

₹QUÉ ES XSL?

INTRODUCCIÓN

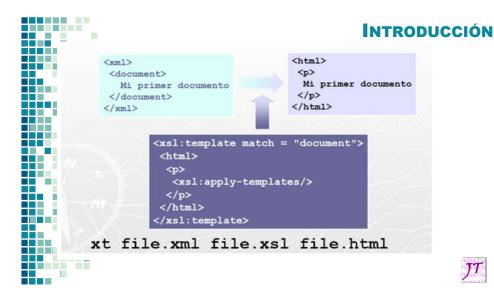
- XSL viene de Extensible Stylesheet Language
- XSL es un lenguaje
- XSL incluye un lenguaje de transformación y un lenguaje de formato

¿PARA QUÉ SIRVE XSL?

- XSL es un lenguaje de transformación
- Por tanto sirve para transformar un XML:
 - En otro XML
 - En un HTML

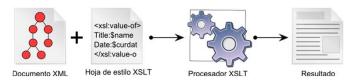


XSLT



La entrada **debe ser un XML**

 XSLT puede trabajar con documentos que satisfagan las reglas de bien formado de XML como XHTML y SGML

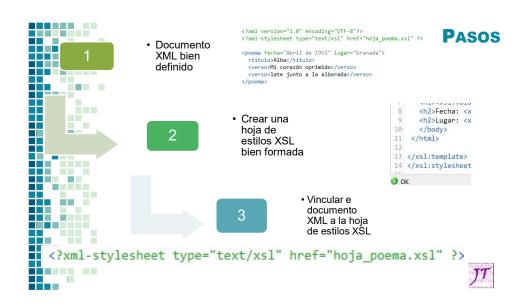


XSLT Uses XPath

XSLT uses XPath to find information in an XML document. XPath is used to navigate through elements and attributes in XML documents.



```
<?xml version="1.0" encoding="UTF-8"?>
                                                                                        EJEMPLO
<?xml-stylesheet type="text/xsl" href="hoja_poema.xsl" ?>
<poema fecha="Abril de 1915" lugar="Granada">
  <titulo>Alba</titulo>
  <verso>Mi corazón oprimido</verso>
  <verso>late junto a la alborada
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform</pre>
                                                                    version="1.0">
<xsl:template match="poema">
 <html>
                                                                     Trabajando con xsl .....
  <head><title>Trabajando con xsl .....</title></head>
   <body>
                                                                     Alba -- (Datos básicos)
  <h1><xsl:value-of select="titulo"/> -- (Datos básicos)</h1>
  <h2>Fecha: <xs1:value-of select="@fecha" /> </h2>
   <h2>Lugar: <xsl:value-of select="@lugar" /> </h2>
                                                                      Fecha: Abril de 1915
   </body>
 </html>
                                                                     Lugar: Granada
                        XSLT Uses XPath
</xsl:template>
                         XSLT uses XPath to find information in an XML document. XPath is used to navigate through elements and attributes in XML documents.
</xsl:stylesheet>
```



```
    Documento 
XML bien

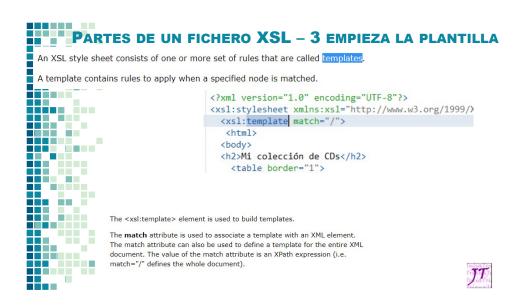
<?xml version="1.0" encoding="UTF-8"?>
                                                                          definido
 <fecha>
                                    Este fichero XML no parece tener
  <dia>25</dia>
  <mes>12</mes>
                                                                                     Crear una
                                                                                     hoja de
estilos XSL
                                      <dia>25</dia>
  <anio>2019</anio>
                                      <mes>12</mes>
<anio>2019</anio>
                                                                                     bien formada
</fecha>
                                     </fecha>

    Vincular e

                                                                                             documento
XML a la hoja
de estilos XSL
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="1.0">
 <xsl:template match="fecha">
   <raiz>Navidad, Navidad, dulce Navidad </raiz>
     <xsl:value-of select="dia" />/
     <xsl:value-of select="mes" />/
     <xsl:value-of select="anio" />
   </hov>
                    <?xml-stylesheet type="text/xsl" href="hoja navidad.xsl" ?>
 </xsl:template>
</xsl:stylesheet>
                                    Navidad, Navidad, dulce Navidad 25/12/2019
JT
```



```
PARTES DE UN FICHERO XSL – 2 SOY UNA HOJA DE ESTILOS
<xsl transform >mlns:xsl="http://
<xsl:template match="/">
                                          <xs[:stylesheet xmlns:xsl="http:/</pre>
        <html>
                                           <xsl:template match="/">
        <body>
                                            <html>
<h2>Mi colección de CDs</h2>
                                           <body>
         <h2>Mi colección de CDs</h2>
                                            Título
                                              Artista
                                               Título
           (/tr>
                                               Artista
      <!-- Se rellena la tabla con los
        <xsl:for-each select="catalogo"</pre>
                                          <!-- Se rellena la tabla con los
        <xsl:for-each select="catalogo"</pre>
        <xsl:value-of select="titul"
<xsl:value-of select="artis
                                           <ts:\value-of select="titul"
                                           <xsl:value-of select="artis"
          (/ysl:for-each)
                                            </xsl:for-each>
</body>
                                            </html>
                                           </body>
       </xsl:template>
                                           </html>
      </xs :transform
                                           </xsl:template>
1/X5
                                          </xsl stylesheet
```





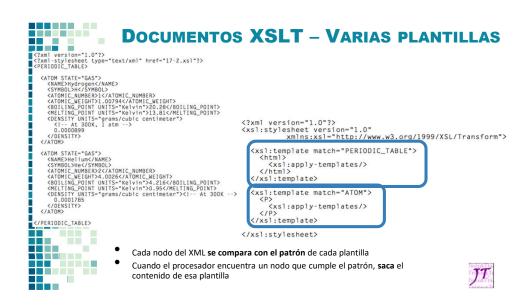


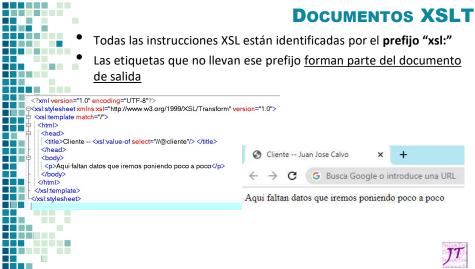
DOCUMENTOS XSLT

- Para describir reglas, plantillas y patrones
- El **elemento raíz** de un XSL es la etiqueta "stylesheet" o "transform"
- Cada plantilla es una etiqueta "xsl:template"
- El **patrón** es el valor del atributo "match" dentro de xsl:template
- La salida está dentro de la etiqueta "xsl:template"

```
<?xml version="1.0" encoding="UTF-8"?>
 <xsl:template match="/">
  <html>
  (head)
   <title>Cliente -- <xsl:value-of select="//@cliente"/> </title>
   </head>
  <body>
   Aqui faltan datos que iremos poniendo poco a poco
  </body>
  </html>
 </xsl:template>
</r>
```





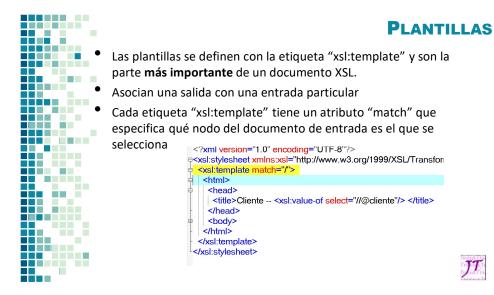


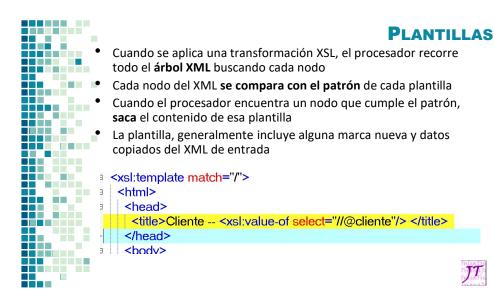
¿DÓNDE OCURRE LA TRANSFORMACIÓN XSL?

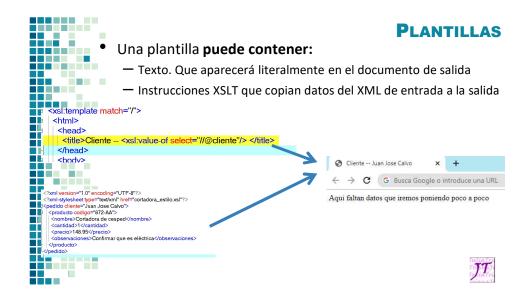
- 1. El documento XML y su XSL asociado se sirven al cliente (navegador web) que transforma el documento como se especifica en la hoja XSL y muestra el resultado al usuario.
- 2. El **servidor aplica la transformación**, genera una salida, generalmente HTML y le envía directamente al cliente el fichero transformado.
- 3. Un tercer programa transforma el XML original (normalmente a HTML) antes de que el fichero se aloje en el servidor. Ambos, servidor y cliente, trabajan siempre con el fichero ya transformado

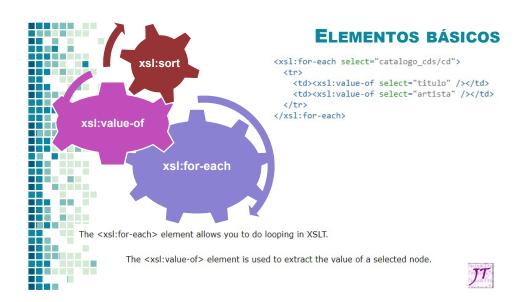


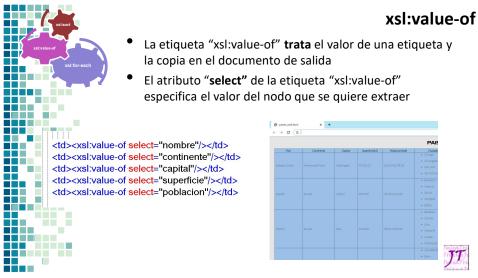
"ENLAZAR" UN XML A UNA XSL Adjuntar un XSL a un XML es fácil. Solo hay que insertar la etiqueta "xml-stylesheet" después de la primera línea del fichero XML Esta instrucción suele tener un atributo "type" con el valor "text/xml" y un atributo "href" cuyo valor es la URL de la hoja XSL <?xml version="1.0" encoding="UTF-8"?> <?xml-stylesheet type="text/xml" href="cortadora estilo.xsl"?> <pedido cliente="Juan Jose Calvo"> codigo="872-AA"> <nombre>Cortadora de cesped</nombre> <cantidad>1</cantidad> cio>148.95</precio> <observaciones>Confirmar que es eléctrica</observaciones> </producto> /pedido>

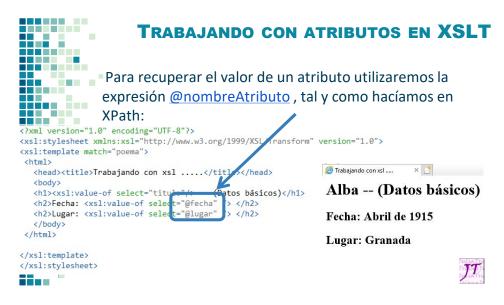








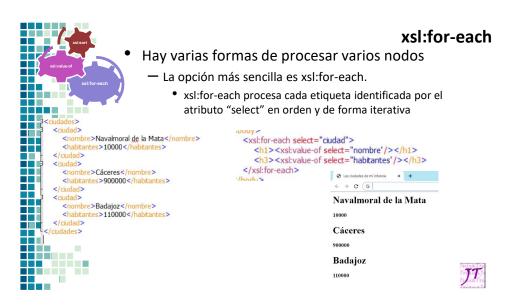








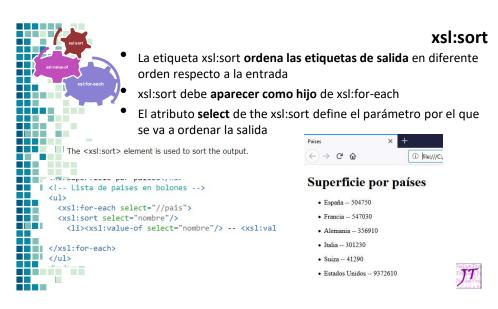


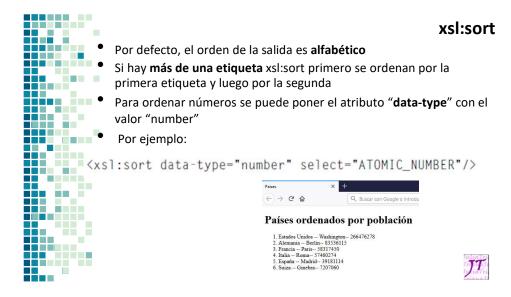














xsl:sort

- También se puede cambiar el orden de ordenación
- Atributo "order". Su valor por defecto es "ascending"

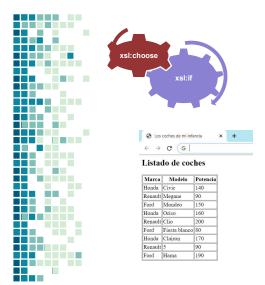








xsl:if



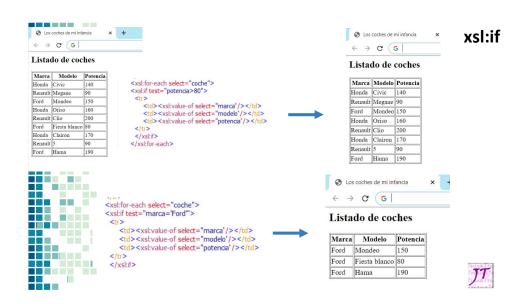
OTROS ELEMENTOS





- Para indicar qué instancias de un elemento queremos procesar, o realizar una "ejecución condicional de código", en XSLT disponemos del elemento xsl:if
- xsl:if va acompañado de un atributo test que contiene una "condición".
- Si la condición se cumple para el elemento que se está procesando, la regla de ejecutará.







xsl:choose, xsl:when y xsl:otherwhise

- Estos elementos "amplían" las posibilidades del elemento xsl:if
- Permiten indicar qué transformación se debe realizar en el caso de que se cumpla una condición.
- Se utilizan de forma conjunta:
 - El elemento xsl:choose contendrá a uno o más elementos xsl:when y a un elemento xsl:otherwise.
- El elemento xsl:when incluye un atributo test que tomará como valor la expresión que se evaluará. Si se cumple, se ejecutará el código escrito entre las etiquetas de inicio y de fin del elemento xsl:when.
- El elemento xsl:otherwise contendrá el código que se ejecutará si no se cumplen las expresiones indicadas en los atributos **test** de los elementos xsl:when.



