

RESUMEN ESQUEMAS XML

Elementos simples, la sintaxis para definir un elemento simple es:

```
<xs:element name=" " type=" " ref=" " minOccurs="int (0, 1, ...)" maxOccurs="int (0,1, ..., unbounded" default="valor" fixed="valor" />
```

Donde:

- Con name definimos el nombre del elemento
- Con type, el tipo de dato, que podéis consultar en la hoja de datos que se proporciona. Los tipos comunes son xs:string, xs:decimal, xs:integer, xs:boolean, xs:date, xs:time
- Con ref, hacemos referencia al elemento cuando lo integramos en otro elemento, como lo hemos hecho en clase.
- Con minOccurs indicamos la ocurrencia mínima, es decir el mínimo número de veces que puede aparecer un elemento.
- Con maxOccurs indicamos la ocurrencia máxima, es decir el número máximo de veces que puede aparecer un elemento, unbounded sería ilimitado.

Atributos, la sintaxis para definir un atributo es:

```
<xs:attribute name=" " type=" " ref=" " use="required | optional | prohibited" default="valor" fixed="valor" />
```

Si utilizamos default u optional el use debe ser optional

Elementos complejos

Son elementos que contienen otros elementos y/o atributos, pero también pueden ser elementos vacíos, o elementos vacíos pero que contienen un atributo.

Para definir estos elementos usamos la etiqueta <xs:complexType>

Veamos un ejemplo para un elemento vacío que contiene un atributo:

```
<xs:element name="product">
  <xs:complexType>
    <xs:attribute name="prodid" type="xs:positiveInteger"/>
  </xs:complexType>
</xs:element>
```

Si contiene otros elementos, tenemos que indicarlo con una secuencia

```
<xs:element name="person">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="firstname" type="xs:string"/>
      <xs:element name="lastname" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

Elemento con atributo, pero que solo contiene texto

```
<xs:element name="somename">
  <xs:complexType>
    <xs:simpleContent>
      <xs:extension base="basetype">
        ...
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>
</xs:element>
```

Restricciones (Facets)

Permiten establecer reglas complejas que deben de cumplir los datos. En este caso dentro de la etiqueta simpleType se indica una etiqueta restriction, dentro de la cual se establecen las posibles restricciones. Sintaxis:

```
<xs:simpleType name="nombre">
  <xs:restriction base="tipo">
    ... definición de la restricción ...
  </xs:restriction>
</xs:simpleType>
```

El atributo obligatorio base sirve para indicar en qué tipo de dato nos basamos al definir la restricción.

Las posibles restricciones que se pueden establecer son:

Facets	Descripción
<code>xs:length</code>	Especifica una longitud fija
<code>xs:minLength</code>	Especifica una longitud mínima
<code>xs:maxLength</code>	Especifica una longitud máxima
<code>xs:pattern</code>	Especifica un patrón de caracteres admitidos
<code>xs:enumeration</code>	Especifica una lista de valores admitidos.
<code>xs:whiteSpace</code>	Especifica cómo se debe tratar a los posibles espacios en blanco, las tabulaciones, los saltos de línea y los retorno de carro que puedan aparecer.
<code>xs:maxInclusive</code>	Especifica que el valor debe ser menor o igual que el indicado.
<code>xs:minInclusive</code>	Especifica que el valor debe ser mayor o igual que el indicado.
<code>xs:maxExclusive</code>	Especifica que el valor debe ser menor que el indicado.
<code>xs:minExclusive</code>	Especifica que el valor debes ser mayor que el indicado
<code>xs:totalDigits</code>	Especifica el número máximo de dígitos que pueden tener un número.
<code>xs:fractionDigits</code>	Especifica el número máximo de decimales que puede tener un número.

`xs:pattern` sirve para definir un patrón de cadenas de caracteres admitidos.

Veamos en la siguiente tabla más posibilidades de los patrones:

Símbolo	equivalencia
.	Cualquier carácter
\w	Cualquier letra
\d	Cualquier dígito
\D	Cualquier carácter no dígito
\s	Cualquier carácter de espaciado
\S	Cualquier carácter de no espaciado
x{n}	Tiene que haber n veces lo que haya delante de los corchetes
x{n,}	Tiene que haber n o más veces lo que haya delante de los corchetes
x{n,m}	Tiene que haber entre n y m lo que haya delante de los corchetes
[Abc]	Tiene que haber algún carácter de los de dentro.
[A-Z]	Tiene que haber uno de los caracteres de la A a la Z
[^abc]	Negación de un grupo de caracteres
[F-J-[H]]	Sustracción de un carácter de un rango.
(a b)	Alternativa de dos expresiones
x*	0 o más ocurrencias del elemento x
x+	1 o más ocurrencias del elemento x
x?	0 o 1 ocurrencias del elemento x

Indicadores

Los indicadores permiten establecer cómo se van a escribir o utilizar los elementos en un documento XML. Hay siete tipos de indicadores que se pueden clasificar en;

- indicadores de orden secuencia (sequence), todo (all) y elección (choice)
- indicadores de ocurrencia; `maxOccurs` y `minOccurs` – ya hemos hablado de ellos
- indicadores de grupo: de elementos (group) y de atributos (attributeGroup) – no los hemos visto

Indicadores de orden

- `xs:all`: han de incluirse todos los elementos, sin importar el orden.
- `xs:sequence`: han de incluirse todos los elementos, respetando el orden.
- `xs:choice`: es necesario incluir uno de los elementos.