

XHTML y CSS 2



Área de Ingeniería Telemática

Contenidos

- XHTML 1.0
 - Modo de renderizado
 - Box Model
 - Tipos de elementos
 - Posicionamiento de elementos
- CSS 2

XHTML 1.0

```
<html>
<head>
  <title>Titulo del documento</title>
</head>
<body>
  <p>Este es un HTML 4.01 correcto<br>
  pero un XHTML 1.0 incorrecto
</body>
</html>
```

```
<html>
<head>
  <title>Titulo del documento</title>
</head>
<body>
  <p>Este es un HTML 4.01 correcto<br />
  y un XHTML 1.0 correcto</p>
</body>
</html>
```

- Diferencias principales con HTML 4.01
 - Elementos XHTML apropiadamente anidados
 - Documentos bien escritos
 - Nombres y atributos de etiquetas en minúsculas
 - Todos los elementos cerrados, incluidos elementos vacíos
 - Los elementos se identifican sólo por 'id' únicos
 - Únicos que pueden usar 'name' elementos de formulario
 - Todos los atributos tienen valor
 - Preferiblemente script y style externos → evitar CDATA y errores con `< &]]>`

XHTML 1.0

- XHTML 1.0 tiene los mismos elementos y atributos que HTML 4.01
- Si documento XHTML 1.0 puro (sin otro lenguaje de marcado) no hay diferencia entre este y uno HTML 4.01 siguiendo rigidez XML
- Se debería de enviar como application/xhtml+xml ó application/xml, pero se puede enviar como text/html
 - IE no lo entiende, resto si → IE necesita text/html
 - Soluciones:
 - W3C: aprovechar bug de IE:
<http://www.w3.org/MarkUp/2004/xhtml1-faq>
 - Gurus: programación en servidor: HTTP_ACCEPT, header
 - Web: enviar todo como text/html

XHTML y CSS

- CSS 1 principalmente para dar formatos de fuentes, tamaños, colores, ...
- CSS 2 para posicionamiento de elementos (layout) + lo anterior
 - Mayor separación entre contenido y forma
 - Facilidad de rediseño (<http://www.csszengarden.com/>)
 - Inicialmente mayor complejidad al desarrollar => ya no se hace con tablas sino con div (capas)
 - Aumenta la usabilidad y accesibilidad
 - Combinación de plataforma/navegador modifica apariencia:
 - No soportan todos el mismo nivel
 - Hay cosas que las representan mal (especialmente IE)

Contenidos

- XHTML 1.0
 - Modo de renderizado
 - Box Model
 - Tipos de elementos
 - Posicionamiento de elementos
- CSS 2

Modo de renderizado

- Navegadores tiene 2 formas de comportarse:
 - Quirks Mode: como navegadores antiguos (<v4)
 - Strict Mode: intentando seguir estándar (modernos)

<http://www.quirksmode.org/css/quirksmode.html>
- Estándar XHTML recomienda poner prolog xml antes que DOCTYPE → IE 6 salta a Quirks → soluciones:
 - No poner prolog → Todos los navegadores modernos en Strict
 - Poner prolog → IE 6 no renderiza como un navegador moderno

Contenidos

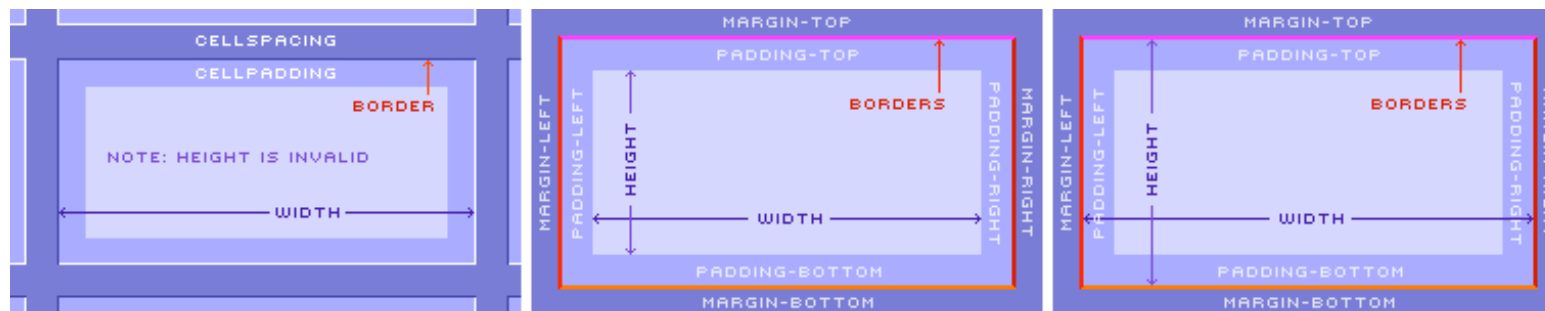
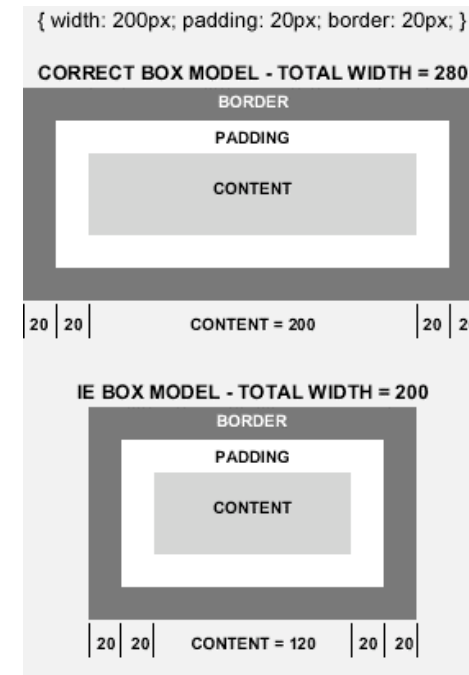
- XHTML 1.0
 - Modo de renderizado
 - Box Model
 - Tipos de elementos
 - Posicionamiento de elementos
- CSS 2

Box Model

- 2 Box Model:
 - Estándar (W3C): lo aplican navegadores en Strict Mode => navegadores modernos (IE 6 sin prolog)
 - Tradicional: dentro del ancho del elemento incluye el padding y el border => navegadores viejos (IE 6 con prolog)

<http://css.maxdesign.com.au/listamatic/about-boxmodel.htm>

<http://css-discuss.incutio.com/?page=BoxModelHack>



Contenidos

- XHTML 1.0
 - Modo de renderizado
 - Box Model
 - Tipos de elementos
 - Posicionamiento de elementos
- CSS 2

Tipos de elementos

- **Block:** están separados de los elementos adyacentes: p, div, ...
- **Inline:** en vez de estar separados de los elementos adyacentes aparecen a continuación de ellos: a, strong, span, cite, ...
- **List-item:** parecidos a los elementos de bloque, pero a la izquierda tienen marcas: li, ul, ol

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
"http://www.w3.org/TR/html4/strict.dtd">
<html>
<head>
<title>Tipos de elementos</title>
</head>
<body>
<p>Esto aparece separado de lo siguiente.</p>
<p>Elemento tipo block, pero <strong>esto es un
elemento tipo inline que aparece</strong> seguido.</p>
<ul>
<li>Lista</li>
</ul>
<p>Texto normal</p>
</body>
</html>
```

Esto aparece separado de lo siguiente.

Elemento tipo block, pero **esto es un elemento tipo inline que aparece seguido.**

- Lista

Texto normal

Contenidos

- XHTML 1.0
 - Modo de renderizado
 - Box Model
 - Tipos de elementos
 - Posicionamiento de elementos
- CSS 2

Posicionamiento de elementos

- CSS2 posiciona los elementos de 4 maneras diferentes
- **Estatico**: modo por defecto, mostrar los elementos en el orden natural de lectura.

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed diam nonummy nibh euismod tincidunt ut Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed diam

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed diam nonummy nibh euismod tincidunt ut Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed diam

- **Absoluto**: establecer posición exacta (top, left, right, bottom) de un elemento respecto a su padre.

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed diam nonummy nibh euismod tincidunt ut Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed diam nonummy nibh euismod tincidunt ut Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed diam nonummy nibh euismod tincidunt ut

- **Fixed**: subtipo del absoluto. El scroll afecta a un elemento posicionado absolutamente, con fixed no. Fixed tiene posición absoluta no sobre la ventana. Se puede usar para simular frames.

Posicionamiento de elementos

- **Float:** se pone el elemento donde estaría posicionado estáticamente y se saca del flujo normal a la derecha o izquierda. El siguiente elemento flota alrededor suyo.

>Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed diam nonummy nibh euismod tincidunt ut Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed diam nonummy nibh euismod tincidunt ut Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed diam nonummy nibh euismod

- **Relativo:** Coge el elemento donde estaría posicionado estáticamente y lo desplaza Zpx hacia abajo y Ypx hacia la derecha. El resto de elementos considera que sigue estando en su posición normal.

>Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed diam nonummy nibh euismod tincidunt ut Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed diam nonummy nibh euismod tincidunt ut Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed diam nonummy nibh euismod

Posicionamiento de elementos

- **Problema:**
 - Para hacer layout's interesantes hay que mezclar diferentes posicionamientos
 - Al mezclar posicionamientos resultado muy enrevesado
- En **teoría** se puede poner cualquier posicionamiento a cualquier tipo de elemento
- En **práctica** el posicionamiento general (layout) se hace prácticamente solo con `div`:
 - Consecuencia de bugs, malas implementaciones, ... de los navegadores

Contenidos

- XHTML 1.0
 - Modo de renderizado
 - Box Model
 - Tipos de elementos
 - Posicionamiento de elementos
- CSS 2

CSS 2

- Añade todos los estilos necesarios para posicionamiento y maquetación de layouts
- Prácticamente soportado **completamente** por todos los navegadores modernos => los viejos pasan de estos estilos
- Problema: la implementación de todos los estilos en todos los navegadores no es coherente => bugs de renderizado para ciertos navegadores en ciertas circunstancias
- Por eso se considera más difícil maquetar con CSS, que con tablas
- Navegador más "estándar": Firefox
- Herramienta útil de desarrollo: Web Developer Toolbar

CSS 2

- position: static, relative, absolute, fixed
- float: none, left, right, both
- display: none, block, inline, inline-table, list-item, ...
- ...