# HyperText Markup Language

Jesús Vegas

Departamento de Informática
Universidad de Valladolid
España
jvegas@infor.uva.es

# Orígenes de HTML

- '60: Ted Nelson introduce el término *hipertexto* en su sistema Xanadu.
- 87: Bill Atkinson en Apple crea las *HyperCards*, tarjetas que pueden contener texto, sonido, video o figuras con botones que el usuario podía utilizar para navegar por ellas (*HyperTalk*).
- Tim Berners-Lee, CERN, desarrollo el WWW como una BD distribuida en la que se utiliza el hipertexto como mecanismo de consulta en la que cada usuario se encarga de administrar su parte de la BD (protocolos URL, HTTP y HTML).

## CREACIÓN DE HTML

- Tim Berners-Lee elige SGML para implementar HTML haciendole independiente de SO y lenguaje.
- Los científicos podían crear el contenido, añadirle las etiquetas apropiadas y hacer que el documento esté disponible para el resto de la comunidad a través de la red.
- 1993: se propone crear un consorcio que trate los problemas de establecimiento de estándares para HTML y HTTP: WWW Consortium (`http://www.w3c.org`) .

# Estructura de un documento

- HTML trata de definir la estructura de los documentos, aunque también tenga alguna posibilidad de modificar su apariencia.
- Se puede definir la estructura de un documento como la relación que mantienen las partes del mismo entre ellas.
- Esto constrasta con el uso de los procesadores de texto que consideran sólo la apariencia de los documentos, y casi nunca su estructura.

# Terminología HTML

- **DTD**: reglas escritas en SGML que define la sintaxis y la estructura de un tipo de documento dado, como por ejemplo HTML.
- **elemento**: cada componente de la estructura de un documento. Está asociado a una etiqueta.
- **etiqueta**: un cógigo que se inserta en el contenido del documento para determinar la estructura del mismo (`<TAG>` y `</TAG>`).
- **atributo**: aporta información adicional que modifica el comportamiento de las etiquetas (`atributo="valor"`).
- **browser**: un parser que evalua las etiquetas y el contenido de un documento HTML y lo muestra conforme a las posibilidades del sistema. También se encarga de solicitar nuevos documentos según HTTP.

# Un Documento HTML

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0//EN">
<HTML>
   <HEAD>
   <!-- esto es la cabecera del documento -->
   </HEAD>
   <BODY>
   <!-- este es el cuerpo del documento -->
   </BODY>
</HTML>
```

## La Declaración <!DOCTYPE>

- Es la primera etiqueta que debiera aparecer en un documento HTML.
- `<!DOCTYPE` *identificador* `>` donde el identificador determina la DTD utilizada para construir el documento, el conjunto de etiquetas válidas y su relación.
- Como existen muchar versiones que evolucionan rápidamente, muchos browsers ignoran esta declaración y las etiquetas que no entienden, aunque sean parte del estándar declarado.

# La Etiqueta <HTML>

- Marca el comienzo y el final del documento.
- En la práctica este par de etiquetas puede parecer opcional, sobre todo la de cierre, pues muchos browsers admiten documentos sin ella, pero esto puede confundir a otros browsers que rechazarán la página Web o la mostrarán de manera impredecible.
- Cualquier etiqueta del documento debe aparecer entre el par de etiquetas `<HTML>` y `</HTML>`.
- Dentro de una etiqueta `<HTML>` puede aparecer la etiqueta `<HEAD>`, seguida de `<BODY>` o `<FRAMESET>`.

## La Etiqueta <HEAD>

Contiene las definiciones globales a todo el documento:

- El título del documento: `<TITLE> título </TITLE>` que será mostrado en la barra de título del browser, por lo que deben ser cortos y representativos.
- El destino por defecto de los hiperenlaces: `<BASE href="protocol://servername/path/" [target=targetFrame]>`.
- Relaciones entre este documento y otros documentos Web para determinar un mismo modo de impresión, hojas de estilo o scripts: `<LINK>`.
- El modo en que el documento debe ser tratado o información utilizable por motores de búsqueda y otros: `<META>`.

# La Etiqueta <META> I

`<META HTTP-EQUIV=valor CONTENT=contenido>`

- `refresh` indica al browser que la página debe ser actualizada después de que pasen los segundos almacenados en `contenido` que puede tomar la forma `N;URL=url`.
- `keywords` especifica una lista de palabras clave almacenadas en `contenido` que clasifican el documento.
- `reply-to` provee una dirección de correo electrónico para ponerse en contacto con el autor de la página Web.

# La Etiqueta <META> II

`<META NAME=valor CONTENT=contenido>`

- `author` identifica el autor que creó la página, y algunas veces el nombre del editor HTML usado para crear la página.
- `description` da una explicación de la página o de su uso, y suele ser utilizado por los motores de búsqueda para hacer un resumen de la página.
- `copyright` es el aviso oficial de derechos de copia de la página.

## La Etiqueta <BODY>

`<BODY [bgcolor=valor] [onload=accion] [onunload=accion]> ... </BODY>`

- Determina la parte visible del documento.
  - `bgcolor` determina el color de fondo en tres códigos hexadecimales o alguno de los 16 colores predefinidos.
  - `onload` determina el script a ejecutar cuando la página ha sido cargada y antes de ser mostrada.
  - `onunload` lo mismo que en el caso anterior pero cuando la página se abandona para ir a otra.
- `<DIV>` y `<SPAN>` permite establecer divisiones dentro del cuerpo del documento para que luego puedan ser mostradas según se indique en las hojas de estilo.

## La Etiqueta <STYLE>

- Determina las propiedades que controlan la apariencia del documento: font, color, alineamiento y borders.
- Está soportado por la recomendación W3C sobre hojas de estilo, *Cascading Style Sheets*, CSS.

# La Etiqueta <SCRIPT>

```
<SCRIPT [type=MIMEtype] [language=lang] [src=URL]>
codigo </SCRIPT>
```

- Especifica el lenguaje utilizado en el documento, como por ejemplo JavaScript y Visual Basic Script.
- Contiene el código a ejecutar. Si se coloca en la cabecera, se interpretará antes que el documento sea cargado.
- Los browsers que no soporten scripts ignorarán la etiqueta pero mostrarán el código como texto normal. Para evitarlo se incluirá el código como si fuera un comentario.

  ```
  <SCRIPT> <!-- código --></SCRIPT>
  ```

# Caracteres Especiales

- Algunos caracteres han de ser 'escapados' o adoptar una forma de entidad SGML para poder ser representados por todos los navegadores (juego de caracteres ISO8859-1, Latin-1).
  - &quote; para ", & para &, > para $>$ y < para $<$.
  - á para á, Ñ para Ñ, α para $\alpha$.

# Alineamiento y Formateo de Texto

- Encabezamientos <H*n*>
- Párrafos <P>
- Rotura de línea <BR>
- Texto preformateado <PRE>
- Modificación de la apariencia <EM>, <STRONG> etc.

# Encabezamiento

`<Hn> encabezado de nivel n </Hn>`

- HTML permite definir 6 niveles de encabezados, siendo el 1 el más importante.
- Los browsers utilizan un tipo de letra mayor y más negrita para un nivel que para el siguiente.
- Un encabezado ocupa siempre una línea en exclusiva.
- Sirven para simular los títulos de capítulos, secciones, etc. pero no determinan que los párrafos que les siguen pertencezcan a capítulo o sección alguna.

# PÁRRAFOS

`<P> esto es un párrafo </P>`

- Un párrafo comienza en una linea nueva y se mantiene junto.
- Aunque la etiqueta de cierre es opcional y se puede omitir se recomienda su utilización siempre.
- Cada párrafo debe especificar su alineamiento con el atributo `align`, si no se les aplicará el por defecto (`align=right`).
- Se pueden introducir saltos de línea dentro de un párrafo con `<BR>`, etiqueta que no se cierra.

# Texto Preformateado

`<PRE> texto tal cual queremos que aparezca </PRE>`

- Permite describir la foma en que queremos que aparezca un texto, respetando el espaciado y los saltos de línea.
- El texto es presentado con una fuente de paso fijo.

## Modificación de la Apariencia I

- Se describe el uso de una palabra o palabras, en vez de los atributos físicos que van a tener.
- El estándar da unas recomendaciones sobre su presentación física, pero los browsers tienen cierta libertad para interpretarlas.
- Incluye la posibilidad de tratar elementos de frase y otros.

# Modificación de la Apariencia II

Enfatización:

- Énfasis básico `<EM> texto </EM>` asociado a cursiva.
- Énfasis fuerte `<STRONG> texto </STRONG>` asociado a negrita.
- El mismo efecto se puede conseguir con `<I>` y `<B>` pero perdemos el significado estructural de que queremos enfatizal algo, quedándonos exclusivamente en la forma física final.

## Modificación de la Apariencia III

- Referenciar un trabajo: `<CITE> referencia </CITE>`.
- Definiciones: `<DFN> término </DFN>`.
- Acrónimos: `<ACRONYM> letras </ACRONYM>`.
- Código de programas: `<CODE> código </CODE>`.
- Variables: `<VAR> variable </VAR>`.
- Ejemplo de salida de programas: `<SAMP> salida </SAMP>`.
- Ejemplo de entrada de programas: `<KBD> entrada </KBD>`.
- Dirección: `<ADDRESS> mi dirección </ADDRESS>`.
- Citas: `<QUOTE> párrafo de una cita </QUOTE>` y `<Q> cita corta </Q>`.
- Líneas: `<HR [align] [width=ancho] [size=pixels] [noshade]>`.

# Listas

- HTML proporciona un conjunto de etiquetas especiales para gestionar las listas, teniendo algún control sobre su apariencia.
- Las listas se pueden anidar.
- Listas Ordenadas: muestran un conjunto de items numerados para mostrar algún tipo de jerarquía o importancia.
- Listas Sin Orden: muestran un conjunto de items sin que el orden en que son mostrados implique grado de importancia o de jerarquía alguno.
- Glosarios: permiten definir una lista de definiciones.

# Listas Ordenadas

```
<OL>
<LI> item 1 </LI>
<LI> item 2 </LI>
...
<LI> item n </LI>
</OL>
```

- Se puede controlar el número inicial de la serie: `<OL start=numero>`.
- También se puede elegir el conjunto de caracteres que identifica cada iten en la lista: `<OL type=tipo>`, con `tipo` a elegir entre `1` para arábico, `a` o `A` para alfabético y `i` o `I` para romano.

# Listas Sin Orden

```
<UL>
<LI> item 1 </LI>
<LI> item 2 </LI>
...
<LI> item n </LI>
</UL>
```

- Se puede controlar el tipo de indicador de los items: `<UL bullet=tipo>` donde `tipo` es uno a elegir entre `circle`, `square` y `disc`.

# Glosarios

```
<DL>
<DT> término 1 </DT><DD> definición 1 </DD>
<DT> término 2 </DT><DD> definición 2 </DD>
...
<DT> término n </DT><DD> definición n </DD>
</DL>
```

- Los browsers se encargarán de mostrar el glosario de manera conveniente.
- Se pueden utilizar las etiquetas de formateo estándar para dar más énfasis a la presentación: `<TT>`, `<I>`, `<B>`, `<U>`, `<S>`, `<STRIKE>`, `<BIG>` y `<SMALL>`.

# Tablas I

```
<TABLE>
<CAPTION> Ejemplo de Tabla 3x2 </CAPTION>
<TR><TH>col1</TH><TH>col2</TH></TR>
<TR><TD>11</TD><TD>12</TD></TR>
<TR><TD>21</TD><TD>22</TD></TR>
<TR><TD>31</TD><TD>32</TD></TR>
</TABLE>
```

- `<TABLE>` engloba la tabla.
- `<CAPTION>` da título a la tabla.
- `<TR>` especifica cada fila.
- `<TH>` declara el encabezado de cada columna.
- `<TD>` delimita cada celda dentro de cada fila.

# Tablas II

- La etiqueta `<TABLE>` puede tener los siguientes atributos:
  - `align`: controla la colocación dentro de la página y accepta los valores `left`, `right` o `center`.
  - `width`: determina la anchura de la tabla en pixels o en un porcentaje de la anchura de la ventana del browser, `50%` por ejemplo.
  - `cols`: el número de columnas que va a tener la tabla.
  - `frame`: si se coloca un marco alrededor de la tabla.
  - `border`: determina la anchura en pixels del borde si lo hay.
  - `rules`: determina si se utilizan lineas para separar las celdas.
  - `cellspacing`: el espacio en pixels entre celdas.
  - `cellpadding`: el espacio en pixels entre el final de la celda y su contenido.

# Figuras I

`<IMG src=url [alt=descripcion]>`

- `src` determina el fichero (su URL) que contiene la imagen.
- `alt` es opcional pero se recomienda su uso, ya que en caso de que el browser no pueda mostrar la imagen, pondrá su descripción en su lugar.

# FIGURAS II

- `align` determina la posición de la figura con respecto a la línea de texto donde aparece. Esta opción está siendo abandonada a favor de utilizar CSS.
- `width` y `height` determinan en pixels la anchura y altura de la figura. El poner estas medidas hacen que la pagina sea cargada y mostrada con más rapidez.
- `border` fija en pixels el borde a dibujar alrededor de la figura. Es útil cuando la figura es parte de un hiperenlace.
- `hspace` y `vspace` determinan en pixels un espacio libre alrededor de la figura.

# URL

Un *Uniform Resource Locator*, URL, especifíca la localización exacta de un recurso en Internet. Es sólo una parte del *Uniform Resource Identifiers*, URI.

- URLs absolutos.
- URLs relativos.
- Fragmentos de URL.
- Esquemas URL.

# URLs Absolutos

`protocolo://servidor/path/fichero`

- Utiliza una dirección Internet completa para dar la localización de un recurso.
- También se puede utilizar la dirección IP para especificar el servidor.

# URLs Relativos

`pathRelativo/fichero`

- Se utilizan para acceder a ficheros que están en el mismo servidor.
- Se llama relativo porque su dirección se da en función de la URL del documento donde se utiliza.
- El browser quita todo lo que esté a la derecha de la última barra, /, de la URL del fichero actual y le añade el URL relativo.
- Los URLs relativos son imprescindibles si queremos que un conjunto de páginas Web sea portable.
- Se puede establecer el URL base en la cabecera con la etiqueta `BASE` siguiente: `<HEAD> <BASE href="http://www.nuevo.com/"> </HEAD>`.

# Fragmentos de URL

`protocolo://servidor/path/fichero#fragmento`

- Permite direccionar un fragmento de un fichero.
- El identificador del fragmento es el nombre o identificador del ancla referenciada.
- El URL puede ser absoluto o relativo.

# Esquemas de URL

- Un esquema es un protocolo de transferencia.
- Aunque existen muchos, en las páginas HTML se suelen utilizar sólo unos pocos: http, mailto, news, ftp, file, telnet, tv, callto, etc.
- Algunos esquemas tienen elementos opcionales que se añaden al URL.

# Hiperenlaces

```
<A name=nombre href=destino> origen </A>
```

- Los hiperenlaces son la esencia del Web.
- Permiten ir de un documento a otro siguiendo los caminos que han sido marcados por los creadores de los documentos Web.
- Tienen dos componentes: el ancla fuente y el ancla destino.
- Un ancla es un zona con nombre dentro de un documento HTML.

# Cascade Style Sheets

- HTML es un lenguaje de definición de estructura de documentos.
- Algunos elementos de HTML permiten alterar su apariencia mediante los atributos (por ejemplo, alineamiento, color, etc).
- Las hojas de estilo, CSS, permiten separan la definición de la estructura de la definición de la apariencia.
- Hay tres modalidades de hojas de estilo:
  - en linea,
  - embebidas y
  - externas.

# CSS En Línea

```
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 4.0//EN">
<HTML>
<HEAD><TITLE>CSS En Línea</TITLE></HEAD>
<BODY style="background: white">
<P style="font-size: 14pt"> parrafo a 14 puntos </P>
<P>párrafo normal</P>
</BODY>
</HTML>
```

- Se incluyen en las etiquetas que afectan.
- No alteran otras etiquetas en la página.
- Se utiliza para dar pequeños retoques de estilo.

# CSS Embebida

```
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 4.0//EN">
<HTML><HEAD><TITLE>CSS Embebida</TITLE></HEAD>
<STYLE>
BODY {background: #FFFFFF}
P {font-size: 14pt; text-ident: 0.5cm}
</STYLE>
<BODY>
...
</BODY></HTML>
```

- Se incluye la etiqueta `<STYLE>` entre `<HEAD>` y `<BODY>`.
- Para cada elemento HTML se especifican entre llaves pares atributo:valor separados con puntos y coma.
- Afectan a todos los elementos del documento.

# CSS Externa

```
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 4.0//EN">
<HTML>
<HEAD>
<TITLE>CSS Externa</TITLE>
<LINK rel=stylesheet href="estilo.css" type="text/css">
</HEAD>
<BODY>
...
</BODY>
</HTML>
```

- Todas las etiquetas afectadas serán interpretadas según el estilo definido en el fichero `.css` que estará compuesto de la misma manera que la etiqueta `<STYLE>`.

# Combinando CSSs

1. Se aplican las CSS externas globalmente.
2. Si se encuentra una CSS embebida en la misma página que tiene una CSS externa, el estilo embebido prevalece sobre el externo.
3. Cualquier estilo en linea va a sobrevivir a los estilos definidos de forma embebida y externa.

# Bibliografía


- Darnell Rick et al. *HTML4 Professional Reference Edition*. Sans Net. 1998.
- HTML Quick Reference
  http://www.cc.ukans.edu/~acs/docs/other/HTML_quick.shtml
- HyperText Markup Language
  http://www.w3c.org/pub/WWW/MarkUp
- A Beginner's Guide to HTML
  http://www.ncsa.uiuc.edu/General/Internet/WWW/HTMLPrimer.html
- Special Characters
  http://www.utoronto.ca/webdocs/HTMLdocs/NewHTML/entities.html
- ISO8859-1 (Latin1) Table
  http://www.uni-passau.de/~ramsch/iso8859-1.html