

Declaraciones en DTD

Declaraciones

Los DTDs describen la estructura de los documentos XML mediante declaraciones. Hay cuatro tipos de declaraciones:

- [Declaraciones de entidades](#)
- [Declaraciones de notaciones](#)
- [Declaraciones de elementos](#), que indican los elementos permitidos en un documento y su contenido (que puede ser simplemente texto u otros elementos).
- [Declaraciones de atributos](#), que indican los atributos permitidos en cada elemento y el tipo o valores permitidos de cada elemento.

Declaración de entidades

Una entidad consiste en un nombre y su valor (son similares a las constantes en los lenguajes de programación). Con algunas excepciones, el procesador XML sustituye las referencias a entidades por sus valores antes de procesar el documento. Una vez definida la entidad, se puede utilizar en el documento escribiendo una referencia a la entidad, que empieza con el carácter "&", sigue con el nombre de la entidad y termina con ";". (es decir, &nombreEntidad;)

Las entidades pueden ser internas o externas y tanto unas como otras pueden ser generales o paramétricas.

Las declaraciones de entidades internas (generales) siguen la siguiente sintaxis:

```
<!ENTITY nombreEntidad "valorEntidad">
```

En las declaraciones de entidades externas (generales) se distinguen dos casos:

- La entidad hace referencia a un fichero de texto y en ese caso la entidad se sustituye por el contenido del archivo.

La entidad puede ser una entidad de sistema, con la siguiente sintaxis:

```
<!ENTITY nombreEntidad SYSTEM "uri">
```

o puede ser una entidad pública, con la siguiente sintaxis:

```
<!ENTITY nombreEntidad PUBLIC "fpi" "uri">
```

- La entidad hace referencia a un fichero que no es de texto (por ejemplo, una imagen) y en ese caso la entidad no se sustituye por el contenido del archivo.

La entidad puede ser una entidad de sistema, con la siguiente sintaxis:

```
<!ENTITY nombreEntidad SYSTEM "uri" NDATA tipo>
```

o puede ser una entidad pública, con la siguiente sintaxis:

```
<!ENTITY nombreEntidad PUBLIC "fpi" "uri" NDATA tipo>
```

En todos estos casos:

- "nombreEntidad" es el nombre de la entidad.
- "valorEntidad" es el valor de la entidad.
- "uri" es el camino (absoluto o relativo) hasta un archivo.
- "tipo" es el tipo de archivo (gif, jpg, etc).
- "fpi" es un indentificador público formal (Formal Public Identifier).

Las declaraciones de entidades paramétricas siguen la mismas sintaxis que las generales, pero llevan el carácter "%" antes del nombre de la entidad. Por ejemplo:

```
<!ENTITY % nombreEntidad "valorEntidad">
```

```
<!ENTITY % nombreEntidad SYSTEM "uri">
```

```
<!ENTITY % nombreEntidad SYSTEM "uri" NDATA tipo>
```

La diferencia entre entidades generales y paramétricas es que las entidades paramétricas se sustituyen por su valor en todo el documento (incluso en la propia declaración de tipo de documento) mientras que las generales no se sustituyen en la declaración de tipo de documento.

Declaración de notaciones

Las notaciones se usan en XML para definir las entidades externas que no va a analizar el procesador XML (aunque sí lo hará la aplicación que trate un documento). Para hacer referencia estas entidades no se utiliza la notación habitual (&nombreEntidad;), sino que se utiliza el nombre de la entidad directamente.

Declaración de elementos

Las declaraciones de los elementos siguen la siguiente sintaxis:

```
<!ELEMENT nombreElemento (contenido)>
```

en la que "nombreElemento" es el nombre del elemento, y "(contenido)" una expresión que describe el contenido del elemento.

Para definir el contenido del elemento se pueden utilizar los términos EMPTY, (#PCDATA) o ANY o escribir expresiones más complejas:

- **EMPTY**: significa que el elemento es vacío, es decir, que no puede tener contenido. Los elementos vacíos pueden escribirse con etiquetas de apertura y cierre sin nada entre ellos, ni siquiera espacios, o con una etiqueta vacía. EMPTY debe escribirse sin paréntesis.

```
<!DOCTYPE ejemplo [
  <!ELEMENT ejemplo EMPTY>
]>
```



```
<ejemplo></ejemplo>
```



```
<ejemplo />
```



```
<ejemplo>Esto es un ejemplo</ejemplo>  
ERROR: contiene texto -->
```

```
<!--
```



```
<ejemplo><a></a></ejemplo>  
ERROR: contiene un elemento <a> -->
```

```
<!--
```

- **(#PCDATA)**: significa que el elemento puede contener texto. #PCDATA debe escribirse entre paréntesis.

```
<!DOCTYPE ejemplo [  
  <!ELEMENT ejemplo (#PCDATA)>  
]>
```



```
<ejemplo />
```



```
<ejemplo>Esto es un ejemplo</ejemplo>
```



```
<ejemplo><a></a></ejemplo>  
ERROR: contiene un elemento <a> -->
```

```
<!--
```

- **ANY**: significa que el elemento puede contener cualquier cosa (texto y otros elementos). ANY debe escribirse sin paréntesis.

```
<!DOCTYPE ejemplo [  
  <!ELEMENT ejemplo ANY>  
  <!ELEMENT a ANY>  
]>
```



```
<ejemplo />
```



```
<ejemplo>Esto es un ejemplo</ejemplo>
```



```
<ejemplo>Esto es <a>un ejemplo</a></ejemplo>
```

Para indicar que un elemento puede o debe contener otros elementos se deben indicar los elementos, utilizando los conectores y modificadores siguientes:

- **, (coma)**: significa que el elemento contiene los elementos en el orden indicado.

```
<!DOCTYPE ejemplo [  
  <!ELEMENT ejemplo (a, b)>  
  <!ELEMENT a EMPTY>  
  <!ELEMENT b EMPTY>  
]>
```

- | (o lógico): significa que el elemento contiene uno de los dos elementos.

```
<!DOCTYPE ejemplo [
  <!ELEMENT ejemplo (a | b)>
  <!ELEMENT a EMPTY>
  <!ELEMENT b EMPTY>
]>
```



```
<ejemplo><a /></ejemplo>
```



```
<ejemplo><b /></ejemplo>
```



```
<ejemplo><a /><b /></ejemplo>
ERROR: están los dos elementos -->
```

```
<!--
```



```
<ejemplo></ejemplo>
ERROR: no hay ningún elemento -->
```

```
<!--
```

- ?: significa que el elemento puede aparecer o no, pero sólo una vez.

```
<!DOCTYPE ejemplo [
  <!ELEMENT ejemplo (a, b?)>
  <!ELEMENT a EMPTY>
  <!ELEMENT b EMPTY>
]>
```



```
<ejemplo><a /></ejemplo>
```



```
<ejemplo><a /><b /></ejemplo>
```



```
<ejemplo><b /></ejemplo>
ERROR: falta el elemento <a /> -->
```

```
<!--
```



```
<ejemplo><b /><b /></ejemplo>
ERROR: el elemento <b /> aparece dos veces -->
```

```
<!--
```

- *: significa que el elemento puede no aparecer o aparecer una o más veces.

```
<!DOCTYPE ejemplo [
  <!ELEMENT ejemplo (a*, b)>
  <!ELEMENT a EMPTY>
  <!ELEMENT b EMPTY>
]>
```



```
<ejemplo><b /></ejemplo>
```



```
<ejemplo><a /><b /></ejemplo>
```



```
<ejemplo><a /><a /><b /></ejemplo>
```



```
<ejemplo><b /><a /></ejemplo>
```

ERROR: el elemento <a /> aparece después de -->

```
<!--
```

- +: significa que el elemento tiene que aparecer una o más veces (no puede no aparecer).

```
<!DOCTYPE ejemplo [
  <!ELEMENT ejemplo (a+, b)>
  <!ELEMENT a EMPTY>
  <!ELEMENT b EMPTY>
]>
```



```
<ejemplo><a /><b /></ejemplo>
```



```
<ejemplo><a /><a /><b /></ejemplo>
```



```
<ejemplo><b /></ejemplo>
```

ERROR: falta el elemento <a /> -->

```
<!--
```

- (): permite agrupar expresiones.

```
<!DOCTYPE ejemplo [
  <!ELEMENT ejemplo (a, (a|b))>
  <!ELEMENT a EMPTY>
  <!ELEMENT b EMPTY>
]>
```



```
<ejemplo><a /><a /></ejemplo>
```



```
<ejemplo><a /><b /></ejemplo>
```



```
<ejemplo><a /></ejemplo>
```

ERROR: falta el elemento <a /> o -->

```
<!--
```

```
<!DOCTYPE ejemplo [
  <!ELEMENT ejemplo ((a, b)|(b, a))>
  <!ELEMENT a EMPTY>
  <!ELEMENT b EMPTY>
]>
```



```
<ejemplo><a /><b /></ejemplo>
```



```
<ejemplo><b /><a /></ejemplo>
```



```
<ejemplo><a /><a /></ejemplo>
```

```
ERROR: sólo admite <a /><b /> o <b /><a /> -->
```

```
<!--
```

Declaración de atributos

Una declaración de atributos sigue la siguiente sintaxis:

```
<!ATTLIST nombreElemento nombreAtributo tipoAtributo valorInicialAtributo >
```

en la que:

- "nombreElemento" es el nombre del elemento para el que se define un atributo.
- "nombreAtributo" es el nombre del atributo.
- "tipoAtributo" es el tipo de datos .
- "valorInicialAtributo" es el valor predeterminado del atributo (aunque también puede indicar otras cosas).

Para definir varios atributos de un mismo elemento, se puede utilizar una o varias declaraciones de atributos. Los siguientes ejemplos son equivalentes:

```
<!ATTLIST nombreElemento nombreAtributo1 tipoAtributo1
valorInicialAtributo1>
<!ATTLIST nombreElemento nombreAtributo2 tipoAtributo2
valorInicialAtributo2>
```

```
<!ATTLIST nombreElemento
nombreAtributo1 tipoAtributo1 valorInicialAtributo1
nombreAtributo2 tipoAtributo2 valorInicialAtributo2
>
```

Los tipos de atributos son los siguientes:

- **CDATA:** el atributo contiene caracteres (sin restricciones).

```
<!DOCTYPE ejemplo [
  <!ELEMENT ejemplo EMPTY>
  <!ATTLIST ejemplo color CDATA #REQUIRED>
]>
```



```
<ejemplo color="" />
```



```
<ejemplo color="amarillo" />
```



```
<ejemplo color="azul marino #000080" />
```



```
<ejemplo />
<!-- ERROR: falta el atributo
"color", obligatorio debido al #REQUIRED -->
```



```
<ejemplo sabor="dulce" />
"sabor" no está definido -->
```

```
<!-- ERROR: el atributo
```

- **NMTOKEN**: el atributo sólo contiene letras, dígitos, y los caracteres punto ".", guión "-", subrayado "_" y dos puntos ":".

```
<!DOCTYPE ejemplo [
  <!ELEMENT ejemplo EMPTY>
  <!ATTLIST ejemplo color NMTOKEN #REQUIRED>
]>
```



```
<ejemplo color="" />
```



```
<ejemplo color="azul-marino" />
```



```
<ejemplo color="1" />
```



```
<ejemplo color="azul marino" />
espacio en blanco -->
```

```
<!-- ERROR: hay un
```



```
<ejemplo color="#F0F0F0" />
contiene el carácter # -->
```

```
<!-- ERROR:
```

- **NMTOKENS**: el atributo sólo contiene letras, dígitos, y los caracteres punto ".", guión "-", subrayado "_", dos puntos ":" (como el tipo NMTOKEN) y también espacios en blanco.

```
<!DOCTYPE ejemplo [
  <!ELEMENT ejemplo EMPTY>
  <!ATTLIST ejemplo color NMTOKENS #REQUIRED>
]>
```



```
<ejemplo color="" />
```



```
<ejemplo color="1" />
```



```
<ejemplo color="azul marino" />
```



```
<ejemplo color="2*2" />
asterisco -->
```

```
<!-- ERROR: hay un
```

- **valores**: el atributo sólo puede contener uno de los términos de una lista. La lista se escribe entre paréntesis, con los términos separados por una barra vertical "|".

```
<!DOCTYPE ejemplo [
  <!ELEMENT ejemplo EMPTY>
  <!ATTLIST ejemplo color (azul|blanco|rojo) #REQUIRED>
]>
```



```
<ejemplo color="" />
```



```
<ejemplo color="azul" />
```



```
<ejemplo color="verde" />  
"verde" no está en la lista de valores -->
```

```
<!-- ERROR:
```

- **ID:** el valor del atributo (no el nombre) debe ser único y no se puede repetir en otros elementos o atributos.

```
<!DOCTYPE ejemplo [  
  <!ELEMENT ejemplo (libro*)>  
  <!ELEMENT libro (#PCDATA) >  
  <!ATTLIST libro codigo ID #REQUIRED>  


```



```
<ejemplo>  
  <libro codigo="L1">Poema de Gilgamesh</libro>  
  <libro codigo="L2">Los preceptos de Ptah-Hotep</libro>  
</ejemplo>
```



```
<ejemplo>  
  <libro codigo="1">Poema de Gilgamesh</libro>  
  <libro codigo="L2">Los preceptos de Ptah-Hotep</libro>  
</ejemplo>
```

```
<!--
```

```
ERROR: el valor de un atributo de tipo ID no puede empezar con un  
número -->
```



```
<ejemplo>  
  <libro codigo="L1">Poema de Gilgamesh</libro>  
  <libro codigo="L1">Los preceptos de Ptah-Hotep</libro>  
</ejemplo>
```

```
<!--
```

```
ERROR: no se puede repetir un atributo de tipo ID -->
```

- **IDREF:** el valor del atributo debe coincidir con el valor del atributo ID de otro elemento.

```
<!DOCTYPE ejemplo [  
  <!ELEMENT ejemplo ((libro|prestamo)*)>  
  <!ELEMENT libro (#PCDATA) >  
  <!ATTLIST libro codigo ID #REQUIRED>  
  <!ELEMENT prestamo (#PCDATA) >  
  <!ATTLIST prestamo libro IDREF #REQUIRED>  


```



```
<ejemplo>  
  <libro codigo="L1">Poema de Gilgamesh</libro>  
  <prestamo libro="L1">Numa Nigerio</prestamo>  
</ejemplo>
```




```
<ejemplo>
  <libro codigo="L1">Poema de Gilgamesh</libro>
  <prestamo libro="L2">Numa Nigerio</prestamo>
ERROR: el valor "L2" no es ID de ningún elemento -->
</ejemplo>
```

- **IDREFS**: el valor del atributo es una serie de valores separados por espacios que coinciden con el valor del atributo ID de otros elementos.

```
<!DOCTYPE ejemplo [
  <!ELEMENT ejemplo ((libro|prestamo)*)>
  <!ELEMENT libro (#PCDATA) >
  <!ATTLIST libro codigo ID #REQUIRED>
  <!ELEMENT prestamo (#PCDATA) >
  <!ATTLIST prestamo libro IDREFS #REQUIRED>
]>
```



```
<ejemplo>
  <libro codigo="L1">Poema de Gilgamesh</libro>
  <libro codigo="L2">Los preceptos de Ptah-Hotep</libro>
  <prestamo libro="L1 L2">Numa Nigerio</prestamo>
</ejemplo>
```



```
<ejemplo>
  <libro codigo="L1">Poema de Gilgamesh</libro>
  <libro codigo="L2">Los preceptos de Ptah-Hotep</libro>
  <prestamo libro="L1">Numa Nigerio</prestamo>
</ejemplo>
```



```
<ejemplo>
  <libro codigo="L1">Poema de Gilgamesh</libro>
  <libro codigo="L2">Los preceptos de Ptah-Hotep</libro>
  <prestamo libro="L3">Numa Nigerio</prestamo>
ERROR: el valor "L3" no es ID de ningún elemento -->
</ejemplo>
```

- **ENTITY**: el valor del atributo es alguna entidad definida en la DTD.
- **ENTITIES**: el valor del atributo es alguna de las entidades de una lista de entidades definida en la DTD.
- **NOTATION**: el valor del atributo es alguna notación definida en la DTD.

Los valores iniciales de los atributos son los siguientes:

- **#REQUIRED**: el atributo es obligatorio, aunque no se especifica ningún valor predeterminado.

```
<!DOCTYPE ejemplo [
  <!ELEMENT ejemplo EMPTY>
  <!ATTLIST ejemplo color CDATA #REQUIRED>
]>
```



```
<ejemplo color="" />
```



```
<ejemplo color="amarillo" />
```



```
<ejemplo color="azul marino #000080" />
```



```
<ejemplo />  
el atributo "color" -->
```

```
<!-- ERROR: falta
```

- **#IMPLIED**: el atributo no es obligatorio y no se especifica ningún valor predeterminado.

```
<!DOCTYPE ejemplo [  
  <!ELEMENT ejemplo EMPTY>  
  <!ATTLIST ejemplo color CDATA #IMPLIED>  


```



```
<ejemplo />
```



```
<ejemplo color="" />
```



```
<ejemplo color="amarillo" />
```



```
<ejemplo color="azul marino #000080" />
```

- **#FIXED valor**: el atributo tiene un valor fijo.

```
<!DOCTYPE ejemplo [  
  <!ELEMENT ejemplo EMPTY>  
  <!ATTLIST ejemplo color CDATA #FIXED "verde">  


```



```
<ejemplo />
```



```
<ejemplo color="verde" />
```



```
<ejemplo color="" />  
atributo "color" no tiene el valor "verde" -->
```

```
<!-- ERROR: el
```



```
<ejemplo color="amarillo" />  
atributo "color" no tiene el valor "verde" -->
```

```
<!-- ERROR: el
```

- **valor**: el atributo tiene un valor predeterminado.

```
<!DOCTYPE ejemplo [  
  <!ELEMENT ejemplo EMPTY>  
  <!ATTLIST ejemplo color CDATA "verde">  


```



```
<ejemplo />
```



```
<ejemplo color="" />
```



```
<ejemplo color="amarillo" />
```



```
<ejemplo color="verde" />
```