

XPath forma, junto a **XSLT** y **XSL-FO** una familia de lenguajes llamada **XSL**, diseñados para acceder, transformar y dar formato de salida a documentos XML.

XPath es un lenguaje de expresiones que no se usa de manera independiente, sino que se emplea en el contexto de un lenguaje anfitrión, como XSLT.

Las especificaciones de XPath, desarrolladas por el W3C, se pueden consultar en:

- Lenguaje: <http://www.w3.org/TR/xpath20>
- Modelo de datos: <http://www.w3.org/TR/xpath-datamodel>

Direccionamiento

Las expresiones XPath, al igual que los direccionamientos en un sistema de ficheros, pueden ser absolutas (empiezan por /) o relativas (con respecto a un nodo determinado, llamado nodo de contexto).

XPath trata un documento XML como un árbol de nodos.

Los **tipos de nodos** en XPath son:

- **Raíz del documento** (uno por documento): contiene todo el documento y se representa por el símbolo /. No tiene nodo padre y tiene al menos un nodo hijo, el nodo documento). Contiene también los comentarios e instrucciones de procesamiento, que no forman parte del documento en sí.
- **Elemento**: todo elemento de un documento XML es un nodo de XPath.
- **Atributo**: todo atributo de un documento XML es un nodo de XPath.
- **Texto**: todo contenido textual de un elemento es un nodo de XPath.
- **Comentario**
- **Instrucciones de procesamiento**.
- **Espacios de nombres**.

Los **tipos de datos básicos** son: string, number, boolean y nodo-set (conjunto de nodos).

Las **expresiones más comunes de XPath** son:

Expresión XPath	Coincidencia
elemento	Elemento de nombre elemento
/elemento	Elemento de nombre elemento ubicado en la raíz del documento
e1/e2	Elemento e2 hijo directo del elemento e1
e1//e2	Elemento e2 descendiente (hijo, nieto, ...) del elemento e1
//elemento	Elemento de nombre elemento ubicado en cualquier nivel por debajo de la raíz del documento.
@atributo	Atributo de nombre atributo
*	Cualquier elemento (todos los elementos)
@*	Cualquier atributo (todos los atributos)

.	Nodo actual
..	Nodo padre
espNom:*	Todos los elementos en el espacio de nombres de prefijo espNom
@espNom:*	Todos los atributos en el espacio de nombres de prefijo espNom

Para probar las expresiones XPath, se podrá usar:

- Un entorno de desarrollo como el que ofrece **BaseX**, con ayudas al usuario sobre qué elementos descendientes de cuáles. Se recomienda esta opción.
- Un analizador de expresiones XPath en línea, como por ejemplo:
 - o <http://www.whitebeam.org/library/guide/techNotes/xpath.rhtm>

Como elemento de trabajo vamos a emplear un documento sobre ciclos Formativos y módulos profesionales llamado formacionProfesional.xml.

```
<?xml version="1.0" encoding="iso-8859-1"?>
<fp>
  <ciclos>
    <ciclo siglas="SMR">
      <nombre>Sistemas Microinformáticos y redes</nombre>
      <familiaProfesional>Informática y comunicaciones</familiaProfesional>
      <duracion unidad="horas">2000</duracion>
      <grado>Medio</grado>
      <referenteInternacional clasificacion="CINE">3</referenteInternacional>
    </ciclo>
    <ciclo siglas="ASIR">
      <nombre>Administrador de sistemas informáticos en red</nombre>
      <familiaProfesional>Informática y comunicaciones</familiaProfesional>
      <duracion unidad="horas">2000</duracion>
      <grado>Superior</grado>
      <referenteInternacional clasificacion="CINE">5b</referenteInternacional>
      <ects>120</ects>
    </ciclo>
    ...
  </ciclos>
  <modulos>
    <modulo codigo="0222">
      <nombre>Sistemas operativos monopuesto</nombre>
      <duracion unidad="horas">80</duracion>
      <curso>1</curso>
      <ciclos>
        <ciclo siglas="SMR"/>
      </ciclos>
    </modulo>
    <modulo codigo="0373">
      <nombre>Lenguajes de marcas y sistemas de gestión de información</nombre>
      <duracion unidad="horas">70</duracion>
      <ects>7</ects>
      <curso>1</curso>
      <ciclos>
        <ciclo siglas="ASIR"/>
        <ciclo siglas="DAM"/>
        <ciclo siglas="DAW"/>
      </ciclos>
    </modulo>
    ...
  </modulos>
</fp>
```

Acceso a elementos/atributos vs acceso a su contenido textual

Cuando se escribe una ruta en XPath, por ejemplo a un cierto elemento `/persona/nombre`, lo que devuelve es el conjunto de nodos llamados `nombre`, descendientes de `persona`, descendiente del `nodo raíz`. Si no existe esa ubicación, devuelve un conjunto vacío.

Hay analizadores de expresiones que devuelven directamente el contenido textual del nodo o nodos accedidos.

Ante la duda y para indicar al analizador que lo que se desea es el contenido textual del nodo, se utiliza la función `text()` seguida de la ruta al nodo, de la forma: `/persona/nombre/text()`

Para acceder a un atributo `dni` de persona, se escribirá `/persona/@dni`.

Si se quiere acceder a su valor, se usará la función `string()` de la forma: `/persona/@dni/string()`

Acceso a elementos con rutas simples

Son rutas equivalentes a las que indicaríamos en un **direccionamiento absoluto** en un sistema Linux, es decir, desde la raíz hasta el elemento del cuál queremos extraer la información. La particularidad, respecto a los sistemas de ficheros es que un nodo puede tener varios hijos que se llamen igual, mientras que *en un sistema de ficheros no puede haber dos directorios hijos de un mismo directorio padre que se llame igual*.

Ejemplos:

- Todos los elementos hijos del nodo raíz, es decir, el elemento documento: `/*`
- Todos los atributos de cualquier elemento del documento XML: `//@*`
- Nombre de ciclos formativos: `/fp/ciclos/ciclo/título`
- Duración de los módulos profesionales: `/fp/modulos/modulo/duración`
- Elementos que se llamen nombre del documento: `//nombre`
- Siglas de los ciclos formativos: `/fp/ciclos/ciclo/@siglas`
- Todos los atributos de los módulos: `/fp/modulos/modulo/@*`

Elementos del lenguaje

Operadores booleanos		Operadores aritméticos	
Operador	Codificación alternativa	Operador	Significado
and		+	Suma
or		-	Resta
not		*	Multiplicación
=		div	División
!=		mod	Resto (módulo)
>	>	Otros operadores	
<	<		
>=	>=		
		Operador	Significado

<=	<=		Unión de resultados
----	-------	--	---------------------

Ejemplo: Descendientes de /fp, de nombre ciclo o modulo, que contengan un descendiente directo nombre, cuyo valor se muestra: `/fp//(ciclo | modulo)/nombre`

Funciones

Funciones numéricas		Funciones de cadenas	
Función	Devuelve	Función	Devuelve
round()	Redondeo <code>round(3.14)=3</code>	substring()	Subcadena <code>substring('Beatels',1,4)=Beat</code>
abs()	Valor absoluto <code>abs(-7)=7</code>	starts-with()	Cadena comienza por <code>starts-with('XML','X')=true</code>
floor()	Suelo <code>floor(7.3)=7</code>	contains()	Cadena contiene <code>contains('XML','XM')=true</code>
Ceiling	Techo <code>ceiling(7.3)=8</code>	normalize-space()	Espacios normalizados <code>normalize-space(' The End ')='The End'</code>
		translate()	Cambia caracteres en una cadena <code>translate('The End','The','An')= 'An End'</code>
		string-length()	Longitud de una cadena <code>string-length('Beatles')=7</code>

Ejemplo: Nombre de los ciclos cuyas siglas empiezan por D:

`/fp/ciclos/ciclo[starts-with(@siglas,'D')]/nombre`

Funciones que devuelven elementos por su posición	
Función	Devuelve
position()=n	Elemento que se encuentra en la posición n
elemento[n]	Elemento que se encuentra en la posición n
last()	El último elemento de un conjunto
last()-i	El último elemento -i

Ejemplo:

- Todos los datos del segundo ciclo: `/fp/ciclos/ciclo[position()=2]` o también ... `/fp/ciclos/ciclo[2]`
- Todos los datos del último módulo: `/fp/modulos/modulo[last()]`

Funciones que devuelven nodos

Función	Devuelve
name()	Nombre del nodo actual
root()	El elemento raíz
node()	Nodos descendientes del actual
comment()	Comentarios
processing-instructions()	instrucciones de procesamiento

Ejemplo:

- Devuelve el nombre del nodo actual, es decir, ciclo:
`/fp/ciclos/ciclo/name()`
- Todos los comentarios existentes en el documento:
`//comment()`
- Todas las instrucciones de procesamiento existentes en el documento:
`//processing-instruction()`
- Todos los nodos del árbol cuyo nombre tenga una longitud de cuatro caracteres:
`//*[string-length(name())=4]`

Funciones de agregado

Función	Devuelve
count()	cuenta elementos
avg()	Media de valores
max()	Valor máximo
min()	Valor mínimo
sum()	Suma de valores

Ejemplo: Nombre de los módulos que se cursan en dos ciclos:

`/fp/modulos/modulo/ciclos[count(ciclo)=2]/.../nombre`

Acceso a elementos con filtros de valores literales

Se trata de una extensión de las expresiones anteriores mediante la cual se puede seleccionar elementos en función del valor de sus atributos o del propio elemento. En este caso los valores serán literales. **Se indican las condiciones entre corchetes junto al elemento al cual se aplican.** Los valores literales se indican entre comillas.

Ejemplos:

- Datos del módulo de código 0373: `/fp/modulos/modulo[@codigo="0373"]` o también, si se quiere tratar el literal como un número:

`/fp/modulos/modulo[@codigo=number("0373")]`

- Nombre de los ciclos de grado medio: `/fp/ciclos/ciclo[grado="Medio"]/nombre`
- Duración de los módulos impartidos en el DAW: `/fp/modulos/modulo/ciclos/ciclo[@siglas="DAW"]/../../duracion`
- Nombre de los ciclos que tengan un reconocimiento de ECTSs: `/fp/ciclos/ciclo/ects/../../nombre` o bien `/fp/ciclos/ciclo[ects]/nombre`

Se pueden combinar múltiples condiciones:

- Nombre de los ciclos cuyas siglas son ASIR o SMR:

`/fp/ciclos/ciclo[@siglas="ASIR" or @siglas="SMR"]/nombre`

- Nombre de los ciclos cuyas siglas se encuentren, en orden alfabético, entre ASIR y SMR:

`/fp/ciclos/ciclo[@siglas>"ASIR" and @siglas<"SMR"]/nombre`

- Datos del segundo módulo de la lista de los cursados en el primer curso: `/fp/modulos/modulo[curso="1"] [2]`
- Datos del tercer ciclo que tenga reconocimiento en ECTSs: `/fp/ciclos/ciclo[ects] [3]`
- Datos de los módulos en las posiciones 3 a 5: `/fp/modulos/modulo [3 to 5]`
- Nombre de los módulos cuya relación (cociente) entre duración y ects sea mayor a 10:

`/fp/modulos/modulo[duracion div ects >10]/nombre`

Acceso a elementos con filtros de recuperados

Es el mismo caso que el apartado anterior pero con valores recuperados mediante otras expresiones

Ejemplos:

- Nombre de los ciclos en los que se cursen módulos de una duración de 80 horas o más:
 - En los elementos módulo se seleccionan aquellos que tengan una duración de 80 horas o más y se obtienen las siglas de los ciclos en los que se cursan: `/fp/modulos/modulo[duracion>=80]/ciclos/ciclo/@siglas`
 - Se muestran los nombres de los elementos ciclo cuyas siglas sean iguales a las extraídas en el primer paso: `/fp/ciclos/ciclo[@siglas=/fp/modulos/modulo[duracion>=80]/ciclos/ciclo/@siglas]/nombre`
- Nombre de módulos que tengan una duración mayor que la de Sistemas Operativos. Se usa la función `number()` para obtener el valor numérico de la duración, de lo contrario haría la comparación como cadenas (donde se cumple que "100"<"20"):

`/fp/modulos/modulo[number(duracion)>/fp/modulos/modulo[nombre="Sistemas Operativos"]/number(duracion)]/nombre`

Acceso a elementos mediante ejes

En XPath, los ejes son expresiones que permiten acceder a trozos del árbol XML, apoyándose en las relaciones de parentesco entre los nodos.

Función	Uso
self::*	Devuelve el propio nodo de contexto. Equivale a .
child::	Devuelve los nodos “hijo” del nodo de contexto
parent::*	Devuelve el nodo padre del nodo de contexto
ancestor::*	Devuelve los antepasados (padre, abuelo, ... hasta el nodo raíz) del nodo de contexto
ancestor-or-self::*	Devuelve los nodos antepasados del nodo de contexto además del propio nodo de contexto
descendant::*	Devuelve los nodos descendientes (hijo, nieto, ...) del nodo de contexto.
descendant-or-self::*	Devuelve los nodos descendientes (hijo, nieto, ...) del nodo de contexto además del propio nodo de contexto. Equivalente a //
following::*	Devuelve los nodos que aparezcan después del nodo de contexto en el documento, excluyendo a los nodos descendientes, los atributos y los nodos de espacio de nombres.
preceding::*	Devuelve los nodos que aparezcan antes del nodo de contexto en el documento, excluyendo a los nodos ascendientes, los atributos y los nodos de espacio de nombres.
preceding-sibling::*	Devuelve los hermanos mayores del nodo de contexto
following-sibling::*	Devuelve los hermanos menores del nodo de contexto
attribute::*	Atributos del nodo de contexto. Equivale a @
namespace::*	Espacio de nombres del nodo de contexto

Ejemplos:

- Elementos hermanos menores del segundo módulo:
`/fp/modulos/modulo[2]/following-sibling::*`
- Descendientes del elemento raíz que se llamen nombre:
`/fp/descendant::nombre` o también `/fp//nombre`