
Tecnologías relacionadas con el XML

Ofimática Avanzada

Profesor: Víctor Fresno Fernández
curso 2006/07

XPath

Lenguaje de rutas XML:

- Determina un modelo de información y una sintaxis para señalar partes de un documento XML
- Representa la base para otras especificaciones, como XSLT
- La especificación XPath define dos componentes principales:
 - Sintaxis de expresiones que permite la descripción de rutas hacia partes de un documento XML
 - Conjunto básico de funciones (p.ej. `count()`), conocido como biblioteca central de XPath.

XPath

Contexto y modelo de información:

- XPath considera al documento XML como un árbol
- Los nodos del árbol pueden pertenecer a alguno de los siguientes tipos:
 - ♦ Nodo raíz
 - ♦ Nodo de elemento
 - ♦ Nodo de texto
 - ♦ Nodo de atributo
 - ♦ Nodo de espacio de nombre
 - ♦ Nodo de instrucción de procesamiento
 - ♦ Nodo de comentario

XPath

Rutas de ubicación:

- Expresión XPath que hace referencia a un nodo o grupo de nodos (`/padre/hijo/..`)
- Ejemplo de rutas lógicas:
 - `child::X[@type = "y"]` está compuesto por
 - el eje **child** que especifica la relación entre los nodos seleccionados y el contexto
 - La *prueba de nodo X* que especifica el nombre de los nodos seleccionados
 - El *conjunto opcional* de predicado `[@type = "y"]` refina más aún el conjunto de nodos seleccionados por el predicado

XPointer

Lenguaje identificador de fragmentos XML:

- A diferencia de los marcadores de HTML, en XML se debe proporcionar:
 - Posibilidad de especificar rangos, además de nodos específicos
 - Sintaxis basada en la estructura, de forma que se evite tener que marcar “señalamientos en el destino”
- Especificación construida sobre XPath que añade algunas funciones (range-to, string-range, range, ..., start-point, here, ..)
- Ejemplo:
 - `xpointer (/nodoRaiz/2/6)`
 - `xpointer(nodoRaiz position () lt; 3)`

XLink

XLink permite marcar elementos arbitrarios como elementos de vinculación:

- El objetivo de la especificación XLink es satisfacer la necesidad de crear vínculos XML que tengan significado y que se puedan usar de maneras complejas.
- Para establecer los vínculos, en lugar de imponer un elemento específico, XLink proporciona atributos globales que marquen cualquier elemento como elemento de vinculación .
- El primer paso para habilitar Xlink en un documento XML es definir su espacio de nombre identificado por la URI:

`http://www.w3c.org/1999/xlink`

XLink

Un elemento de vinculación XLink define las relaciones entre los recursos

- Un *recurso* es cualquier cosa que se pueda direccionar en la red
- A los elementos usados para señalar recursos externos se les llama *localizadores*.
- Las relaciones entre recursos se definen mediante *arcos*.
- Todos los elementos de vinculación están marcados por un atributo **xlink:type**.
 - El más importante es el vínculo *extendido*, que es un contenedor de *recursos* (locales), *localizadores* (que apuntan a recursos remotos) o *arcos* (que unen recursos locales o remotos).

XSLT

Lenguaje de transformación de hojas de estilo XML (*XSLT*, XML Stylesheet Language Transformation)

- Las transformaciones XSLT no se expresan en un lenguaje de programación como Java.
- Se especifican en documentos XML llamados hojas de estilo, compuestas por un conjunto de reglas.
- Las reglas (o patrones) están definidas por un patrón y un cuerpo.

```
<xsl:template match="elemento">
```

```
<xsl:text> Hola mundo </xsl:text>
```

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

XSLT

El modelo de procesamiento básico para las hojas de estilo XSLT es:

- Un procesador XSLT lee un documento fuente, recorre nodo tras nodo revisando si existe una plantilla que se pueda aplicar.
- En caso afirmativo, su cuerpo se convierte en un documento.
- El contenido de una plantilla es la combinación de código XML bien formado, que no pertenece al espacio de nombre XSLT, y elementos XSLT que se tratan de forma especial (elementos calificados con significado dentro del espacio de nombre)

<http://www.w3c.org/1999/XSL/Transform>

XSLT

Ejemplo:

```
<xsl:stylesheet          xmlns:xsl="http://www.w3c.org/1999/XSL/Transform"
  version="1.0">
  <xsl:template match="/element1">
    <html><head><title>título que aparecerá</title></head><body>
    ..
    ..
  </xsl:template>
</xsl:stylesheet>
```

- ♦ Los elementos que no forman parte de XSLT simplemente se copian

XSLT

Ejemplo:

- ◆ Para transformar elementos secundarios

```
<xsl:apply-templates select="*" />
```

..

- ◆ Para capturar el valor de atributos

```
<xsl:value-of select="@nombreAtributo" />
```

..

XSLT

Algunos aspectos importantes:

- El elemento raíz de la hoja de estilo incluye el espacio de nombre XSLT.
- Cada plantilla tiene un atributo `match`, cuyo valor es una expresión XPath.
- Cuando un nodo coincide con una plantilla, el elemento `xsl-value-of` solicita el valor de “@nombreAtributo” para cada elemento
- Una vez que el procesador encuentra una plantilla coincidente, debe recibir instrucciones explícitas para que vaya hacia abajo y procese los elementos secundarios a través del elemento `xsl:apply-templates`

