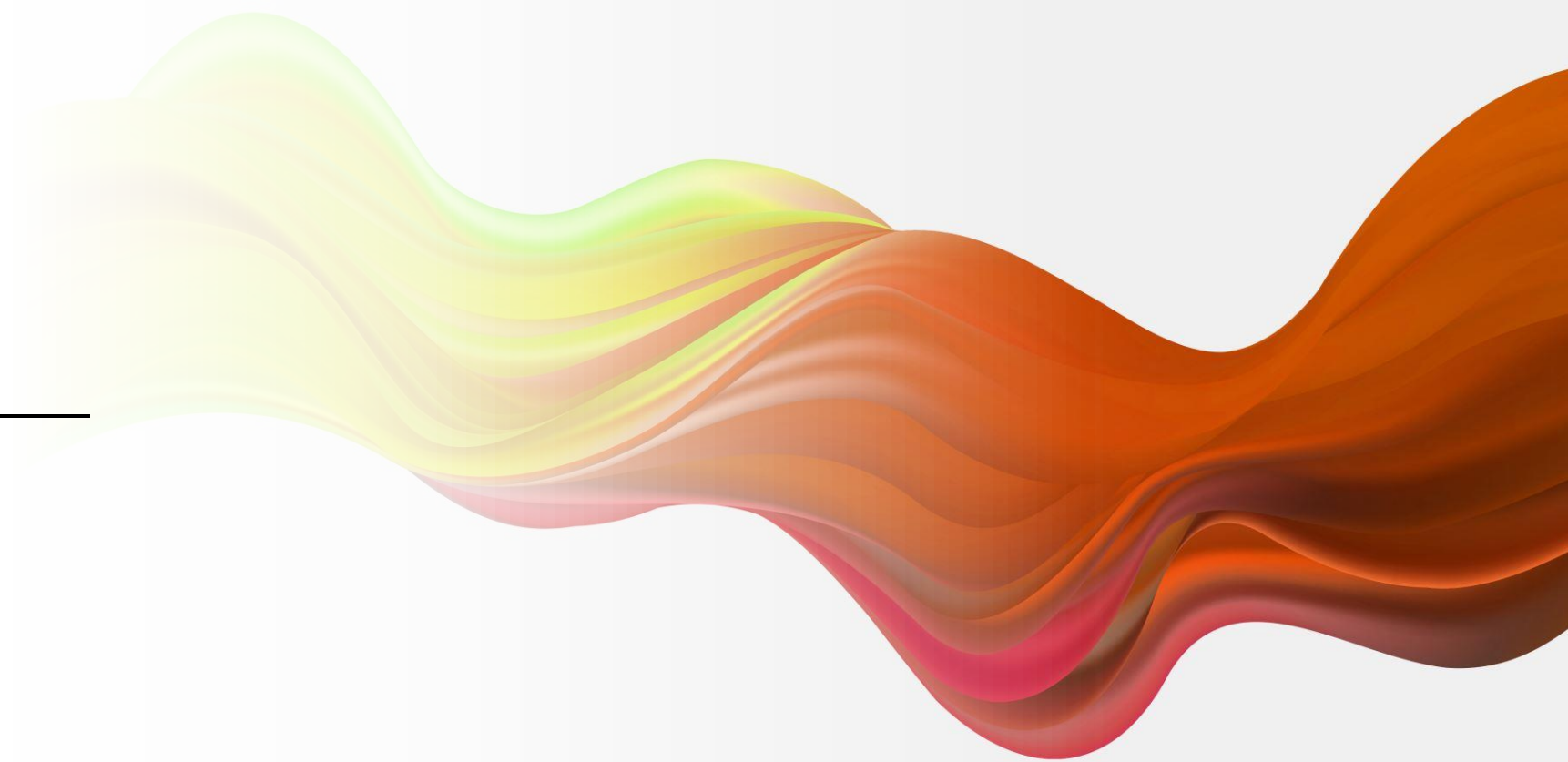




# XML--DTD

---



# DTD. Generalidades

- DTD: Definición tipo de documento
- Descripción de la estructura de un documento XML.
- Se especifica:
  - Qué elementos tienen que aparecer
  - En que orden
  - Cuales son optativos
  - Cuales son obligatorios
  - Qué atributos tienen los elementos.
- Es un mecanismo de validación del documento que existía antes de que apareciera XML.
- Se usaba para validar documentos SGML (precursor de todos los lenguajes de marcas).
- Existen documentos validados con DTD
- Se tiende a utilizar otras técnicas de validación SCHEMAS(xsd)

# DTD. Generalidades

- Generación automática de DTDs.
  - Si los documentos tienen muchos datos, los DTD que los validan suelen ser muy extensos. Para facilitarlos existen herramientas de generación automática de DTDs a partir de un documento XML.
  - Estas herramientas realizan deducciones “gruesas” de las reglas gramaticales, que el desarrollador debe afinar para que se ajusten a los requisitos semánticos
  - Existen:
    - Generadores on-line
    - Generadores instalables
    - Editores XML que generan DTD a partir de un documento XML
      - XMLSpy de Altova

# ESTRUCTURA DE UN DTDs

## • Declaración de un DTDs

- <!DOCTYPE...>
  - Es la instrucción que indica que DTD validará el documento XML
  - Aparece al comienzo del documento
  - El primer dato que aparece es el nombre del elemento del raíz.
  - Las características que defines el tipo son:
    - **Ubicación:**
      - **Interna:** dentro del propio documento XML ( pag 6 ,7 y 8 apuntes)
      - **Externa** :las reglas aparecen en un archivo independiente.
      - **Mixto:** mezcla de las 2 anteriores.
    - **Carácter.**
      - **Privado,** se identifica con la **palabra SYSTEM**
      - **Público** se identifica con **PUBLIC.**( pag 9 y 10 apuntes)

# ESTRUCTURA DE UN DTDs

## Ejemplo de documento XML con DTD interna

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<!DOCTYPE Casas_Rurales [
```

```
    <!ELEMENT Casas_Rurales (Casa)*>
```

```
    <!ELEMENT Casa (Dirección, Descripción, Estado, Tamaño)>
```

```
    <!ELEMENT Dirección (#PCDATA) >
```

```
    <!ELEMENT Descripción (#PCDATA) >
```

```
    <!ELEMENT Estado (#PCDATA) >
```

```
    <!ELEMENT Tamaño (#PCDATA) >
```

```
]>
```

```
<Casas_Rurales>
```

```
    <Casa>
```

```
        <Dirección>Calle Las Palmas 23. La Sagra. Toledo </Dirección>
```

```
        <Descripción>Se trata de una casa del siglo XVII, ...</Descripción>
```

```
        <Estado>El estado de conservación es magnífico, ...</Estado>
```

```
        <Tamaño>La casa tiene 259 metros cuadrados, ...</Tamaño>
```

```
    </Casa>
```

```
    <Casa>
```

```
        <Dirección>Calle Or 5, Fregenal de la Sierra. Badajoz </Dirección>
```

```
        <Descripción>De arquitectura espectacular la casa ...</Descripción>
```

```
        <Estado>Recientemente restaurada su estado actual es ...</Estado>
```

```
        <Tamaño>La superficie habitable es de ...</Tamaño>
```

```
    </Casa>
```

```
</Casas_Rurales>
```

# ESTRUCTURA DE UN DTDs. Ejemplo dtd externa

```
<?xml version="1.0" encoding="iso-8859-1"?>
```

```
<!DOCTYPE BDsms SYSTEM "BDsms.dtd">
```

```
<BDsms>
```

```
<sms>
```

```
<telefono>23456756</telefono>
```

```
<fecha>1/3/2012</fecha>
```

```
<hora>11:30</hora>
```

```
<mensaje>Juego : parchis</mensaje>
```

```
</sms>
```

```
<sms>
```

```
<telefono>1213231</telefono>
```

```
<fecha>1/3/2012</fecha>
```

```
<hora>11:30</hora>
```

```
<mensaje>Juego : oca</mensaje>
```

```
</sms>
```

```
<sms>
```

```
<telefono>435465366</telefono>
```

```
<fecha>1/3/2012</fecha>
```

```
<hora>11:30</hora>
```

```
<mensaje>Juego : quien es quien</mensaje>
```

```
</sms>
```

```
</BDsms>
```

# ESTRUCTURA DE UN DTDs

BDsms.dtd

```
<!ELEMENT BDsms (sms*)>
```

```
<!ELEMENT sms (telefono,fecha,hora,mensaje ) >
```

```
<!ELEMENT telefono (#PCDATA)>
```

```
<!ELEMENT fecha (#PCDATA)>
```

```
<!ELEMENT hora (#PCDATA)>
```

```
<!ELEMENT mensaje (#PCDATA)>
```

# ESTRUCTURA DE UN DTDs

## • Componentes de un DTDs:

- Elementos
- Atributos
- Entidad
- Notificación

- **MIRAD EL FICHERO QUE ADJUNTO DONDE APARECE EN ESQUEMA COMO UTILIZAR EELEMENTOS Y ATRIBUTOS**



[XML esquema\\_dtd.pdf](#)



# COMPONENTES DE DTDs

## • Elementos

### • Declaración:

- **<!element NombreElemento type>**

- **NombreElemento:**

- Determina el nombre de la etiqueta
- Debe ser único dentro de un DTD
- No debe contener caractere & o empezar con XML...

- **Tipo del elemento(type)**

- Vacio
- Solo texto
- ANY
- Solo elementos
- mixto

# ELEMENTOS

- **Elementos vacíos**

- No tienen contenido
- Pueden contener atributos
- `<!ELEMENT NombreElemento EMPTY>`
- `<imagen foto="xy.gif/">`

- **Elementos de sólo texto**

- Contienen datos de tipo carácter
- `<!ELEMENT NombreElemento (#PCDATA)>`

- **Elementos ANY ( no se usan demasiado)**

- Pueden contener datos de caracteres, tipos de elementos declarados o una mezcla de ambos.
- `<!ELEMENT NombreElemento ANY>`
- Carecen de estructura definida.
- A veces se usan en el desarrollo de prueba.

## ELEMENTOS

### • Elementos de elementos

- Contiene elementos secundarios
- Se declaran especificando un modelo de contenido en el elemento, con solo elementos secundarios

- `<!ELEMENT NombreElemento ModeloContenidos>`

### • Modelo de contenidos

- **E**s el patrón que establece los elementos secundarios y el orden de aparición.
- Combinación de símbolos y elementos secundarios
- Símbolos del modelo de contenidos:
  - `()` => grupo de elementos
  - `,` => orden dentro de un grupo
  - `|` => grupo de elementos alternativos
  - `?` => opcional ( 0 o 1 vez)
  - `*` => opcional (0...N)
  - `+` => obligatorio (1..N)
  - Sin símbolo => obligatorio solo una vez.
- **`<!ELEMENT historial(paciente,(visita|tratamiento)+, hobbies?, alergias*)>`**

## ELEMENTOS

### • Elementos de elementos mixtos

- Contiene datos de caracteres y elementos secundarios
- **Modelo de contenidos:**
  - Los elementos de solo texto deben aparecer al principio
  - No limita el orden de los elementos o el número de veces que puede aparecer.
  - **<!ELEMENT (memoria(#PCDATA)|titulo|asunto)\*>**
  - No es recomendable su utilización.
  - Carece de estructura definida.
- **Hacer ejemplo de la pag 30 de los apuntes**

## ATRIBUTOS ATTLIST

- Los atributos especifican información adicional del elemento.

- **DECLARACIÓN**

**<!ATTLIST** nombre\_de\_elemento **nombre\_de\_atributo** tipo declaración\_por\_defecto>

- **elemento**: es el nombre del elemento al que corresponden los atributos
- **nombre**: es el nombre del atributo
- **tipo**: CDATA, (valor | valor | ... ), ID, IDREF, IDREFS, NMTOKEN, NMTOKENS
- **tratamiento\_por\_defecto**: #REQUIRED, #IMPLIED, #FIXED valor\_por\_defecto, valor\_por\_defecto

<!ELEMENT comprador EMPTY>

<!ATTLIST comprador apellido CDATA nombre CDATA fecha CDATA compra CDATA precio CDATA identificador ID>

<comprador apellido="Martinez" nombre="Manuel"  
fecha="21/03/2021" compra="Manzanas"  
precio="3" identificador="T225" />

## ATRIBUTOS ATTLIST

- TIPOS DE ATRIBUTOS: (pag 35 de los apuntes)
  - **CDATA**: Caracteres que no contienen etiquetas.
    - `<!ATTLIST persona apellido CDATA #REQUIRED  
nombre CDATA #REQUIRED  
fecha CDATA #IMPLIED  
Ciudad CDATA "Madrid">`
  - **ENUMERADO**: presenta un conjunto de valores permitido por el atributo, son valores de tipo token.
    - `<!ATTLIST jugador nombre CDATA #REQUIRED  
Posición (izquierda | centro | derecha) "centro">`
- \*(no es necesario que los valores vayan entre comillas)
- \*(el operador pipe "|" es opcional, los diferentes valores pueden ponerse con espacios)

## ATRIBUTOS ATTLIST

- TIPOS DE ATRIBUTOS:

- **ID:** Se pone un solo atributo ID por tipo de elemento.

- Identifica de forma única al elemento.
- Este atributo se suele definir como #REQUIRED.
- El valor del atributo ID debe ser un nombre de XML válido.

(ver pag 37 y 38 de los apuntes).

- **IDREF:** El valor del atributo IDREF debe corresponder con el valor de un atributo id de algún elemento existente

(ver pag 38 y 39 de los apuntes)

```
<!ELEMENT nombre_estudiante (#PCDATA)>
<!ATTLIST nombre_estudiante estudiante_id ID #REQUIRED>
<nombre_estudiante estudiante_id="E50716">
    Pepito Grillo
</ nombre_estudiante>
```

## ATRIBUTOS ATTLIST

- TIPOS DE ATRIBUTOS:
  - **IDREFS:** Referencia a más de un atributo.

(ver pag 38 y 39 de los apuntes)

```
<!ELEMENT nombre_estudiante (#PCDATA)>  
  <!ATTLIST nombre_estudiante estudiante_id ID #REQUIRED>  
  <nombre_estudiante estudiante_id="E50716">  
    Pepito Grillo  
  </ nombre_estudiante>
```



## ATRIBUTOS ATTLIST

- TIPOS DE ATRIBUTOS:

- **NMTOKEN:** Cadena de caracteres, con la restricción de que ha de ser un nombre XML. No contiene espacios en blanco( letras, números, puntos, guiones, subrayado, dos puntos) (.
  - **NMTOKENS** : Admite espacios en blanco.

(ver pag 39 y 44 de los apuntes)

## ATRIBUTOS ATTLIST

- VALORES DE TRATAMIENTO POR DEFECTO:
  - **#REQUIRED**=> El atributo es obligatorio.
  - **#IMPLIED**=> El atributo es opcional.
  - **#FIXED** valor=> El atributo tienen un valor fijo.
  - **default**=> El valor predeterminado del atributo.
- **(ver pag 35 de los apuntes)**