. Evolución de los sistemas de marcas
 1. GML
 2. SGML
 3. HTML
 4. XML
5. Etiquetas
6. Herramientas de Edición
7. Estructura y sintaxis
8. Documentos XML bien formados
9. Utilización del espacio de nombres

2.- Clasificación

- De presentación:
 - Define el formato del texto.
- De procedimientos:
 - Orientado también a la presentación pero, en este caso, el programa que representa el documento debe interpretar el código en el mismo orden en que aparece.
- Descriptivo o semántico:
 - Describen las diferentes partes en las que se estructura el documento, pero sin especificar cómo deben representarse.

ÍNDICE

PRESENTACIÓN

1. Lenguaje de marcas
2. Clasificación
3. **Agrupación por su ámbito de aplicación**
4. Evolución de los sistemas de marcas
 1. GML
 2. SGML
 3. HTML
 4. XML
5. Etiquetas
6. Herramientas de Edición
7. Estructura y sintaxis
8. Documentos XML bien formados
9. Utilización del espacio de nombres

3.- Agrupación por su ámbito de aplicación

- Documentación electrónica
- Tecnologías de internet
- Otros lenguajes especializados

3.- Agrupación por su ámbito de aplicación

- Documentación electrónica
 - RTF (Rich Text Format):
 - Formato de Texto Enriquecido
 - Fue desarrollado por Microsoft en 1987.
 - Permite el intercambio de documentos de texto ente distintos procesadores de texto.
 - TeX:
 - Su objetivo es la creación de ecuaciones matemáticas complejas.
 - Wikitexto:
 - Permite la creación de páginas wiki en servidores preparados para soportar este lenguaje.
 - DocBook:
 - Permite generar documentos separando la estructura lógica del documento de su formato.
 - De este modo, dichos documentos, pueden publicarse. en diferentes formatos sin necesidad de realizar modificaciones en el documento original.

3.- Agrupación por su ámbito de aplicación

- Tecnologías de internet
 - HTML, XHTML (Hypertext Markup Language, eXtensible Hype-rtext Markup Language):
 - Su objetivo es la creación de páginas web.
 - RSS:
 - Permite la difusión de contenidos web.

3.- Agrupación por su ámbito de aplicación

- Otros lenguajes especializados
 - MathML (Mathematical Markup Language):
 - Su objetivo es expresar el formalismo matemático de tal modo que pueda ser entendido por distintos sistemas y aplicaciones.
 - VoiceXML (Voice Extended Markup Language):
 - Tiene como objetivo el intercambio de información entre un usuario y una aplicación con capacidad de reconocimiento de habla.
 - MusicXML:
 - Permite el intercambio de partituras entre distintos editores de partituras.

ÍNDICE

PRESENTACIÓN

1. Lenguaje de marcas
2. Clasificación
3. Agrupación por su ámbito de aplicación
4. **Evolución de los sistemas de marcas**
 1. GML
 2. SGML
 3. HTML
 4. XML
5. Etiquetas
6. Herramientas de Edición
7. Estructura y sintaxis
8. Documentos XML bien formados
9. Utilización del espacio de nombres

4.- Evolución de los sistemas de marcas

- En los años 70 surgen unos lenguajes informáticos, distintos de los lenguajes de programación, orientados a la gestión de información.
 - Con el desarrollo de los editores y procesadores de texto surgen los primeros lenguajes informáticos especializados en tareas de descripción y estructuración de información: los lenguajes de marcas.
- Paralelamente, también, surgen otros lenguajes informáticos orientados a la representación, almacenamiento y consulta eficiente de grandes cantidades de datos: lenguajes y sistemas de bases de datos.
- Los lenguajes de marcas surgieron, inicialmente, como lenguajes formados por el conjunto de códigos de formato que los procesadores de texto introducen en los documentos para dirigir el proceso de presentación (impresión) mediante una impresora.

4.- Evolución de los sistemas de marcas

- Ejemplo

- Código de marcas anterior a GML. Las etiquetas son de invención propia.
- Dado el siguiente documento:

```
1 <times 14>
2   <color verde>
3     <centrado>
4       Este texto es un ejemplo para mostrar la utilización primitiva de las marcas
5     </centrado>
6   </color verde>
7 </times 14>
8 <times 10>
9   <color granate>
10    <cursiva>
11      Se utilizan etiquetas de nuestra invención. Las partes importantes del texto pueden resaltarse usando la
12      <negrita>
13        negrita
14      </negrita>
15      , o el
16      <subrayar>
17        subrayado
18      </subrayar>
19    </cursiva>
20  </color granate/>
21 </times 10>
22
```

4.- Evolución de los sistemas de marcas

Este texto es un ejemplo para mostrar la utilización primitiva de
las
marcas

*Se utilizan etiquetas de nuestra invención. Las partes importantes
del texto pueden
resaltarse usando la **negrita** , o el subrayado*

4.- Evolución de los sistemas de marcas

- Posteriormente, se añadieron como medio de presentación a la pantalla.
- Los códigos de estilo de visualización anteriores ya no aparecen, y se emplean otros medios para marcados, distintos de la inclusión a mano de cadenas formateadas.
- Ese proceso se automatiza y basta pulsar una combinación de teclas, o pulsar un botón, para lograr los resultados requeridos.
- Aunque esto es sólo una abstracción, para su uso interno, porque las aplicaciones siguen utilizando marcas para delimitar aquellas partes del texto que tienen un formato especial.

4.1 GML (GENERALIZED MARKUP LANGUAGE)

- Uno de los problemas que se conocen desde hace décadas en la informática es la falta de estandarización en los formatos de información usados por los distintos programas.
- Para resolver este problema, en los años sesenta IBM encargó a Charles F. Goldfab la construcción de un sistema de edición, almacenamiento y búsqueda de documentos legales.
- Tras analizar el funcionamiento de la empresa llegaron a la conclusión de que para realizar un buen procesado informático había que establecer un formato estándar para todos los documentos que se manejaban en la empresa.
- El formato de documentos que se creó como resultado de este trabajo fue GML, cuyo objetivo era describir los documentos de tal modo que el resultado fuese independiente de la plataforma y la aplicación utilizada

4.2 SGML (STANDARD GENERALIZED MARKUP LANGUAGE)

- El formato GML evolucionó hasta que en 1986 dio lugar al estándar ISO 8879 que se denominó SGML.
- Éste era un lenguaje muy complejo y requería de unas herramientas de software caras. Por ello su uso ha quedado relegado a grandes aplicaciones industriales.

4.3 HTML (HYPERTEXT MARKUP LANGUAGE)

- En 1989/90 Tim Berners-Lee creó el World Wide Web y se encontró con la necesidad de organizar, enlazar y compatibilizar gran cantidad de información procedente de diversos sistemas.
- Para resolverlo creó un lenguaje de descripción de documentos llamado HTML, que, en realidad, era una combinación de dos estándares ya existentes:
 - ASCII: Es el formato que cualquier procesador de textos sencillo puede reconocer y almacenar. Por tanto es un formato que permite la transferencia de datos entre diferentes ordenadores.
 - SGML: Lenguaje que permite dar estructura al texto, resaltando los títulos o aplicando diversos formatos al texto.

4.3 HTML (HYPERTEXT MARKUP LANGUAGE)

- HTML es una versión simplificada de SGML, ya que sólo se utilizaban las instrucciones absolutamente imprescindibles.
- Era tan fácil de comprender que rápidamente tuvo gran aceptación logrando lo que no pudo SGML.
- HTML se convirtió en un estándar general para la creación de páginas web.
- Además, tanto las herramientas de software como los navegadores que permiten visualizar páginas HTML son cada vez mejores.

4.3 HTML (HYPERTEXT MARKUP LANGUAGE)

Principales Desventajas:

- No soporta tareas de impresión y diseño.
- El lenguaje no es flexible, ya que las etiquetas son limitadas.
- No permite mostrar contenido dinámico.
- La estructura y el diseño están mezclados en el documento.

4.4 XML (EXTENSIBLE MARKUP LANGUAGE)

- Para resolver estos problemas de HTML el W3C establece, en 1998, el estándar internacional XML, un lenguaje de marcas puramente estructural que no incluye ninguna información relativa al diseño.
- A diferencia de HTML las etiquetas indican el significado de los datos en lugar del formato con el que se van a visualizar los datos.
- XML es un metalenguaje caracterizado por:
 - Permitir definir etiquetas propias.
 - Permitir asignar atributos a las etiquetas.
 - Utilizar un esquema para definir de forma exacta las etiquetas y los atributos.
 - La estructura y el diseño son independientes.

4.4 XML (EXTENSIBLE MARKUP LANGUAGE)

En realidad XML es un conjunto de estándares relacionados entre sí y que son:

- XSL (eXtensible Style Language):
 - Permite definir hojas de estilo para los documentos XML e incluye capacidad para la transformación de documentos.
- XML (Linking Language):
 - Incluye Xpath, Xlink y Xpointer. Determinan aspectos sobre los enlaces entre documentos XML.
- XML Namespaces:
 - Proveen un contexto al que se aplican las marcas de un documento de XML y que sirve para diferenciarlas de otras con idéntico nombre válidas en otros contextos.
- XML Schemas:
 - Permiten definir restricciones que se aplicarán a un documento XML.
 - Actualmente los más usados son las DTD.

Comparativa XML con HTML

XML	HTML
Es un perfil de sgml.	Es una aplicación de sgml.
Especifica cómo deben definirse conjuntos de etiquetas aplicables a un tipo de documento.	Aplica un conjunto limitado de etiquetas sobre un único tipo de documento.
Modelo de hiperenlaces complejo.	Modelo de hiperenlaces simple.
El navegador es una plataforma para el desarrollo de aplicaciones.	El navegador es un visor de páginas.
Fin de la guerra de los navegadores y etiquetas propietarias.	El problema de la 'no compatibilidad' y las diferencias entre navegadores ha alcanzado un punto en el que la solución es difícil.

Comparativa de XML con SGML

XML	SGML
Su uso es sencillo. Facilita el desarrollo de aplicaciones de bajo coste.	Es una aplicación de SGML. Las aplicaciones son muy costosas.
Trabaja con documentos bien formados, o exige que estén validados.	Aplica un conjunto limitado de etiquetas sobre un único tipo de documento.
Es muy utilizado en informática y en más áreas de aplicación.	Sólo se utiliza en sectores muy específicos.
El navegador es una plataforma para el desarrollo de aplicaciones.	El navegador es un visor de páginas.
Compatibilidad e integración con HTML.	No hay una compatibilidad con HTML definida.
No usa etiquetas opcionales.	

ÍNDICE

PRESENTACIÓN

1. Lenguaje de marcas
2. Clasificación
3. Agrupación por su ámbito de aplicación
4. Evolución de los sistemas de marcas
 1. GML
 2. SGML
 3. HTML
 4. XML
- 5. Etiquetas**
6. Herramientas de Edición
7. Estructura y sintaxis
8. Documentos XML bien formados
9. Utilización del espacio de nombres

5.- Etiquetas

- Los lenguajes de marcas utilizan una serie de etiquetas especiales intercaladas en un documento de texto sin formato.
- Dichas etiquetas serán posteriormente interpretadas por los intérpretes del lenguaje y ayudan al procesado del documento.
- Las etiquetas se escriben encerradas entre ángulos, es decir < y >.
- Normalmente, se utilizan dos etiquetas: una de inicio y otra de fin para indicar que ha terminado el efecto que queríamos presentar. La única diferencia entre ambas es que la de cierre lleva una barra inclinada "/" antes del código.
- Las últimas especificaciones emitidas por el W3C indican la necesidad de que vayan escritas siempre en minúsculas para considerar que el documento está correctamente creado.

<etiqueta>texto que sufrirá las consecuencias de la etiqueta</etiqueta>

ÍNDICE

PRESENTACIÓN

1. Lenguaje de marcas
2. Clasificación
3. Agrupación por su ámbito de aplicación
4. Evolución de los sistemas de marcas
 1. GML
 2. SGML
 3. HTML
 4. XML
5. Etiquetas
- 6. Herramientas de Edición**
7. Estructura y sintaxis
8. Documentos XML bien formados
9. Utilización del espacio de nombres

6.- Herramientas de Edición

Para trabajar en XML es necesario editar los documentos y luego procesarlos, por tanto tenemos dos tipos de herramientas:

- **Editores XML:**

- Se basan en la utilización de ficheros de texto plano por lo que basta utilizar un procesador de texto normal y corriente para construir un documento XML.
- Para crear documentos XML complejos e ir añadiendo datos es conveniente usar algún editor XML. Estos nos ayudan a crear estructuras y etiquetas de los elementos usados en los documentos.
- Además algunos incluyen ayuda para la creación de otros elementos como DTD, hojas de estilo CSS o XSL.
- El W3C ha desarrollado un editor de HTML, XHTML, CSS y XML gratuito cuyo nombre es Amaya.

6.- Herramientas de Edición

• Procesadores XML:

- Para interpretar el código XML se puede utilizar cualquier navegador.
- Los procesadores de XML permiten leer los documentos XML y acceder a su contenido y estructura.
- Un procesador es un conjunto de módulos de software entre los que se encuentra un parser o analizador de XML que comprueba que el documento cumple las normas establecidas para que pueda abrirse.
- Estas normas pueden corresponderse con las necesarias para trabajar sólo con documentos de tipo válido o sólo exigir que el documento esté bien formado.
 - Los primeros se conocen como **validadores**
 - Los segundos como **no validadores**.
- El modo en que los procesadores deben leer los datos XML está descrito en la recomendación de XML establecida por W3C.
- Para publicar un documento XML en Internet se utilizan los procesadores XSLT, que permiten generar archivos HTML a partir de documentos XML.

ÍNDICE

PRESENTACIÓN

1. Lenguaje de marcas
2. Clasificación
3. Agrupación por su ámbito de aplicación
4. Evolución de los sistemas de marcas
 1. GML
 2. SGML
 3. HTML
 4. XML
5. Etiquetas
6. Herramientas de Edición
- 7. Estructura y sintaxis**
8. Documentos XML bien formados
9. Utilización del espacio de nombres

7.- XML : Estructura y Sintaxis

- El XML, o Lenguaje de Etiquetas Extendido, es lenguaje de etiquetas, creadas por el programador, que estructuran y guardan de forma ordenada la información.
- No representa datos por sí mismo, solamente organiza la estructura.
- El XML ahorra tiempos de desarrollo y proporciona ventajas, dotando a webs y a aplicaciones de una forma realmente potente de guardar la información.
- Además, se ha convertido en un formato universal que ha sido asimilado por todo tipo de sistemas operativos y dispositivos móviles.

7.- XML : Estructura y Sintaxis

- Al igual que en HTML un documento XML es un documento de texto, en este caso con extensión ".xml", compuesto de parejas de etiquetas, estructuradas en árbol, que describen una función en la organización del documento, que puede editarse con cualquier editor de texto y que es interpretado por los navegadores Web.

7.- XML : Estructura y Sintaxis

Características básicas de XML

- Dado que XML se concibió para trabajar en la Web, es directamente compatible con protocolos que ya funcionan, como HTTP y los URL.
- Todo documento que verifique las reglas de XML está conforme con SGML.
- No se requieren conocimientos de programación para realizar tareas sencillas en XML.
- Los documentos XML son fáciles de crear.
- La difusión de los documentos XML está asegurada ya que cualquier procesador de XML puede leer un documento de XML.
- El marcado de XML es legible para los humanos

7.- XML : Estructura y Sintaxis

Características básicas de XML

- El diseño XML es formal y conciso.
- XML es extensible, adaptable y aplicable a una gran variedad de situaciones.
- XML es orientado a objetos.
- Todo documento XML se compone exclusivamente de datos de marca-do y datos carácter entremezclados.

7.- XML : Estructura y Sintaxis

Etapas del procesado de creación de un documento XML.

- El éxito de cada una de ellas se basa en la calidad de la anterior. Son:
 - Especificación de requisitos.
 - Diseño de etiquetas.
 - Marcado de los documentos.
- El marcado en XML son etiquetas que se añaden a un texto para estructurar el contenido del documento. Esta información extra permite a los ordenadores "interpretar" los textos. El marcado es todo lo que se sitúa entre los caracteres "<" y ">" o "&" y ";"

7.- XML : Estructura y Sintaxis

Etapas del procesado de creación de un documento XML.

- Los documentos XML pueden tener comentarios, que no son interpreta-dos por el interprete XML. Estos se incluyen entre las cadenas "<!--" y "-->", pueden estar en cualquier posición en el documento salvo:
 - Antes del prólogo.
 - Dentro de una etiqueta.
- Los documentos XML pueden estar formados por una parte opcional llamada prólogo y otra parte obligatoria llamada ejemplar.

7.1 El prologo de un documento XML

- Si se incluye, el prólogo debe preceder al ejemplar del documento.
- Su inclusión facilita el procesado de la información del ejemplar y está dividido en dos partes:
 - - **La declaración XML**
 - - **La declaración del tipo de documento:**

7.1 El prologo de un documento XML

La declaración XML

- En el caso de incluirse ha de ser la primera línea del documento, de no ser así se genera un error que impide que el documento sea procesado.
- El hecho de que sea opcional permite el procesamiento de documentos HTML y SGML como si fueran XML, si fuera obligatoria éstos deberían incluir una declaración de versión XML que no tienen.
- El prólogo puede tener tres funciones:
 - Declaración la versión de XML usada para elaborar el documento.
 - Declaración de la codificación empleada para representar los caracteres.
 - La declaración del tipo de documento:

7.1 El prologo de un documento XML

La declaración XML

- Declaración la versión de XML usada para elaborar el documento.

Para ello se utiliza la etiqueta:

<?xml versión= "1.0" ?>

En este caso indica que el documento fue creado para la versión 1.0 de XML.

- **Declaración de la codificación empleada para representar los caracteres.**

Determina el conjunto de caracteres que se utiliza en el documento. Para ello se escribe:

<?xml versión= "1.0" encoding="iso-8859-1" ?>

En este caso se usa el código iso-8859-1 (Latin-1) que permite el uso de acentos o caracteres como la ñ.

7.1 El prologo de un documento XML

La declaración XML

- Los códigos más importantes son:

ESTANDAR ISO	Código del país
UTF-8 (Unicode)	Conjunto de caracteres universal.
ISO -8859-1 (Latin-1)	Europa occidental, Latinoamérica.
ISO -8859-2 (Latin-2)	Europa central y oriental.
ISO -8859-3 (Latin-3)	Suroeste de Europa.

7.1 El prologo de un documento XML

- **La declaración XML**

La declaración del tipo de documento: define qué tipo de documento estamos creando para ser procesado correctamente. Toda declaración de tipo de documento comienza por la cadena:

<!DOCTYPE Nombre_tipo ...>

7.1 El prologo de un documento XML

- **La declaración del tipo de documento:**

Define qué tipo de documento estamos creando para ser procesado correctamente.

Toda declaración de tipo de documento comienza por la cadena:

<!DOCTYPE Nombre_tipo ...>

7.2. EL EJEMPLAR. LOS ELEMENTOS

- Es la parte más importante de un documento XML, ya que contiene los datos reales del documento.
- Está formado por elementos anidados.
- Los elementos son los distintos bloques de información que permiten definir la estructura de un documento XML.
- Están delimitados por una etiqueta de apertura y una etiqueta de cierre.
- A su vez los elementos pueden estar formados por otros elementos y/o por atributos.

7.2. EL EJEMPLAR. LOS ELEMENTOS

- Ejemplo: ejemplo1.xml

```
<?xml version="1.0" encoding="iso-8859-1"?>
<!DOCTYPE libro>
<libro>
  <titulo>XML practico </titulo>
  <autor>Sebastien Lecomte</autor>
  <autor>Thierry Boulanger</autor>
  <editorial>Ediciones Eni</editorial>
  <isbn>978-2-7460-4958-1</isbn>
  <edicion>1</edicion>
  <paginas>347</paginas>
</libro>
```

7.2. EL EJEMPLAR. LOS ELEMENTOS

- En realidad, el ejemplar es el elemento raíz de un documento XML. Todos los datos de un documento XML han de pertenecer a un elemento del mismo.
- Los nombres de las etiquetas han de ser autodescriptivos, lo que facilita el trabajo que se hace con ellas.
- La formación de elementos ha de cumplir ciertas normas para que queden perfectamente definidos y que el documento XML al que pertenecen pueda ser interpretado por los procesadores XML sin generar ningún error fatal.

7.2. EL EJEMPLAR. LOS ELEMENTOS

- **Reglas que se deben cumplir:**

- En todo documento XML debe existir un elemento raíz, y sólo uno.
- Todos los elementos tienen una etiqueta de inicio y otra de cierre. En el caso de que en el documento existan elementos vacíos, se pueden sustituir las etiquetas de inicio y cierre por una de elemento vacío.
- Ésta se construye como la etiqueta de inicio, pero sustituyendo el carácter ">" por "/>". Es decir, <elemento></elemento> puede sustituirse por: <elemento/>
- Al anidar elementos hay que tener en cuenta que no puede cerrarse un elemento que contenga algún otro elemento que aún no se haya cerrado.
- Los nombres de las etiquetas de inicio y de cierre de un mismo elemento han de ser idénticos, respetando las mayúsculas y minúsculas.

7.2. EL EJEMPLAR. LOS ELEMENTOS

- **Reglas que se deben cumplir:**

- Pueden ser cualquier cadena alfanumérica que no contenga espacios y no comience ni por el carácter dos puntos, ":", ni por la cadena "xml" ni ninguna de sus versiones en que se cambien mayúsculas y minúsculas ("XML", "XmL", "xML",...).
- El contenido de los elementos no puede contener la cadena "]]>" por compatibilidad con SGML.
- Además no se pueden utilizar directamente los caracteres mayor que, >, menor que, <, ampersand, &, dobles comillas, ", y apostrofe, '. En el caso de tener que utilizar estos caracteres se sustituyen por las siguientes cadenas:

Carácter	Cadena
>	>
<	<

Carácter	Cadena
&	&
"	"

Carácter	Cadena
'	'

7.2. EL EJEMPLAR. LOS ELEMENTOS

- **Reglas que se deben cumplir:**

- Para utilizar caracteres especiales, como £, ©, ®,... hay que usar las expresiones `&#D;` o `&#H;` donde D y H se corresponden respectivamente con el número decimal o hexadecimal correspondiente al carácter que se quiere representar en el código UNICODE.

Por ejemplo, para incluir el carácter de Euro, €, se usarían las cadenas `€` o `€`

- En el siguiente enlace encontrarás una tabla con los caracteres AS-CII, el nombre HTML, y el número HTML de cada uno de ellos que te será imprescindible a la hora de realizar documentos en HTML y XML.

<http://ascii.cl/es/codigos-html.htm>

7.2.1. LOS ATRIBUTOS

- Permiten añadir propiedades a los elementos de un documento. Los atributos no pueden organizarse en ninguna jerarquía, no pueden contener ningún otro elemento o atributo y no reflejan ninguna estructura lógica.
- No se debe utilizar un atributo para contener información susceptible de ser dividido.

7.2.1. LOS ATRIBUTOS

- Ejemplo: ejemplo2.xml

```
<?xml version="1.0" encoding="iso-8859-1" standalone="yes" ?>
<!DOCTYPE biblioteca>
<biblioteca>
  <ejemplar tipo_ejem="libro" titulo="XML práctico" editorial="Edicion">
    <tipo>
      <libro isbn="978-2-7460-4958-1" edicion="1" pagi-nas="347"></libro>
    </tipo>
    <autor nombre="Sebastien Lecomte"></autor>
    <autor nombre="Thierry Boulanger"></autor>
    <autor nombre="Angel Belinchon Calleja" funcion="traductor"></autor>
    <prestado lector="Pepito Grillo">
      <fecha_pres dia="13" mes="mar" año="2009"></fecha_pres>
      <fecha_devol dia="21" mes="jun" año="2009"></fecha_devol>
    </prestado>
  </ejemplar>
</biblioteca>
```

7.2.1. LOS ATRIBUTOS

Al abrirlo en el Firefox

Este fichero XML no parece tener ninguna información de estilo asociada. Se muestra debajo el árbol del documento.

```
-<biblioteca>
  -<ejemplar tipo Ejem="libro" titulo="XML práctico" editorial="Edicion">
    -<tipo>
      <libro isbn="978-2-7460-4958-1" edicion="1" paginas="347"/>
    </tipo>
    <autor nombre="Sebastien Lecomte"/>
    <autor nombre="Thierry Boulanger"/>
    <autor nombre="Angel Belinchon Calleja" funcion="traductor"/>
  -<prestado lector="Pepito Grillo">
    <fecha_pres dia="13" mes="mar" año="2009"/>
    <fecha_devol dia="21" mes="jun" año="2009"/>
  </prestado>
</ejemplar>
</biblioteca>
```

7.2.1. LOS ATRIBUTOS

Vemos que los elementos aparecen coloreados en ciruela, los nombres de los atributos en negro y sus valores en azul.

- Como se observa en el ejemplo, los atributos se definen y dan valor dentro de una etiqueta de inicio o de elemento vacío, a continuación del nombre del elemento o de la definición de otro atributo siempre separado de ellos por un espacio.
- Los valores del atributo van precedidos de un igual que sigue al nombre del mismo y tienen que definirse entre comillas simples o dobles.
- Los nombres de los atributos han de cumplir las mismas reglas que los de los elementos, y no pueden contener el carácter menor que, <.

ÍNDICE

PRESENTACIÓN

1. Lenguaje de marcas
2. Clasificación
3. Agrupación por su ámbito de aplicación
4. Evolución de los sistemas de marcas
 1. GML
 2. SGML
 3. HTML
 4. XML
5. Etiquetas
6. Herramientas de Edición
7. Estructura y sintaxis
8. **Documentos XML bien formados**
9. Utilización del espacio de nombres

8. DOCUMENTOS XML BIEN FORMADOS

- Todos los documentos XML deben verificar las reglas sintácticas que define la recomendación del W3C para el estándar XML. Esas normas básicas son:
- El documento ha de tener definido un prólogo con la declaración xml completa.
- Existe un único elemento raíz para cada documento: es un solo elemento en el que todos los demás elementos y contenidos se encuentran anidados.
- Hay que cumplir las reglas sintácticas del lenguaje XML para definir los distintos elementos y atributos del documento.

ÍNDICE

PRESENTACIÓN

1. Lenguaje de marcas
2. Clasificación
3. Agrupación por su ámbito de aplicación
4. Evolución de los sistemas de marcas
 1. GML
 2. SGML
 3. HTML
 4. XML
5. Etiquetas
6. Herramientas de Edición
7. Estructura y sintaxis
8. Documentos XML bien formados
9. **Utilización del espacio de nombres**

9. UTILIZACIÓN DE ESPACIOS DE NOMBRE XML

- Permiten definir la pertenencia de los elementos y los atributos de un documento XML al contexto de un vocabulario XML. De este modo se resuelven las ambigüedades que se pueden producir al juntar dos documentos distintos, de dos autores diferentes, que han utilizado el mismo nombre de etiqueta para representar cosas distintas.
- Los espacios de nombres también conocidos como name spaces, permiten dar un nombre único a cada elemento, indexándolos según el nombre del vocabulario adecuado además están asociados a un URI que los identifica de forma única.
- En el documento, las etiquetas ambiguas se sustituyen por otras en las que el nombre del elemento está precedido de un prefijo, que determina el contexto al que pertenece la etiqueta, seguido de dos puntos :.

9. UTILIZACIÓN DE ESPACIOS DE NOMBRE XML

- Esto es:
<prefijo:nombre_etiqueta></prefijo:nombre_etiqueta>
- Esta etiqueta se denomina "nombre cualificado". Al definir el prefijo hay que tener en cuenta que no se pueden utilizar espacios ni caracteres espaciales y que no puede comenzar por un dígito.
- Antes de poder utilizar un prefijo de un espacio de nombres, para resolver la ambigüedad de dos o más etiquetas, es necesario declarar el espacio de nombres, es decir, asociar un índice con el URI asignado al espacio de nombres, mediante un atributo especial xmlns.
- Esto se hace entre el prólogo y el ejemplar de un documento XML y su sintaxis es la siguiente:
<conexion>://<direccionservidor>/<apartado1>/<apartado2>/...

9.- UTILIZACIÓN DE ESPACIOS DE NOMBRE XML

- Ejemplo: ejemplo3.xml
- Sean los documentos XML que organizan la información sobre profesores y alumnos de un Ciclo respectivamente:

```
<?xml version="1.0" encoding="iso-8859-1" standalone="yes"
<!DOCTYPE alumnos>
<alumnos>
  <nombre>Fernando Fernández González</nombre>
  <nombre>Isabel González Fernández</nombre>
  <nombre>Ricardo Martínez López</nombre>
</alumnos>

<?xml version="1.0" encoding="iso-8859-1" standalone="yes" ?>
<!DOCTYPE profesores>
<profesores>
  <nombre>Pilar Ruiz Pérez</nombre>
  <nombre>Tomás Rodríguez Hernández</nombre>
</profesores>
```

9.- UTILIZACIÓN DE ESPACIOS DE NOMBRE XML

- Ejemplo: ejemplo3_b.xml

Al hacer un documento sobre los miembros del Ciclo no se distinguirían los profesores de los alumnos, para resolverlo definiremos un espacio de nombres para cada contexto:

```
<?xml version="1.0" encoding="iso-8859-1" standalone="yes" ?>
<!DOCTYPE miembros>
    DTD está prohibido.
<alumnos xmlns:alumnos="http://DAM/alumnos">
<profesores xmlns:profesores="http://DAM/profesores">
<asistentes>
    <alumnos:nombre>Fernando Fernández González</alumnos:nombre>
    <alumnos:nombre>Isabel González Fernández</alumnos:nombre>
    <alumnos:nombre>Ricardo Martínez López</alumnos:nombre>
    <profesores:nombre>Pilar Ruiz Pérez</profesores:nombre>
    <profesores:nombre>Tomás Rodríguez Hernández</profesores:nombre>
</asistentes>
```