

Direccionamiento, enlaces y formularios en XML

Jose Emilio Labra Gayo

Noviembre 2006

Contenidos

Direcciones relativas e inclusiones

XPointer

XLink

XForms

Direcciones absolutas vs relativas

Las URLs pueden contener:

- ▶ Direcciones absolutas. Por ejemplo:
`http://www.uniovi.es/cursos/xml/alumnos.html`
- ▶ Direcciones relativas. Por ejemplo:
`../notas/alumnos.html` Las direcciones relativas se resuelven respecto al documento actual.

En HTML era posible indicar una dirección de base mediante BASE

```
<HEAD>
  <TITLE>Notas</TITLE>
  <BASE href=" http://www.uniovi.es/cursos/xml/alumnos.html">
</HEAD>
<BODY>
  <A href=" ../notas/notas.html">Notas</A>
</HTML>
```



xml:Base

Permite especificar la URL de base de un documento XML
Las direcciones relativas tomarán dicha URL como base

```
<html xmlns="http://www.w3.org/TR/xhtml1/strict">  
  xml:base="http://www.uniovi.es/cursos">  
  <head><title>Lista de cursos</title></head>  
  <body>  
    <p><a href="cursos.html">Cursos</a></p>  
    <ul>  
      <li><a href="cocina.html">Cocina</li>  
      <li><a href="baile.html">Baile</li>  
    </ul>  
</html>
```

xml:Base

El atributo `xml:base` puede a su vez ser relativo

```
<html xmlns="http://www.w3.org/TR/xhtml1/strict">
  xml:base="http://www.uniovi.es/cursos">
  <head><title>Lista de cursos</title></head>
  <body>
    ...
    <ul xml:base=" ../deportes">
      <li><a href="basket.html">Baloncesto</li>
      <li><a href="gimnasia.html">Gimnasia</li>
    </ul>
  </html>
```

Inclusiones de XML

XInclude (2004) es una recomendación que permite incluir documentos XML dentro de otros documentos
Existía la posibilidad de realizar inclusiones mediante entidades en los DTDs

Se define un elemento `include`

```
<personas xmlns:xi="http://www.w3.org/2001/XInclude">  
  <xi:include href="alumnos.xml" />  
  <xi:include href="profesores.xml" />  
</personas>
```

Atributos de XInclude

- ▶ `href` indica la URI. Si no se incluye, se supone que es , indicando el propio documento
- ▶ `parse` indica si el recurso a incluir es estructurado (`xml`) o textual (`text`)
- ▶ `xpointer` indica la porción del documento a incluir. Sólo se permite cuando `parse="xml"`
- ▶ `encoding` indica tipo de codificación
- ▶ `accept` y `accept-language` permiten realizar negociación de contenido con el protocolo HTTP

```
<xi:include href="profesores.xml"
  accept-language="sp" accept="text/html" />
```

Último recurso (fallback)

`fallback` permite especificar el contenido que se incluirá en caso de que no sea accesible el recurso externo

```
<xi:include href="profesores.xml">
  <xi:fallback>
    No hay profesores
  </xi:fallback>
</xi:include>
```

Si existe el recurso, el contenido de `fallback` se ignora

Último recurso (fallback)

Es posible utilizar varias posibilidades

```
<xi:include href="profesores.xml">
  <xi:fallback>
    <xi:include href="ayudantes.xml">
      <xi:fallback>
        No hay profesores ni ayudantes
      </xi:fallback>
    </xi:include>
  </xi:fallback>
</xi:include>
```

Apuntadores en HTML

HTML incluía un mecanismo para apuntar a partes de una página Web

`http://www.uniovi.es/alumnos.html#pepe` apuntaba al elemento de nombre `pepe` en la página `alumnos.html`

XPointer

XPointer ofrece un mecanismo genérico para señalar partes de un documento XML

Objetivo: ofrecer funcionalidad que permita marcar porciones de documentos XML

Esta funcionalidad fue utilizada previamente e incluso patentada por Sun

XPointer en vínculos

Se identifica mediante `xpointer(expr)` donde `expr` es una expresión XPath

Pueden encadenarse varias expresiones XPointer. Selecciona el contenido de la primera que no falle

Las URIs pueden contener un identificador de fragmento XPointer

De momento no es soportado por los principales navegadores

Suele utilizarse en combinación con XLink

```
alumnos.html#xpointer(//nombre[position()=last()])"
```

Seleccionaría el nombre del último alumno

Apuntadores abreviados

Si no se indica nada, se refiere al elemento con el valor ID del documento

`http://www.uniovi.es/alumnos.html#pepe` se refiere al elemento cuyo valor ID es pepe

Requiere la utilización de DTDs

Secuencias de hijos

Habitualmente se necesita referirse a una secuencia de hijos

Ejemplo: tercer hijo del segundo hijo del elemento raíz

```
xpointer(/child::*[position()=1/  
  child::*[position()=2]/  
  child::*[position()=3])
```

Puede simplificarse como:

```
element(/1/2/3)
```

Espacios de nombre

`xmlns(a=e)` es un mecanismo que permite asociar el alias `a` al espacio de nombres `e`

```
xpointer(svg=http://www.w3.org/2000/svg)
  xpointer(//svg:rect)
```

Selecciona los elementos `rect` del espacio de nombres de SVG

Puntos

En ocasiones, puede interesar seleccionar algo que no sea un elemento.

Por ejemplo, la tercera palabra de un párrafo o el año en 12/06/2004

Los puntos se identifican por su nodo contenidos y un índice (comenzando por 0)

Los nodos de texto tienen un punto inicial 0 y un punto entre cada carácter numerado de forma consecutiva

Ejemplo: `<nombre>Pepe</nombre>` incluye 5 puntos

XPointer añade 2 funciones a XPath

- ▶ `start-point()` selecciona el punto inicial
- ▶ `end-point()` selecciona el punto final

Rangos

Un rango es un intervalo entre 2 puntos. XPointer añade 4 funciones a XPath

- ▶ `range(expr)` devuelve el rango que cubre la expresión XPath `expr`
- ▶ `range-inside(expr)` devuelve el contenido interno de la expresión XPath `expr`
- ▶ `range-to(expr)` devuelve el rango evaluando la expresión respecto al nodo de contexto
- ▶ `string-range(expr, cad, desp, long)` evalúa la `expr` y selecciona todas las cadenas que coincidan con `cad`. Es posible indicar un desplazamiento `desp` y una longitud `long` de las cadenas resultantes
`xpointer(//titulo, "Poema", 6, 4)` selecciona los 4 primeros caracteres que sigan Poema dentro del título

XPointer relativos

Existen 2 funciones que permiten modificar el contexto de la expresión XPather

- ▶ `here()` se refiere al nodo que contiene al XPather
- ▶ `origin()` devuelve la localización del elemento desde el cuál se cargó el documento. Podría ser distinto del documento actual

XLink

XLink define un mecanismo para enlazar documentos entre sí

Permite enlazar 2 o más recursos

Facilita la inclusión de meta-información en los enlaces

Se pueden expresar enlaces que residen en documentos externos

XLink: Enlaces simples

Define una conexión de una sola vía entre 2 recursos

```
<novela xmlns:xlink="http://www.w3.org/1999/xlink"
  xlink:type="simple"
  xlink:href="ftp://archivos.com/novelas/234.tar.gz">
  <titulo>La Regenta</titulo>
  <autor>Clarín</autor>
</novela>
```

XLink: Atributos de Enlaces simples

- ▶ `xlink:type`: tipo del enlace
- ▶ `xlink:href`: URL del recurso
- ▶ `xlink:title`: título del enlace
- ▶ `xlink:role` rol del enlace
- ▶ `xlink:arcrole`: relación entre el recurso que enlaza y el enlazado
- ▶ `xlink:actuate`: define el momento de actuación. Puede ser: `none`, `onLoad`, `onRequest` ó `other`
- ▶ `xlink:show`: Define el comportamiento del enlace en el momento de actuar. Puede ser: `embed`, `new`, `none`, `other`, ó `replace`

Comportamiento del enlace

Mediante los atributos `show` y `actuate` se pueden sugerir acciones al software que utilice los enlaces

```
<imagen xlink:type="simple"
  xlink:actuate="onLoad" xlink:show="embed"
  xlink:href="foto.jpg"
  width="160" height="200" alt="Foto_de_Pepe" />
```

Semántica del enlace

Los atributos `title` y `role` facilitan la inclusión de descripciones sobre los vínculos. El atributo `role` debe ser una URI que puede representar propiedades de web semántica.

```
<novela xlink:type="simple"
  xlink:href="http://www.libros.com#clarin"
  xlink:title="Leopoldo_Alas_Clarin"
  xlink:role="http://purl.org/dc/elements/1.1/creator">
  La Regenta
</novela>
```

Vínculos extendidos

Se indican con el valor del atributo `xlink:type` igual a `extended`'
Pueden contener enlaces de los siguientes tipos

- ▶ Localizadores (`locator`) que permiten localizar recursos
- ▶ Arcos (`arc`) que definen vínculos entre localizadores
- ▶ Recursos (`resource`) definen recursos locales

Vínculos extendidos

```
<novelas xlink:type="extended" xmlns:xlink="...">
  <novela xlink:type="locator" xlink:label="regenta"
    xlink:href="http://www.libros/regenta.html" />
  <novela xlink:type="locator" xlink:label="quijote"
    xlink:href="http://www.libros/quijote.html" />
  <siguiente xlink:type="arc"
    xlink:from="regenta" xlink:to="quijote" />
  <anterior xlink:type="arc"
    xlink:from="quijote" xlink:to="regenta" />
</novelas>
```

Bases de vínculos

XLink permite establecer vínculos entre recursos externos

Un documento que define una lista de vínculos se denomina una base de vínculos (*link base*)

Los navegadores podrían cargar las bases de vínculos y actuar en consecuencia. Esta funcionalidad todavía no está implementada.

Aplicaciones de XLink

- ▶ El soporte para XLink en los navegadores es escaso. Firefox da soporte a enlaces básicos
- ▶ SVG, FO y otras tecnologías de W3c utilizan enlaces básicos de XLink
- ▶ XBRL (*eXtensible Business Reporting Language*) es un estándar basado en XML para definir información financiera. Cada aplicación define una taxonomía, compuesta por documentos XML Schema y bases de vínculos. Varias bases de vínculos predefinidas:
 - ▶ `Presentation` organización jerárquica para visualización
 - ▶ `Label` etiquetas que pueden usarse para mostrar los conceptos
 - ▶ `Reference` referencias a recursos externos
 - ▶ `Calculation` relaciones de cálculo entre conceptos. Ejemplo total es la suma de `suma1` y `suma2`
 - ▶ `Definition` relaciones lógicas entre conceptos

Formularios en HTML

```
<head><title>Busqueda</title></head>
<body>
  <form action="http://www.ejemplo.org/busca"
        method="get">
    Cadena: <input type="text" name="cad" />
           <input type="submit" value="Buscar" />
  </form>
</body>
</html>
```

Mezcla de presentación y contenido

No se realizan chequeo de tipos

Otros problemas: accesibilidad, uso de Javascript, etc.

XForms

XForms (Recomendación W3c en 2006) define un mecanismo para definir formularios separando la presentación del contenido
Facilita el chequeo de tipos mediante uso de XML
Independiente del dispositivo

Ejemplo en XForms

```
<html xmlns="http://www.w3.org/2002/06/xhtml12"
  xml:lang="sp">
<head>
  <title>Busqueda</title>
  <model>
    <submission action="http://www.ejemplo.org/busca"
      method="get" id="busca" />
  </model>
</head>
<body>
  <p>
    <input ref="cad">
      <label>Cadena: </label>
    </input>
    <submit submission="busca">
      <label>Buscar</label>
    </submit>
  </p>
</body>
</html>
```



Fin de la presentación