



XML: eXtensible Markup Languaje

Tecnologías XML Introducción

Dr. Juan Manuel Cueva Lovelle
Departamento de Informática
Universidad de Oviedo
cueva@uniovi.es

- Lenguajes de marcas
- Introducción a XML
- Documentos XML
- Documentos XML bien formados
- Documentos XML válidos
- Ventajas de XML
- Inconvenientes de XML
- Bibliografía
- Referencias Web
- Ejercicios propuestos

Software y estándares para la Web

Esquema

Grado en
Ingeniería
Informática
del Software

- **Lenguajes de marcas**
- Introducción a XML
- Documentos XML
- Documentos XML bien formados
- Documentos XML válidos
- Ventajas de XML
- Inconvenientes de XML
- Bibliografía
- Referencias Web
- Ejercicios propuestos

- Desde el HTML a XHTML
 - HTML
 - XHTML
 - Hojas de Estilo (CSS)
- Desde SGML a XML
 - SGML
 - XML
 - DTD
 - Esquemas XML

Software y estándares para la Web

Lenguajes de Marcas (II): HTML en su inicio

Grado en
Ingeniería
Informática
del Software


AlCapone.html

```
<html>
<head>
<title>Pizzeria Al Capone</title>
</head>
<body bgcolor="blue" text="yellow"
  link="red" vlink="white">

<h1>Pizzería Al Capone</h1>
<p>Lista de enlaces</p>
<ul>
<li><a href="Pizzas.html">
  Tipos de Pizzas</a></li>
<li><a href="http://www.mafia.it">
  Patrocinadores</a></li>
<li><a href="#Contacto">Contacto</a></li>
</ul>

<h2><a name="Contacto">Contacto</a></h2>
<p><font color="red">Dirección: </font>
  C/ Génova Nº 3, Oviedo, España</p>
<p><font color="red">Teléfono: </font>
  985203040</p>
</body>
</html>
```

Página visualizada



Problema:
Presentación y contenido
mezclados

Software y estándares para la Web

Lenguajes de Marcas (III): XHTML

Grado en
Ingeniería
Informática
del Software

- HTML (1991). HyperText Markup Language
- HTML 2.0 (1995)
- HTML 4.0 (1997)



- XHTML (eXtensible HyperText Markup Language)
- XHTML es HTML expresado como XML válido
- XHTML 1.0 fue una recomendación del W3C (26-Enero-2000)
- XHTML 1.1 fue una recomendación del W3C (31-Mayo-2001)



- HTML 5 (2008)
- HTML 5.1 (2016)
- HTML 5.2 (2017)
- El estándar conocido como **XHTML5.2** está siendo desarrollado como una adaptación a XML de la especificación **HTML5.2**
- Como es XML se pueden utilizar fácilmente herramientas creadas para procesamiento de documentos XML genéricos



Software y estándares para la Web

Lenguajes de marcas (IV): Hojas de estilo CSS

Grado en
Ingeniería
Informática
del Software

AlCapone.html

```
<html>
<head>
<title>Pizzeria Al Capone</title>
<link rel="stylesheet" href="pizzeria.css">
</head>
<body>
<h1>Pizzería Al Capone</h1>
<p>Lista de enlaces</p>
<ul>
<li><a href="Pizzas.html">
  Tipos de Pizzas</a></li>
<li><a href="http://www.mafia.it">
  Patrocinadores</a></li>
<li><a href="#Contacto">
  Contacto</a></li>
</ul>

<h2><a name="Contacto">Contacto</a></h2>
<p><span class="item">Dirección:</span>
  C/ Génova Nº 3, Oviedo, España</p>
<p><span class="item">Teléfono:</span>
  985203040</p>
</body>
</html>
```

pizzeria.css

```
body { color : yellow;
        background: blue
    }
a:link { color: red }
a:visited { color: white }
span.item { color : red }
```

Página visualizada



Enlace a hoja de estilo

Sin aspectos visuales

Identificación elementos

Software y estándares para la Web

Lenguajes de Marcas (V): Reutilización de estilos



Misma
apariencia

pizzas.html

```
<html>
<head>
<title>Tipos Pizzas</title>
<link rel="stylesheet" href="pizzeria.css">
</head>
<body>
<h1>Pizzas del Restaurante Al Capone</h1>
<table><caption>Tipos de Pizzas</caption>
<tr>
<th>
<td>Pizza</td><td>Ingredientes</td><td>Precio</td></th>
<tr>
<td>Barbacoa</td>
<td>Salsa barbacoa, Mozzarella, Pollo, Bacon, Ternera</td>
<td>8 &euro;</td></tr>
...
<tr>
<td>Margarita</td>
<td>Tomate, Jamón, Queso</td>
<td>6 &euro;</td>
</tr>
</table>
</body>
</html>
```



Problema:

¿significado de las marcas?
¿Procesamiento automático?

Software y estándares para la Web

Lenguajes de Marcas(VI): de SGML a XML

Grado en
Ingeniería
Informática
del Software

- **SGML** (*Standard Generalized Markup Language*)
 - *Descendiente de IBM GML de los años 1960*
 - *Utilizado para el intercambio de documentos*
 - *Principio: Separar contenido de la forma de representarlo*
 - *Permite utilizar un conjunto de marcas específico para cada aplicación*
 - *Definido como estándar ISO en 1986*
 - *XML es un subconjunto de SGML*
 - *Problema de SGML: Demasiado complicado para su adopción en la Web*
- **XML** (*eXtended Markup Language*)
 - *Desarrollado por el consorcio Web (1995)*
 - *Versión simplificada de SGML*
 - *Es un metalenguaje*
 - *Objetivos:*
 - *Estándar de intercambio de información a través de la Web*
 - *Formato abierto, independiente de la plataforma*
 - *Permite utilizar vocabularios específicos de una aplicación*
 - *Permite la auto-descripción de dichos vocabularios (documentos auto-descritos)*
 - *Las aplicaciones pueden descubrir el formato de la información y actuar en consecuencia*

Software y estándares para la Web

Lenguajes de Marcas (VII): SGML (*Standard Generalized Markup Language*)

Grado en
Ingeniería
Informática
del Software

- Estándar internacional para la definición de la estructura y el contenido de documentos
 - **SGML**: norma ISO 8879:1986(E)
- Es anterior a la Web
- Utiliza un documento tipo DTD (***D**ocument **T**ype **D**efinition*) que definen su estructura
- La diferencia de SGML con otros lenguajes de marcas es que se pueden utilizar otros caracteres sin ser “<>” para definir las etiquetas, por ejemplo el “-”
- Al ser tan genérico, es difícil de manejar

Software y estándares para la Web

Lenguajes de Marcas (VIII): XML (eXtensible Markup Language)

Grado en
Ingeniería
Informática
del Software

- **Metalenguaje** (lenguaje para definir lenguajes)
- **Extensible**: Etiquetas no predefinidas
- Diseñado para **describir datos**
- Puede ser estructurado si se usan **DTDs** para validar contenido y sintaxis
 - **Flexibilidad**: se pueden usar o no DTDs
- **Schemas** son similares a los DTD's, pero usan un formato distinto. Son útiles cuando un grupo de documentos XML comparten una serie de reglas
- <http://www.w3.org/TR/xml11/>
(Última versión 29-Septiembre-2006)



Software y estándares para la Web

Esquema

Grado en
Ingeniería
Informática
del Software

- Lenguajes de marcas
- **Introducción a XML**
- Documentos XML
- Documentos XML bien formados
- Documentos XML válidos
- Ventajas de XML
- Inconvenientes de XML
- Bibliografía
- Referencias Web
- Ejercicios propuestos

Software y estándares para la Web

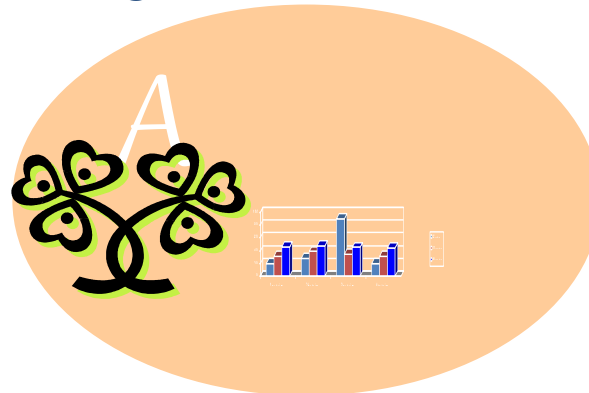
Introducción a XML (I): Concepto de documento

Grado en
Ingeniería
Informática
del Software

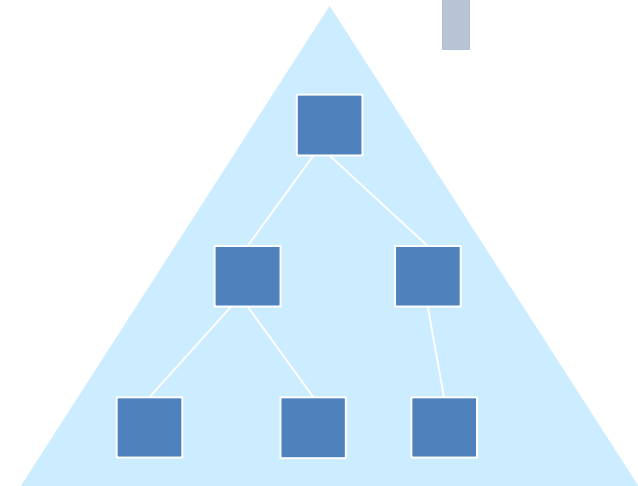
- XML especifica el contenido y la estructura, pero no la presentación

Esto es
el contenido
del documento

CONTENIDO



PRESENTACIÓN

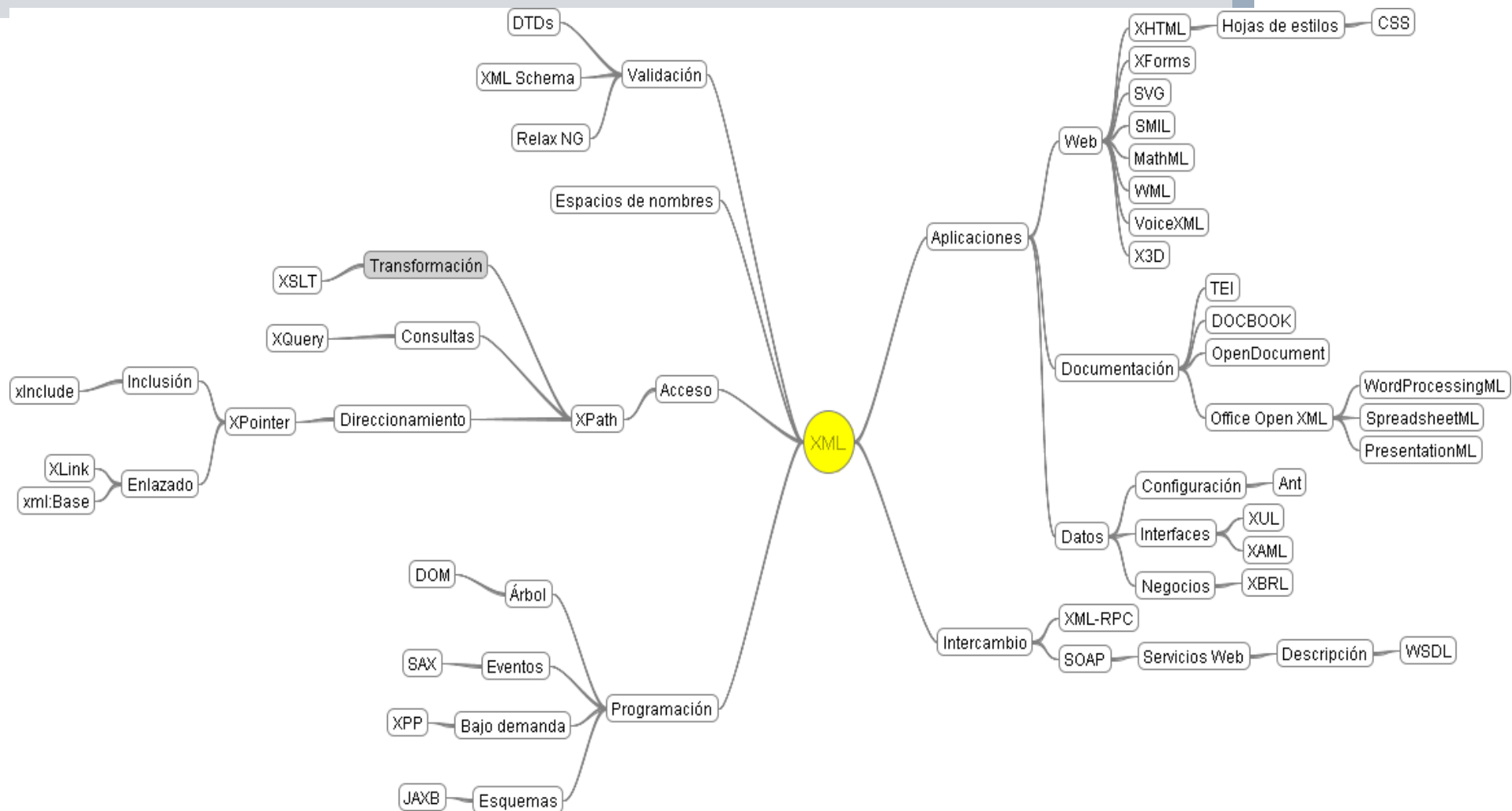


**ESTRUCTURA
EN ÁRBOL**

- Un documento XML puede ser fácilmente procesado y sus datos manipulados
- Los documentos XML permiten el **intercambio de información** entre diferentes sistemas
- Existen APIs para **procesar** esos documentos en JavaScript, Java, C#, C, C++, PHP, etc.
- XML define **datos portables** al igual que Java define código portable

Software y estándares para la Web

Introducción a XML (III): Mapa conceptual XML



XML es el pilar fundamental de muchas otras tecnologías en la Web

Software y estándares para la Web

Introducción a XML (IV): Ejemplo *pizzas.xml* (a)

Grado en
Ingeniería
Informática
del Software

pizzas.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE pizzas SYSTEM "pizzas.dtd">

<pizzas>

<pizza nombre="Barbacoa" precio="8,0€">
<ingrediente nombre="Salsa Barbacoa"/>
<ingrediente nombre="Mozzarella"/>
<ingrediente nombre="Tocineta"/>
<ingrediente nombre="Tenera"/>
</pizza>

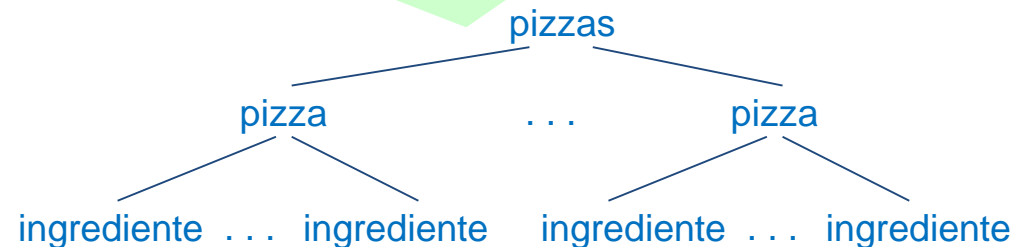
<pizza nombre="Margarita" precio="6,5€">
<ingrediente nombre="Tomate"/>
<ingrediente nombre="Jamón"/>
<ingrediente nombre="Queso"/>
</pizza>
...
</pizzas>
```

DTD = Declaración de Tipo de Documento

pizzas.dtd

```
<!ELEMENT pizzas (pizza*)>
<!ELEMENT pizza (ingrediente*)>
<!ELEMENT ingrediente (#PCDATA)>
<!ATTLIST pizza nombre CDATA #REQUIRED>
<!ATTLIST pizza precio CDATA #REQUIRED>
<!ATTLIST ingrediente nombre CDATA #REQUIRED>
```

Estructura de árbol



Las marcas tienen un significado propio de la aplicación

Software y estándares para la Web

Introducción a XML (V): Ejemplo *pizzas.xml* (b)

pizzas.xml

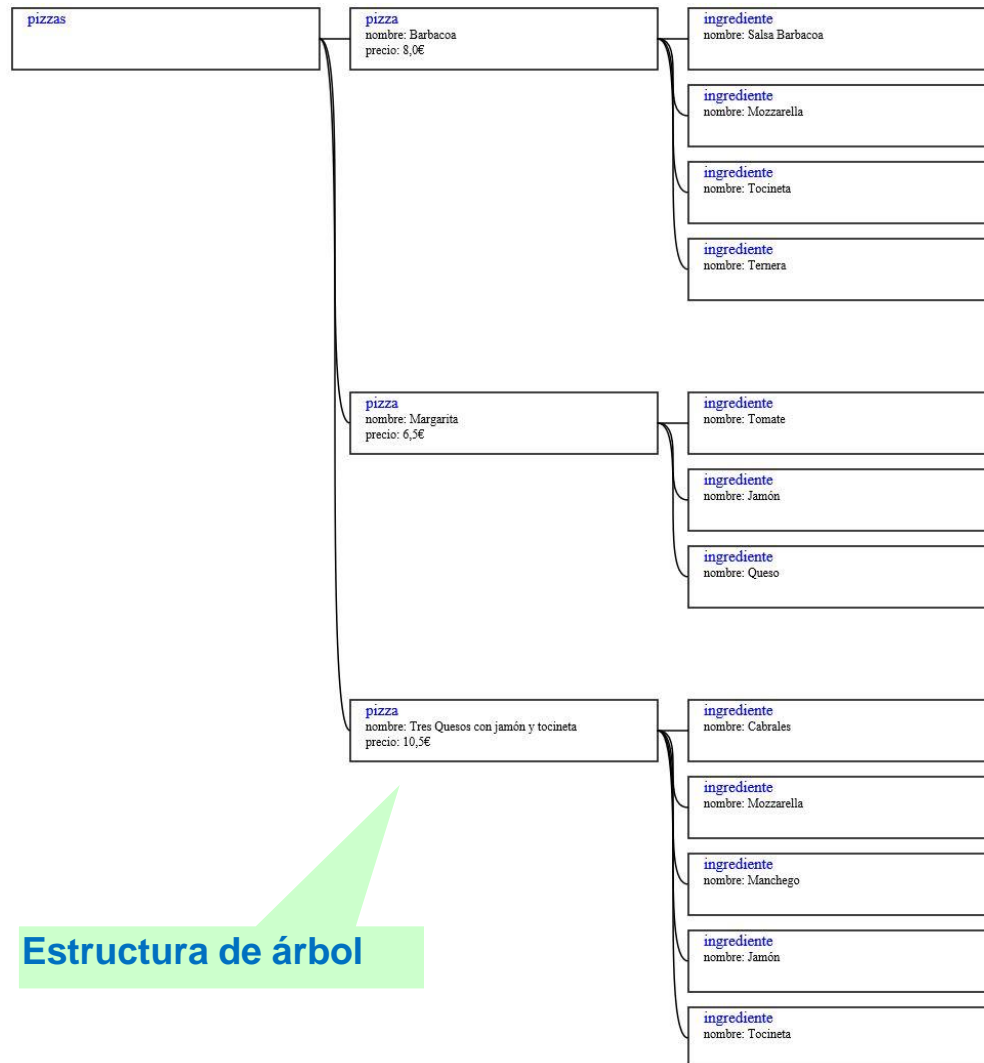
```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE pizzas SYSTEM "pizzas.dtd">
<pizzas>
  <pizza nombre="Barbacoa" precio="8,0€" >
    <ingrediente nombre="Salsa Barbacoa"/>
    <ingrediente nombre="Mozzarella"/>
    <ingrediente nombre="Tocineta"/>
    <ingrediente nombre="Tenera"/>
  </pizza>
  <pizza nombre="Margarita" precio="6,5€" >
    <ingrediente nombre="Tomate"/>
    <ingrediente nombre="Jamón"/>
    <ingrediente nombre="Queso"/>
  </pizza>
  <pizza nombre="Tres Quesos con jamón y tocineta" precio="10,5€" >
    <ingrediente nombre="Cabrales"/>
    <ingrediente nombre="Mozzarella"/>
    <ingrediente nombre="Manchego"/>
    <ingrediente nombre="Jamón"/>
    <ingrediente nombre="Tocineta"/>
  </pizza>
</pizzas>
```

Grado en
Ingeniería
Informática
del Software

Software y estándares para la Web

Introducción a XML (VI): Ejemplo *pizzas.xml* (c)

Grado en
Ingeniería
Informática
del Software



Estructura de árbol

Software y estándares para la Web

Introducción a XML (VII): Ejemplo *pizzas.dtd* (d)

Grado en
Ingeniería
Informática
del Software

- Una **declaración de tipo de documento** (*Document Type Definition, DTD*) define la estructura de los elementos y atributos permitidos en un documento XML
- Un DTD puede estar incrustado en el documento XML o en un archivo aparte
- DTD externo:

pizzas.dtd

```
<!ELEMENT pizzas (pizza*)>  
<!ELEMENT pizza (ingrediente*)>  
<!ELEMENT ingrediente (#PCDATA)>  
<!ATTLIST pizza nombre CDATA #REQUIRED>  
<!ATTLIST pizza precio CDATA #REQUIRED>  
<!ATTLIST ingrediente nombre CDATA #REQUIRED>
```

Software y estándares para la Web

Introducción a XML (VIII): Ejemplo *alba.xml* (a)

Grado en
Ingeniería
Informática
del Software

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE poema SYSTEM "poema.dtd">
<poema autor="Federico García Lorca" fecha="Abril 1915" lugar="Granada" >
<titulo>Alba</titulo>
<verso>Mi corazón oprimido</verso>
<verso>siente junto a la alborada</verso>
<verso>el dolor de sus amores</verso>
<verso>Y el sueño de las distancias.</verso>
<verso>La luz de la aurora lleva</verso>
<verso>Semilleros de nostalgias</verso>
<verso>Y la tristeza sin ojos</verso>
<verso>De la médula del alma.</verso>
<verso>La gran tumba de la noche</verso>
<verso>Su negro velo levanta</verso>
<verso>Para ocultar con el día</verso>
<verso>La inmensa cumbre estrellada.</verso>
</poema>
```

alba.xml

Software y estándares para la Web

Introducción a XML (IX): Ejemplo *poema.dtd* (b)

poema.dtd

Grado en
Ingeniería
Informática
del Software

```
<!ELEMENT poema (título, verso+)>  
<!ELEMENT título (#PCDATA)>  
<!ELEMENT verso (#PCDATA)>  
<!ATTLIST poema  
    autor CDATA #REQUIRED  
    fecha CDATA #REQUIRED  
    lugar CDATA #IMPLIED>
```

Opcional

Software y estándares para la Web

Introducción a XML (X): Árbol *alba.xml* (c)

Grado en
Ingeniería
Informática
del Software



Software y estándares para la Web

Introducción a XML (XI): Ejemplo *libros.xml* (a)

libros.xml

Grado en
Ingeniería
Informática
del Software

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE libros SYSTEM "libros.dtd">
<libros>
  <libro isbn="9788420633114">
    <título>El Aleph</título>
    <autor>Jorge Luis Borges</autor>
    <año>1946</año>
    <precio moneda="Euro">7,80</precio>
    <editorial>Alianza Editorial</editorial>
    <clasificación>Literatura</clasificación>
    <idioma>Español</idioma>
  </libro>
```


Software y estándares para la Web

Introducción a XML (XII): Ejemplo *libros.xml* (b)

libros.xml (continuación)

Grado en
Ingeniería
Informática
del Software

```
<libro isbn="9780470036662">
<título>Domain-Specific Modeling. Enabling full code generation</título>
<autor>Steven Kelly</autor>
<autor>Juha-Pekka Tolvanen</autor>
<año>2008</año>
<precio moneda="Dolar USA">74,03</precio>
<editorial>Wiley</editorial>
<clasificación>Informática</clasificación>
<idioma>Inglés</idioma>
</libro>

</libros>
```

Software y estándares para la Web

Introducción a XML (XIII): Ejemplo *libros.dtd* (c)

libros.dtd

```
<!ELEMENT libros (libro+)>
<!ELEMENT libro (título, autor+, año?, precio, editorial, clasificación, idioma)>
<!ELEMENT título (#PCDATA)>
<!ELEMENT autor (#PCDATA)>
<!ELEMENT año (#PCDATA)>
<!ELEMENT precio (#PCDATA)>
<!ELEMENT editorial (#PCDATA)>
<!ELEMENT clasificación (#PCDATA)>
<!ELEMENT idioma (#PCDATA)>
<!ATTLIST libro isbn ID #REQUIRED>
<!ATTLIST precio moneda CDATA #REQUIRED>
```

Grado en
Ingeniería
Informática
del Software

Software y estándares para la Web

Introducción a XML (XIV): Árbol *libros.xml* (d)

Grado en
Ingeniería
Informática
del Software



Software y estándares para la Web

Introducción a XML (XV): Algunos lenguajes derivados de XML

Grado en Ingeniería Informática del Software

- MathML
 - Visualización de ecuaciones matemáticas
- SVG
 - Gráficos vectoriales
- SMIL
 - Presentaciones multimedia
- P3P
 - Descripción de características de privacidad
- WML
 - Similar a HTML para teléfonos móviles (obsoleto)
- VoiceML
 - Portales basados en voz
- XML Signature
 - • Firma de recursos Web
- XKMS
 - Firmas y criptografía
- XML Query
 - Consultas de documentos
- XBRL
 - Contabilidad
- ebXML
 - Negocios electrónicos (e-business)
- SyncXML
 - Sincronización de dispositivos
- UPnP
 - Plug and Play universal
- KML, KMZ
 - Información geográfica (Google Earth, Google Maps, ...)

Software y estándares para la Web

Introducción a XML (XVI): KML

Grado en
Ingeniería
Informática
del Software

- KML es un formato de archivo que se utiliza para mostrar **datos geográficos** usando software, como **Google Earth**, Google Maps y Google Maps para móviles.
- KML utiliza una estructura basada en etiquetas con atributos y elementos anidados y está **basado en el estándar XML**
- **KMZ** es un archivo KML empaquetado en usando ZIP, el resultado es un archivo en binario

Software y estándares para la Web

Introducción a XML (XVII): Oviedo.kml

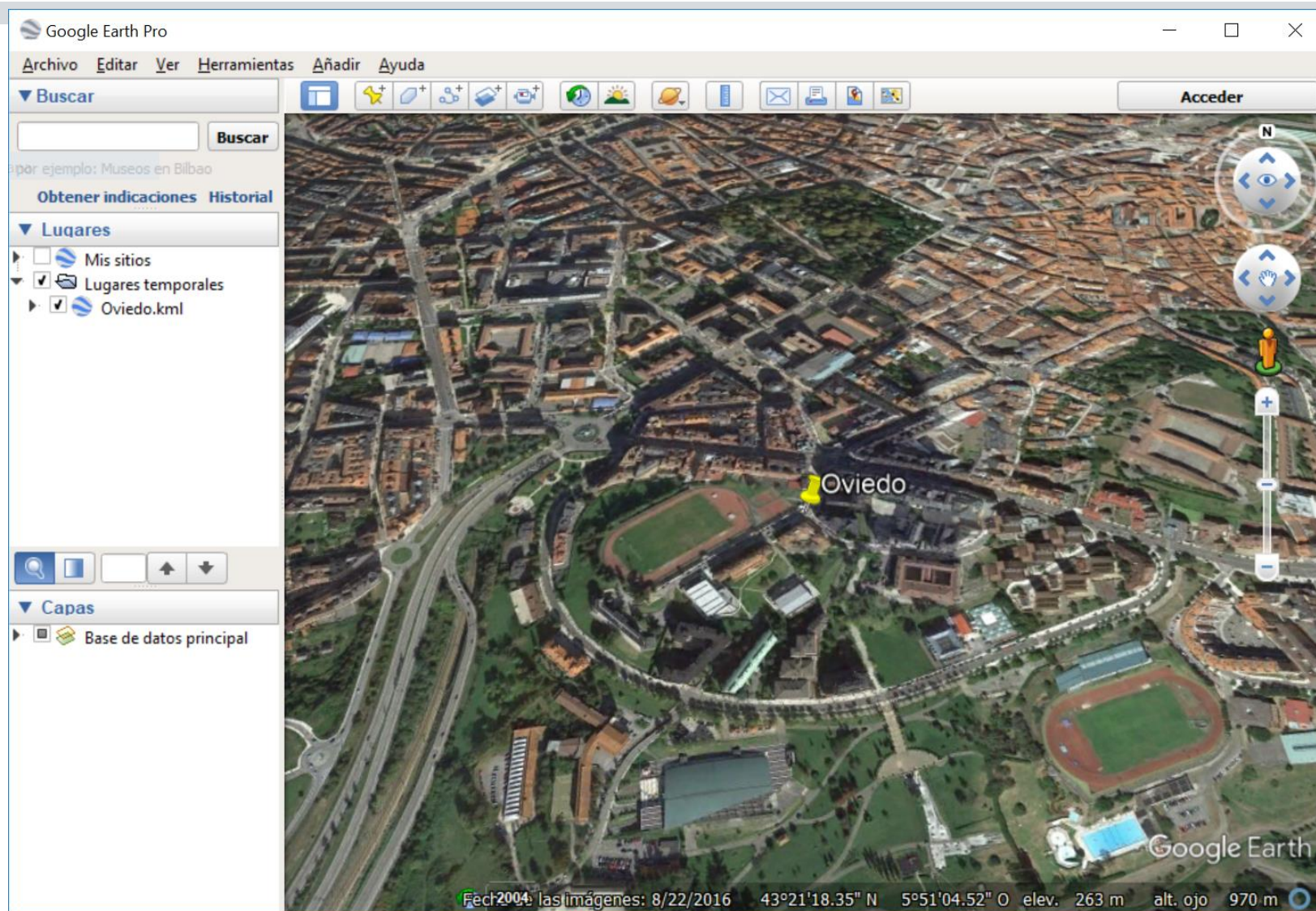
Scheme = Validación con URI

Grado en
Ingeniería
Informática
del Software

```
<?xml version="1.0" encoding="UTF-8"?>
<kml xmlns="http://www.opengis.net/kml/2.2">
<Document id='Universidad de Oviedo'>
<Placemark>
<name>Oviedo</name>
<description>Escuela de Ingeniería Informática</description>
<Point>
<coordinates>
-5.851256792487727,43.355098167157855
</coordinates>
<altitudeMode>relativeToGround</altitudeMode>
</Point>
</Placemark>
</Document>
</kml>
```


Software y estándares para la Web

Introducción a XML (XVIII): Oviedo.kml en Google Earth

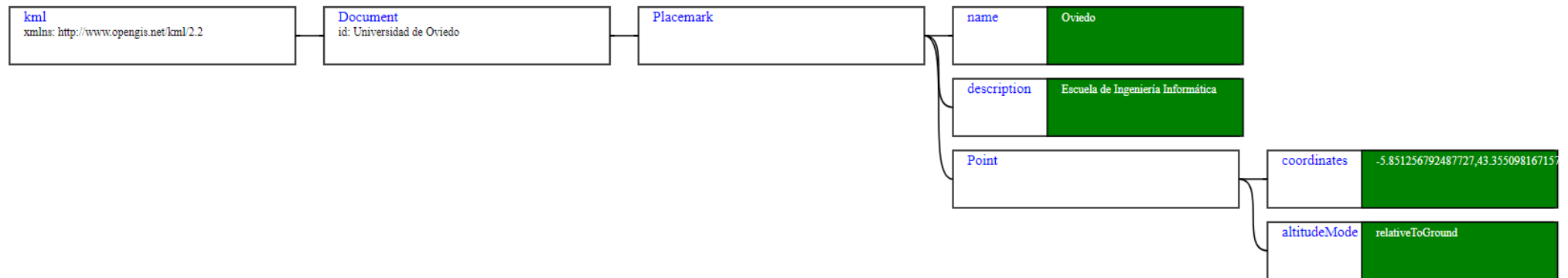


Grado en
Ingeniería
Informática
del Software

Software y estándares para la Web

Introducción a XML (XIX): Árbol de Oviedo.kml

Grado en
Ingeniería
Informática
del Software



Software y estándares para la Web

Esquema

Grado en
Ingeniería
Informática
del Software

- Lenguajes de marcas
- Introducción a XML
- **Documentos XML**
- Documentos XML bien formados
- Documentos XML válidos
- Ventajas de XML
- Inconvenientes de XML
- Bibliografía
- Referencias Web
- Ejercicios propuestos

- *“Información jerarquizada, en forma de texto, que constituye un objeto de datos que puede ser presentado mediante una estructura de árbol, que puede estar almacenado en un único archivo o dividido en varios”*
- Editores para crear archivos XML:
 - Editores de texto: **Bloc de notas, Notepad++, Brackets, Eclipse, Sublime,...**
 - Editor especializado: **XMLSpy**

Documento XML = datos + marcado

Software y estándares para la Web

Documentos XML (II): Partes de un documento XML

Grado en
Ingeniería
Informática
del Software

- **Prólogo**
 - Declaración XML
 - Declaración del Tipo de Documento (DTD) o Scheme
- **Cuerpo**
 - Elementos
 - Atributos
 - Entidades predefinidas
 - Instrucciones de proceso
 - Secciones CDATA
- **Comentarios:**
 - Se pueden introducir en cualquier lugar del cuerpo o del prólogo, pero nunca dentro de las declaraciones, etiquetas u otros comentarios.
 - Dentro de los comentarios no pueden aparecer los caracteres --

`<!-- Texto de comentario-->`


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

- **version (obligatoria):** versión de XML usada en el documento
 - 1.0 = versión más habitual
 - 1.1 aumenta capacidad de soporte de Unicode
- **encoding (opcional):** la forma en que se ha codificado el documento.
 - Por defecto es UTF-8, aunque podrían ponerse otras, como UTF-16, US-ASCII, ISO-8859-1, etc.
 - UTF-8= caracteres Unicode
 - iso-8859-1 = caracteres latinos
- **standalone (opcional)**
 - Indica al procesador XML si un documento es independiente, es decir no usa fuentes externas (standalone="no")
 - Indica que se basa en información de fuentes externas, es decir, si depende de declaraciones de marca externas como una DTD externa (standalone="yes").

Software y estándares para la Web

Documentos XML (IV): Prólogo. Declaración del Tipo de Documento

Grado en
Ingeniería
Informática
del Software

```
<!DOCTYPE nombre SYSTEM|PUBLIC uri>
```

- Declara una DTD que se quiera incorporar al documento
- **DOCTYPE**: indica que la etiqueta contiene una DTD
- **nombre**: declara el nombre de la DTD
- **PUBLIC**: indica que la DTD es pública y está disponible
- **SYSTEM**: La DTD no es pública
- **URI** (*Identificadores únicos de recursos*)
 - URI = URL + URN
 - URL (*Locator*) tiene doble funcionalidad:
 - Identificar recursos
 - Protocolo de acceso
 - Ejemplo: `http://www.uniovi.es`
 - URN: Nombre único de recursos
 - Ejemplo: `urn:isbn:0-395-36341-1`

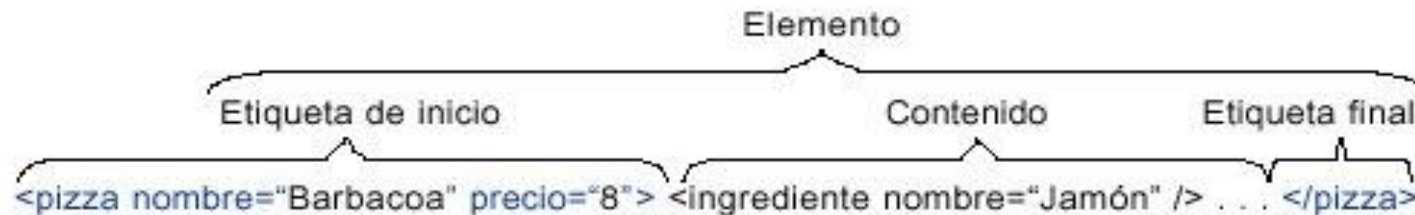
```
<!DOCTYPE pizzas SYSTEM "pizzas.dtd">
```

Software y estándares para la Web

Documentos XML (V): Cuerpo. Elementos (a)

Grado en
Ingeniería
Informática
del Software

- El cuerpo del documento está formado por un elemento
- Un elemento está formado por:
 - Una **etiqueta inicial** (nombre entre signos < y >): `<etiqueta>`
 - La etiqueta inicial puede contener una lista **de atributos y valores**:
`<etiqueta atributo="valor">`
 - Los contenidos del elemento (puede estar vacío)
 - El elemento debe acabar con **una etiqueta final** con el mismo nombre
 - El **contenido** del elemento es todo lo que hay entre la etiqueta inicial y la final
 - El contenido pueden ser otros elementos



- En caso de un elemento vacío puede usarse la sintaxis: `<etiqueta />`

`<ingrediente nombre="Jamón" calorías="8"></ingrediente>`

|| (es equivalente a)

`<ingrediente nombre="Jamón" calorías="8" />`

- Es necesario cerrar todas las etiquetas
- XML es sensible a mayúsculas/minúsculas

- Sintaxis de los elementos:
`<NombreElemento>Contenido</NombreElemento>`
- Representación de elementos (llamados “*singletons*”) vacíos:
`<NombreElemento/>`
- Anidamiento de elementos jerárquico

```
<Alumno>  
  <NombreAlumno>Juan Manuel</NombreAlumno>  
  <Edad>18</Edad>  
  <Matriculadas>80</Matriculadas>  
</Alumno>
```

Convención para nombres

- Consisten de uno o más caracteres sin espacios en el medio.
 - Solo puede comenzar con una letra o con un “_”
 - Los siguientes caracteres pueden ser cualquiera del estándar “Unicode”.
- Sensibles a mayúsculas/minúsculas
- **Ejemplos:**
 - *Válidos*
 - “Juan”
 - “_Juan”
 - “Juan1234”
 - “Juanß”
 - *No válidos*
 - “-Juan”
 - “1234”
 - “Juan 1234”

Software y estándares para la Web

Documentos XML (VIII): Cuerpo. Anidamiento

Grado en
Ingeniería
Informática
del Software

- Las etiquetas que se abran deben cerrarse sin que se produzcan anidamientos

- *Correcto*

```
<externo>  
  <interno> texto </interno>  
  
</externo>
```

- *Incorrecto*

```
<externo>  
  <interno> texto </externo>  
</interno>
```

Software y estándares para la Web

Documentos XML (IX): Cuerpo. Atributos

Grado en
Ingeniería
Informática
del Software

- Los valores de los atributos se escriben entre comillas dobles o simples.
- El orden de atributos no es significativo
- No puede haber nombres de atributos repetidos
- Pueden incluirse valores entrecomillados siempre que las comillas sean diferentes de las externas.
 - Ejemplo: frase="Diego dijo 'digo' "

```
<Matriculadas unidad="créditos">80</Matriculadas>
```

- Hay varios atributos predefinidos.
- Por ejemplo:
 - **xml:lang** especifica el código del idioma: **en** (inglés), **sp** (español), etc.
 - **xml:space** especifica cómo tratar el espacio en blanco:
 - **preserve** = mantenerlo
 - **default** = dejar libertar a la aplicación para tratarlo como quiera
 - **xml:base** define la URL que identifica las direcciones relativas

Software y estándares para la Web

Documentos XML (XI): Ejemplo de atributos predefinidos

Grado en
Ingeniería
Informática
del Software

```
<p xml:lang="en">The quick brown fox jumps over the lazy dog.</p>
<p xml:lang="en-GB">What colour is it?</p>
<p xml:lang="en-US">What color is it?</p>
<p xml:lang="sp">¿Qué color es?</p>
<sp who="Faust" desc='leise' xml:lang="de">
  <l>Habe nun, ach! Philosophie,</l>
  <l>Juristerei, und Medizin</l>
  <l>und leider auch Theologie</l>
  <l>durchaus studiert mit heißem Bemüh'n.</l>
</sp>
```

Software y estándares para la Web

Documentos XML (XII): Cuerpo. Entidades predefinidas

Grado en
Ingeniería
Informática
del Software

Entidad	Carácter
&	&
<	<
>	>
'	'
"	"

- Cualquier carácter Unicode puede indicarse mediante & seguido del número y acabado por ;

Software y estándares para la Web

Documentos XML (XIII): Cuerpo. Instrucciones de proceso

Grado en
Ingeniería
Informática
del Software

- Mecanismo que permite a los documentos XML contener instrucciones específicas para las aplicaciones que los van a usar, sin que éstas formen parte de los datos del propio documento
- Ejemplos:

- `<?xml version='1.0' ?>`
- `<?xml-stylesheet type= "text/xsl" href= "MiHojaDeEstilo.xsl"?>`
- `<?php ... ?>`

- Construcción en XML que permite **especificar datos, utilizando cualquier carácter, especial o no, sin que se interprete como marcado XML**
- La razón de esta construcción es que a veces es necesario para los autores de documentos XML, **poder leerlo fácilmente** sin tener que descifrar los códigos de entidades.
- Por ejemplo con **código fuente** de lenguajes de programación.

Ejemplo **sin** CDATA

```
<codigo>
if ( x &lt; 3 &amp;&amp; x &gt; 4)
    printf (&quot; Hola&quot; ) ;
</codigo>
```

Ejemplo **con** CDATA

```
<codigo>
<![CDATA[
if ( x < 3 && x > 4)
    printf ( " Hola " ) ;
] ]>
</codigo>
```

Software y estándares para la Web

Esquema

Grado en
Ingeniería
Informática
del Software

- Lenguajes de marcas
- Introducción a XML
- Documentos XML
- **Documentos XML bien formados**
- Documentos XML válidos
- Ventajas de XML
- Inconvenientes de XML
- Bibliografía
- Referencias Web
- Ejercicios propuestos

Software y estándares para la Web

Documentos XML bien formados (I)

- **Documento bien formado**

- Sigue las reglas sintácticas de XML
- No se comprueba su validez con un DTD o Schema
- Importante:
 - Contiene un único elemento raíz
 - Todas las etiquetas están correctamente anidadas

```
<pizzas>
  <pizza nombre="Margarita" precio="6">
    <ingrediente nombre="Tomate" />
    <ingrediente nombre="Queso" />
  </pizza>
</pizzas>
```

```
<pizzas>
  <pizza nombre="Margarita" precio="6">
    <ingrediente nombre="Tomate" >
  </pizzas>
```

- El documento puede contener varias **instrucciones de procesamiento**
 - Indican cómo debe procesarse el documento

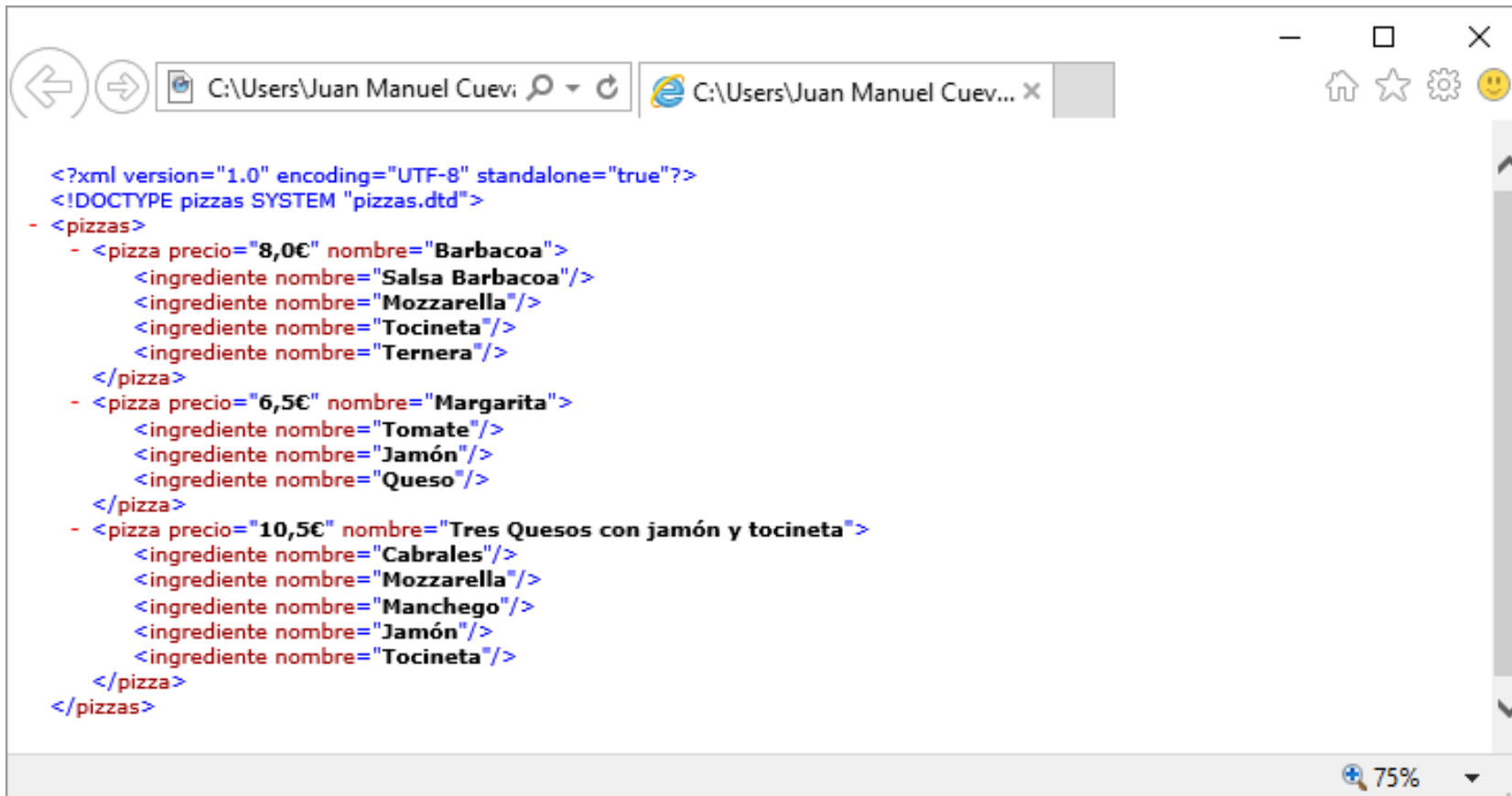
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>

Software y estándares para la Web

Documentos XML bien formados (II)

- Los navegadores comprueban si un documento XML está bien formado. **Bien formado.**

Grado en
Ingeniería
Informática
del Software



```
<?xml version="1.0" encoding="UTF-8" standalone="true"?>
<!DOCTYPE pizzas SYSTEM "pizzas.dtd">
- <pizzas>
  - <pizza precio="8,0€" nombre="Barbacoa">
    <ingrediente nombre="Salsa Barbacoa"/>
    <ingrediente nombre="Mozzarella"/>
    <ingrediente nombre="Tocineta"/>
    <ingrediente nombre="Ternera"/>
  </pizza>
  - <pizza precio="6,5€" nombre="Margarita">
    <ingrediente nombre="Tomate"/>
    <ingrediente nombre="Jamón"/>
    <ingrediente nombre="Queso"/>
  </pizza>
  - <pizza precio="10,5€" nombre="Tres Quesos con jamón y tocineta">
    <ingrediente nombre="Cabrales"/>
    <ingrediente nombre="Mozzarella"/>
    <ingrediente nombre="Manchego"/>
    <ingrediente nombre="Jamón"/>
    <ingrediente nombre="Tocineta"/>
  </pizza>
</pizzas>
```

- Ejemplo
 - **Documento XML mal formado.**
 - No se ha cerrado la etiqueta <pizza>

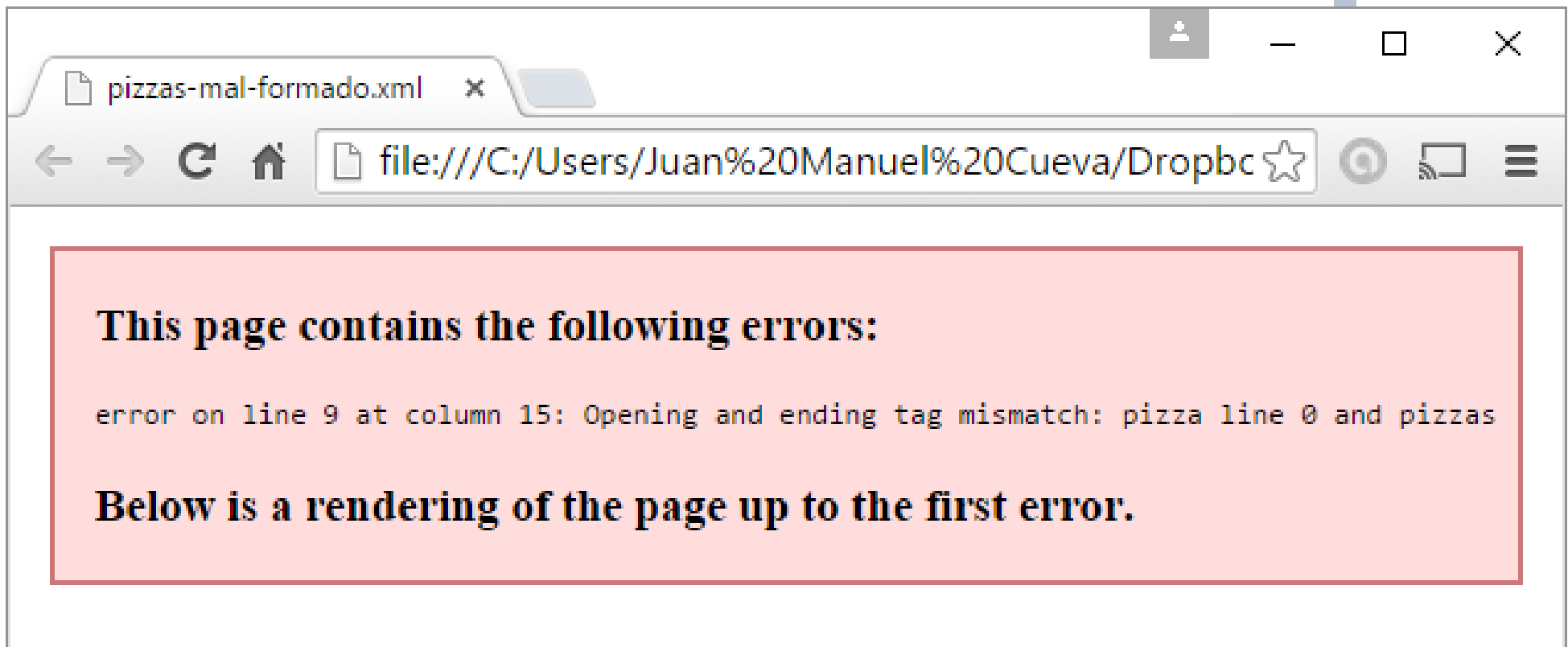
```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<!DOCTYPE pizzas SYSTEM "pizzas.dtd">
<pizzas>
<pizza precio="8" nombre="Barbacoa">
<ingrediente nombre="Salsa Barbacoa"/>
<ingrediente nombre="Mozzarella"/>
<ingrediente nombre="Tocineta"/>
<ingrediente nombre="Ternera"/>
</pizzas>
```

Software y estándares para la Web

Documentos XML mal formados (II)

Grado en
Ingeniería
Informática
del Software

- Los navegadores detectan que un documento XML está **mal formado**.



Software y estándares para la Web

Esquema

Grado en
Ingeniería
Informática
del Software

- Lenguajes de marcas
- Introducción a XML
- Documentos XML
- Documentos XML bien formados
- **Documentos XML válidos**
- Ventajas de XML
- Inconvenientes de XML
- Bibliografía
- Referencias Web
- Ejercicios propuestos

Software y estándares para la Web

Documentos XML válidos (I)

Grado en
Ingeniería
Informática
del Software

- Se puede incluir una declaración del tipo de documento (DTD)

pizzas.xml

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE pizzas SYSTEM "pizzas.dtd">
<pizzas>
  <pizza nombre="Margarita" precio="6">
    <ingrediente nombre="Tomate" />
  </pizza>
</pizzas>
```

pizzas.dtd

```
<!ELEMENT pizzas (pizza*)>
<!ELEMENT pizza (ingrediente*)>
<!ELEMENT ingrediente (#PCDATA)>
<!ATTLIST pizza nombre CDATA #REQUIRED>
<!ATTLIST pizza precio CDATA #REQUIRED>
<!ATTLIST ingrediente nombre CDATA #REQUIRED>
```

- Documento XML válido**
 - Está bien formado y
 - La estructura encaja con la declaración del tipo de documento (DTD)

- Es posible validar la estructura de los documentos utilizando diversas alternativas:
 - DTDs
 - XML Schema
 - <http://www.w3.org/XML/Schema>
 - Relax NG
 - <http://relaxng.org/>
 - Schematron
 - <http://www.schematron.com/>

Software y estándares para la Web

Esquema

Grado en
Ingeniería
Informática
del Software

- Lenguajes de marcas
- Introducción a XML
- Documentos XML
- Documentos XML bien formados
- Documentos XML válidos
- **Ventajas de XML**
- Inconvenientes de XML
- Bibliografía
- Referencias Web
- Ejercicios propuestos

Software y estándares para la Web

Ventajas de XML

**Grado en
Ingeniería
Informática
del Software**

- Es un formato estructurado
- Contiene información y meta-información
 - Ha sido diseñado específicamente para Internet
- Soportado por visualizadores y servidores
 - Numerosas herramientas de procesamiento
 - Legible por personas humanas (fichero de texto)
 - Admite la definición de vocabularios específicos
 - Separa contenido del procesamiento y visualización
 - Aumenta la seguridad mediante la validación de documentos
 - Formato abierto, respaldado por numerosas organizaciones
 - Una vez definido un DTD o un esquema XML común, facilita intercambio de información

Software y estándares para la Web

Esquema

Grado en
Ingeniería
Informática
del Software

- Lenguajes de marcas
- Introducción a XML
- Documentos XML
- Documentos XML bien formados
- Documentos XML válidos
- Ventajas de XML
- **Inconvenientes de XML**
- Bibliografía
- Referencias Web
- Ejercicios propuestos

Software y estándares para la Web

Inconvenientes de XML(I)

- Puede requerir demasiado espacio, ancho de banda y tiempo de procesamiento
 - Documentos largos con mucha información redundante
- Es una sintaxis de documentos, no un lenguaje de programación

```
int main(void) {  
    printf("Hola");  
    return 0;  
}
```

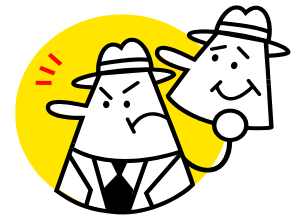
```
<function name="main" type="int">  
  <arg type="void" />  
  <call function="printf">  
    <param>Hola</param>  
  </call>  
  <return value="0"/>  
</function>
```

- Es posible crear formatos y vocabularios propietarios
- Puede fomentar la proliferación de vocabularios específicos
- Bueno para texto, malo para datos binarios

```
<?xml version="1.0">  
<imagen formato="base64">  
DS34JSCDF029876D76523981DFNDF3F2134F5FD019A  
FGF23DAND345CD2135911943DCBKAPFGDAJJK32A10  
....  
</imagen>
```

- Poco eficiente como lenguaje de almacenamiento de bases de datos

Grado en
Ingeniería
Informática
del Software

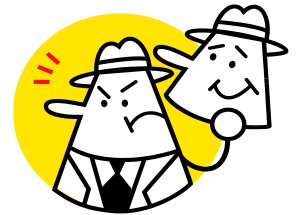
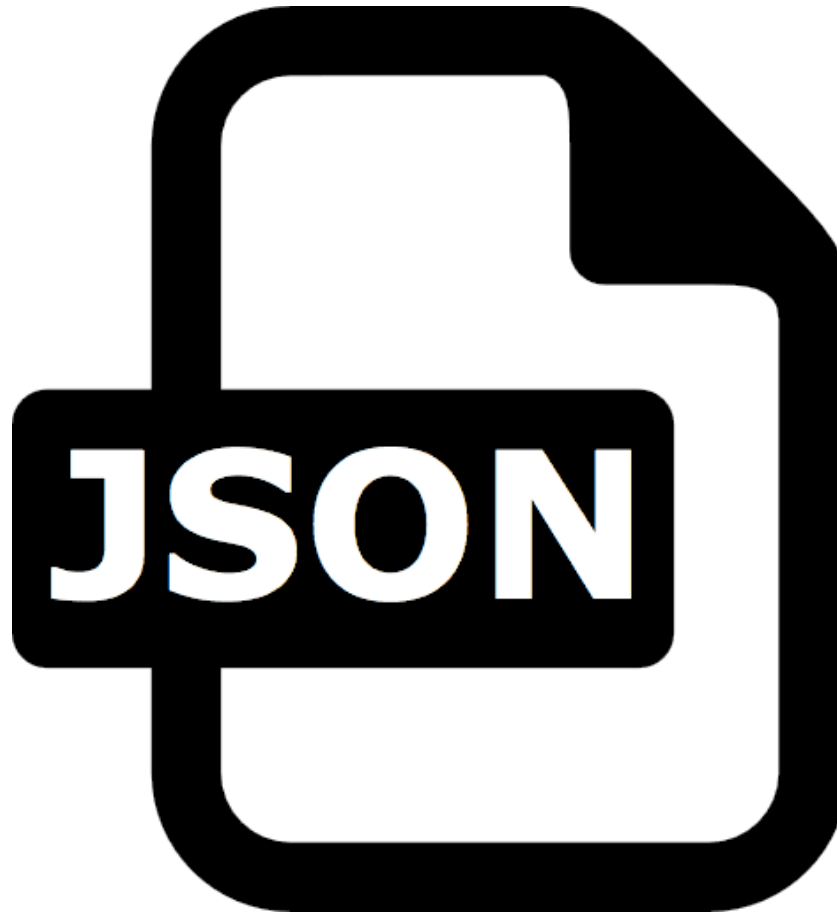


Software y estándares para la Web

Inconvenientes de XML(II)

- Está siendo sustituido en muchas aplicaciones por JSON (JavaScript Object Notation)

Grado en
Ingeniería
Informática
del Software



Software y estándares para la Web

Esquema

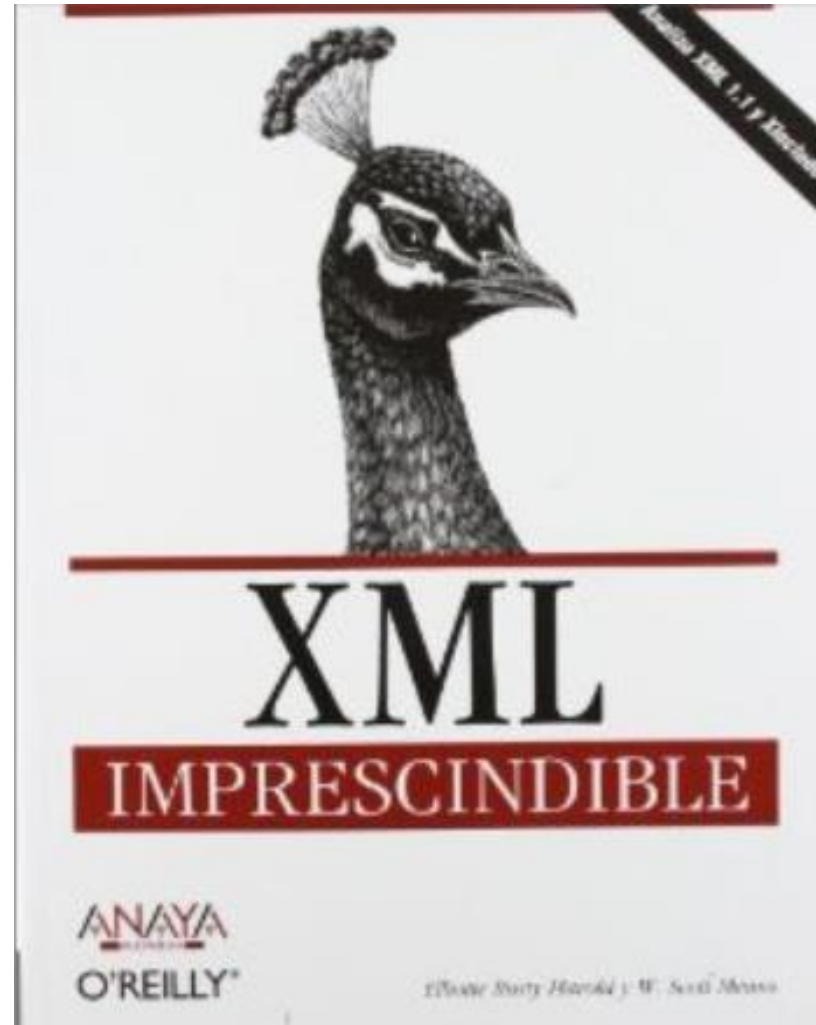
Grado en
Ingeniería
Informática
del Software

- Lenguajes de marcas
- Introducción a XML
- Documentos XML
- Documentos XML bien formados
- Documentos XML válidos
- Ventajas de XML
- Inconvenientes de XML
- **Bibliografía**
- Referencias Web
- Ejercicios propuestos

Software y estándares para la Web

Bibliografía (I)

- Libro recomendado de lectura y consulta:
 - “**XML imprescindible**”
 - ANAYA/O'Reilly (2005)
 - E. Rusty Harold y W. Scott Means



Grado en
Ingeniería
Informática
del Software

Software y estándares para la Web

Bibliografía (II)

- Libro recomendado de consulta:
 - “**Beginning XML**”
 - John Wiley & Sons (2012)
 - Joe Fawcett, Liam R.E. Quin, and Danny Ayers



Grado en
Ingeniería
Informática
del Software

Software y estándares para la Web

Esquema

Grado en
Ingeniería
Informática
del Software

- Lenguajes de marcas
- Introducción a XML
- Documentos XML
- Documentos XML bien formados
- Documentos XML válidos
- Ventajas de XML
- Inconvenientes de XML
- Bibliografía
- **Referencias Web**
- Ejercicios propuestos

Software y estándares para la Web

Referencias Web

Grado en
Ingeniería
Informática
del Software

- <http://www.w3.org/TR/xml11/>
 - Especificación del estándar XML en W3C
 - Última versión 29-Septiembre-2006
- <http://www.w3schools.com/xml>
 - Tutoriales on-line de XML
- <https://developers.google.com/kml/documentation/>
 - Documentación de KML
- <http://json.org/>
 - Especificación del estándar JSON

Software y estándares para la Web

Esquema

Grado en
Ingeniería
Informática
del Software

- Lenguajes de marcas
- Introducción a XML
- Documentos XML
- Documentos XML bien formados
- Documentos XML válidos
- Ventajas de XML
- Inconvenientes de XML
- Bibliografía
- Referencias Web
- **Ejercicios propuestos**

Software y estándares para la Web

Ejercicios propuestos (I)

Grado en
Ingeniería
Informática
del Software

- Construir un **documento XML bien formado** para contener artículos de revista con los siguientes requisitos mínimos:
 - Título del artículo
 - Autores y su correo electrónico
 - Resumen
 - Palabras clave
 - Nombre de la revista
 - Número o volumen de la revista
 - Página de inicio del artículo
 - Página final del artículo
 - Año

Software y estándares para la Web

Ejercicios propuestos (II)

Grado en
Ingeniería
Informática
del Software

- Construir un **documento XML bien formado** para contener recetas de cocina con los siguientes requisitos mínimos:
 - Nombre de la receta (por ejemplo “Fabada Asturiana”)
 - Tipo de plato (postre, primer plato, entrante,. . .)
 - Ingredientes con cantidades (por ejemplo “Fabas 500 gramos”)
 - Calorías del plato (opcional)
 - Proceso de elaboración, especificado en pasos, por ejemplo:
 - Paso 1: Poner *les fabes* a remojo la noche anterior
 - Paso 2: Poner *les fabes* a cocer con agua y laurel
 - Paso 3: etc...
 - Dificultad del proceso de elaboración (por ejemplo “Fácil”, “Medio”, “Difícil”,...)
 - Tiempo de elaboración (por ejemplo “45 minutos”)
 - Elementos utilizados para la elaboración (microondas, wok, horno, freidora,. . .)
 - Origen de la receta (por ejemplo “Receta de mi abuela”, “Libro de M^a Luisa”, “Libro de las 1001 recetas”, “www.recetasMUYricas.com”)