

## Unidad 6

### Conversión y adaptación de documentos XML

XSLT





## La familia de lenguajes de hojas de estilo extensibles: XSL

### •Transformaciones XML: conversión de los datos de un documento XML a:

- Mismo lenguaje, como un subconjunto de los datos originales.
- Otro lenguaje (por ejemplo HTML).
- A otro formato (por ejemplo PDF).





## La familia de lenguajes de hojas de estilo extensibles: XSL

- **XSL: eXtensible Stylesheet Language**, «*lenguajes de hojas de estilo extensible*», en castellano).
- **XSL agrupa las tecnologías de transformación.**
  - XSLT (transformación)
  - XSL-FO (formato)
  - XPath (selección)





# XSLT

- **XSLT se puede transformar un documento expresado en el formato XML en un documento equivalente expresado en otro formato, normalmente HTML.**
- **Proceso de transformación:**
  - Selección (mediante XPath).
  - Transformación.
- **Las transformaciones se almacenan en ficheros XML con extensión .xsl**



# XSLT

**• Los componentes que forman parte de una transformación son los siguientes:**

- Un documento XML de entrada.
- Un fichero de transformación XSLT.
- Un procesador con capacidad para aplicar las transformaciones (un editor o un navegador web).



# XSLT

- **Estructura:**

- El documento XML a transformar debe incluir una referencia a la hoja de transformación.

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

```
<?xml-stylesheet type="text/xsl" href="ejemplo1.xsl"?>
```



# XSLT

- **Estructura:**

- El documento XSL es un documento XML que contiene las reglas de transformación.

```
<?xml version="1.0"?>
<xsl:stylesheet version="1.0"
xmlns:xsl="http://www.w3.org/1999/XSL/
Transform">
<xsl:template match="/">
...
</xsl:template>
</xsl:stylesheet>
```



# XSLT

- **Transformación. El documento de transformación alterna:**
  - Texto libre (normalmente código HTML, se vuelca en la salida sin modificaciones).
  - Instrucciones XSL que seleccionarán partes del documento XML de entrada mediante XPath y transformarán su contenido mediante los elementos de XSLT.





# XSLT


- Ejemplo de documento XSL:

```
<?xml version="1.0"?>
<xsl:stylesheet version="1.0"
xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
<xsl:template match="/">
<html>
<body>
<p><xsl:value-of select="pelicula/titulo"/></p>
</body>
</html>
</xsl:template>
</xsl:stylesheet>
```

# XSLT: Elementos de XSL

Elemento	Descripción
<b>&lt;xsl:value-of&gt;</b>	Obtiene el valor de un elemento o de un atributo del documento XML.
<b>&lt;xsl:for-each&gt;</b>	Recorre un conjunto de elementos.
<b>&lt;xsl:sort&gt;</b>	Ordena un conjunto de elementos.
<b>&lt;xsl:if&gt;</b>	Permite condicionar la transformación en función de una condición.
<b>&lt;xsl:choose&gt;</b>	Permite condicionar la transformación en función de múltiples condiciones.
<b>&lt;xsl:template&gt;</b>	Permite establecer una plantilla y determinar sobre qué conjunto de elementos se va a realizar la transformación.
<b>&lt;xsl:output&gt;</b>	Define el formato de salida del documento.
<b>&lt;xsl:comment&gt;</b>	Delimita un comentario. El contenido no será procesado.

# Ejercicio XSL, paso a paso



*Dado el siguiente XML crear un programa con XSLT que muestre los títulos y los autores de los libros cuya fecha de edición sea posterior al 2000.*


```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="ejercicio1.xsl"?>
  <catalogo>
    <libro fechaedicion="1999">
      <titulo>Don Quijote</titulo>
      <autores>
        <autor>Cervantes</autor>
      </autores>
    </libro>
    <libro fechaedicion="2005">
      <titulo>
        La sociedad civil moderna
      </titulo>
      <autores>
        <autor>Luis Diaz</autor>
        <autor>Pedro Campos</autor>
      </autores>
    </libro>
  </catalogo>
```



*En primer lugar tendremos que crear el fichero ejercicio1.xsl y crear la estructura básica:*

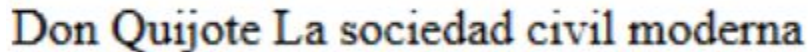
```
<?xml version="1.0" encoding="utf-8"?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
version="1.1">
<xsl:template match="/">

</xsl:template>
</xsl:stylesheet>
```



*Ahora recorramos los libros que hay en el catálogo (recordemos que la estructura es **catálogo/libro**. Simplemente por ver si funciona, de momento el navegador solo muestra los títulos y en una sola línea.*

```
<?xml version="1.0" encoding="utf-8"?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
version="1.1">
<xsl:template match="/">
  <xsl:for-each select="catalogo/libro">
    <xsl:value-of select="titulo" />
  </xsl:for-each>
</xsl:template>
</xsl:stylesheet>
```



Don Quijote La sociedad civil moderna

## Creemos una estructura HTML válida

```
<?xml version="1.0" encoding="utf-8"?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
version="1.1">
<xsl:template match="/">
<html>
  <head>
    <title>Filtrado con XSLT</title>
  </head>
  <body>
    <h1>Filtrado con XSLT</h1>
    <ol>
      <xsl:for-each select="catalogo/libro">
        <li>
          <xsl:value-of select="titulo"/>
        </li>
      </xsl:for-each>
    </ol>
  </body>
</html>
</xsl:template>
</xsl:stylesheet>
```



## Filtrado con XSLT

1. Don Quijote
2. La sociedad civil moderna

*Vamos a procesar solo los libros cuya fecha de edición sea posterior al 2000. Añadamos un if*

```
<?xml version="1.0" encoding="utf-8"?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="1.1">
<xsl:template match="/">
<html>
  <head>
    <title>Filtrado con XSLT</title>
  </head>
  <body>
    <h1>Filtrado con XSLT</h1>
    <ol>
      <xsl:for-each select="catalogo/libro" >

        <xsl:if test="@fechaedicion > 2000" >

          <li>
            <xsl:value-of select="titulo" />
          </li>

        </xsl:if>
      </xsl:for-each>
    </ol>
  </body>
</html>
</xsl:template>
</xsl:stylesheet>
```

## Filtrado con XSLT

1. La sociedad civil moderna



*Ahora para cada libro queremos también mostrar los elementos autor con su propia lista*

```
<?xml version="1.0" encoding="utf-8"?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="1.1">
<xsl:template match="/">
<html>
  <head>
    <title>Filtrado con XSLT</title>
  </head>
  <body>
    <h1>Filtrado con XSLT</h1>
    <ol>
      <xsl:for-each select="catalogo/libro">
        <xsl:if test="@fechaedicion > 2000">
          <li>
            <xsl:value-of select="titulo"/>
          </li>
          <ol>
            <xsl:for-each select="autores/autor">
              <li>
                <!--El elemento actual es .-->
                <xsl:value-of select="."/>
              </li>
            </xsl:for-each>
          </ol>
        </xsl:if>
      </xsl:for-each>
    </ol>
  </body>
</html>
</xsl:template>
</xsl:stylesheet>
```

## Filtrado con XSLT

### 1. La sociedad civil moderna

1. Luis Diaz
2. Pedro Campos




# **Ejercicio: condiciones complejas**





**Supongamos que nos dan el siguiente fichero de inventario:**

```
<inventario>
  <producto codigo="P1">
    <peso unidad="kg">10</peso>
    <nombre>Ordenador</nombre>
    <lugar edificio="B">
      <aula>10</aula>
    </lugar>
  </producto>
  <producto codigo="P2">
    <peso unidad='g'>450</peso>
    <nombre>Altavoz</nombre>
    <lugar edificio="A">
      <aula>6</aula>
    </lugar>
  </producto>
</inventario>
```



Nos dicen que se necesita extraer la información relativa a los productos que pesan más de 5. Una primera aproximación equivocada sería esta:

```
<xsl:template match="/">
  <inventario>
    <xsl:for-each select="inventario/producto" >
      <xsl:if test="peso > 5">
        <nombre>
          <xsl:value-of select="nombre" />
        </nombre>
      </xsl:if>
    </xsl:for-each>
  </inventario>
</xsl:template>
</xsl:stylesheet>
```

**Esta solución está equivocada porque de entrada la pregunta está mal. Si se refieren a 5kg solo debería mostrarse el ordenador y si se refieren a 5g deberían mostrarse ambos.**

Una solución correcta sería esta. Observa como se meten unos if dentro de otros para extraer la información deseada.

```
<xsl:template match="/">
  <inventario>
    <xsl:for-each select="inventario/producto">
      <xsl:if test="./peso/@unidad = 'kg'">
        <xsl:if test="peso > 5">
          <nombre>
            <xsl:value-of select="nombre"/>
          </nombre>
        </xsl:if>
      </xsl:if>
    </xsl:for-each>
    <xsl:if test="peso/@unidad = 'g'">
      <xsl:if test="peso > 5000">
        <nombre>
          <xsl:value-of select="nombre"/>
        </nombre>
      </xsl:if>
    </xsl:if>
  </inventario>
</xsl:template>
</xsl:stylesheet>
```



# Transformación en tabla



Se nos pide convertir el inventario de antes en la tabla siguiente donde el peso debe estar normalizado y aparecer siempre en gramos:

Ordenador	10000
Altavoz	450



```
<xsl:stylesheet
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
<xsl:template match="/">
<html>
  <head><title>Tabla de inventario</title></head>
  <body>
    <table border='1'>
      <xsl:for-each select="inventario/producto">
        <tr>
          <td><xsl:value-of select="nombre"/></td>
          <td>
            <xsl:if test="peso/@unidad='kg'">
              <xsl:value-of select="peso * 1000"/>
            </xsl:if>
            <xsl:if test="peso/@unidad='g'">
              <xsl:value-of select="peso"/>
            </xsl:if>
          </td>
        </tr>
      </xsl:for-each>
    </table>
  </body>
</html>
</xsl:template>
</xsl:stylesheet>
```



# **Ejercicio propuesto 1:**

Transformación de pedidos

Dado el archivo de la moodle pedido.xml.

1. Crear un fichero de estilos que permita mostrar la información de los portátiles en forma de tabla.

## Resultado

Peso	RAM	Disco	Precio
1430	4	500	499
1830	6	1000	1199
1250	2	750	699

# **Ejercicio propuesto 2:**

Transformación de pedidos

Con el mismo fichero de pedidos crear una sola tabla que tenga 3 columnas y aglutine información tanto de portátiles como de tablets:

1. Cuando procesamos portátiles, las columnas serán respectivamente «precio», «ram» y «disco». Solo se procesan portátiles con más de 2GB de RAM.
2. Cuando procesamos tablets, las columnas serán «plataforma», «ram» y «batería». Solo se procesan los tablets con más de 2GB de RAM y que además tengan un tamaño superior a 7 pulgadas.