

# CSS: hojas de estilo en cascada

Luis Fernando Llana Díaz

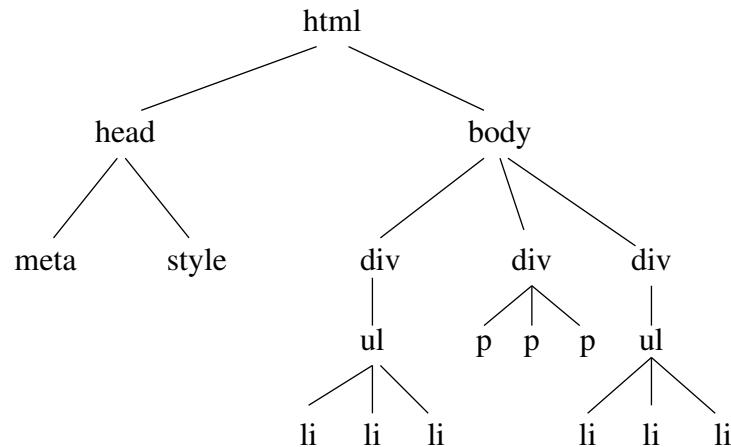
Departamento de Sistemas Informáticos y Computación  
Universidad Complutense de Madrid

7 de mayo de 2007

## Visualización de una página I

- 1 Analizar el HTML y crear un árbol que representa el documento (document tree).

## Visualización de una página II



## Visualización de una página III

- 2 Identificar el *medio* en el que va a representar la página: braille, embossed, handheld, print, projection, screen, speech, tty, tv.
- 3 Descargar las hojas de estilo asociadas al documento y al medio.
- 4 Anotar en cada nodo del árbol con cada propiedad disponible según las hojas de estilo, si el medio es *screen* se aplica el *visual formatting model*.
- 5 Generar una estructura de formato, es parecido al árbol anterior pero puede aparecer y desaparecer información:
  - Si un elemento tiene el valor `none` en la propiedad `display`, no aparece.
  - En las listas aparece puede aparecer una *imagen bola*.

## Visualización de una página IV

No se modifica el *document tree*.

- 6 Transferir la estructura de formato al medio concreto (impimir los resultados, mostrarlos en la ventana de un navegador gráfico, pronunciar el texto, etc.).

## CSS: Cascade Style Sheets

HTML estructura lógica del documento.

CSS presentación (estilo) del documento.

# Principios del diseño CSS I

- 1 Compatibilidad, hacia adelante y hacia atrás.
- 2 Complemento a los documentos estructurados. Documentos uniformes. Es fácil cambiar la apariencia de un documento sin alterar el HTML.
- 3 Independencia del fabricante, plataforma y dispositivo.
- 4 Mantenibilidad. Si el jefe dice que los comentarios deben ir en rojo, basta cambiar una línea
- 5 Mejora del rendimiento de la red.
- 6 Flexibilidad. Se pueden especificar información de estilo en el navegador (user agent), del usuario, las que aparecen el documento.
- 7 Riqueza. La calidad depende de la creatividad del autor.

# Principios del diseño CSS II

- 8 Accesibilidad.
  - Se puede controlar la apariencia del texto, no son necesarios los “botones gráficos”.
  - Hace innecesario el uso de trucos para la maquetación: marcos, imágenes invisibles, o tablas.
  - El modificador **!important** permite al usuario definir sus necesidades.
  - El valor **inherit** mejora la generalidad de la cascada y permite una mejor y más consistente ajuste.
  - Se soporta medios como braille, embossed y tty.

```

body{
  margin:0;
  padding:0;
  font-family:sans-serif;
}
body, div#derecha, div#izquierda,h1,h2,h3,h4,h5,h6 {
  background-color:#99FF33
}
div.lateral {
  padding-top:2em;
}
div.lateral p {
  margin-left:1em;
  text-indent:0
}
div.lateral a, div.lateral p, div.lateral div.iconos{
  font-size:xx-small;
}
div.lateral ul li a {
  margin-top:0.5em;
  padding:0.5em;
  display:block;
  background-color:yellow;
  border: 2pt outset #FFCC00;
}
div.lateral ul li a:hover{
  background-color:red;
  border: 2pt outset #CC0000;
}

```

1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29

```

<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-15">
<link rel="stylesheet" type="text/css" href="estilo2.css" media="screen">
<title>Prueba 3</title>
</head>

```

1  
2  
3  
4  
5

```

<style type="text/css" media="screen">
/* cosas que entiende IE6 */
@import url("estilo.css");

/* cosas que entienden los navegadores buenos, pero no IE6,
   oculta el mensaje sin CSS */
@import url("completo.css") screen;
</style>

```

1  
2  
3  
4  
5  
6  
7  
8

```

<p style="text-align:center">Texto centrado</p>

```

1

all

braille Dispositivos braille táctiles.

embossed Para impresoras braille.

handheld Pantalla pequeña, poco ancho de banda.

print Para ser impreso, con páginas.

projection

screen Pantallas de ordenador a color.

speech

tty Terminales con tipo de letra de ancho fijo y capacidades limitadas.

tv Televisión, baja resolución con color.

	<i>continuous/paged</i>	<i>visual/audio/speech/tactile</i>	<i>grid/bitmap</i>	<i>interactive/static</i>
<i>braille</i>	continuous	tactile	grid	both
<i>embossed</i>	paged	tactile	grid	static
<i>handheld</i>	both	visual, audio, speech	both	both
<i>print</i>	paged	visual	bitmap	static
<i>projection</i>	paged	visual	bitmap	interactive
<i>screen</i>	continuous	visual, audio	bitmap	both
<i>speech</i>	continuous	speech	N/A	both
<i>tty</i>	continuous	visual	grid	both
<i>tv</i>	both	visual, audio	bitmap	both

# La cascada I

Los agentes de usuario (navegadores) deben asignar las propiedades de la siguiente forma:

- 1 Si la *cascada* calcula un valor le asigna ese.
- 2 Si la propiedad es *inherit*, le asigna el valor del nodo padre.
- 3 En caso contrario le asigna el valor *inicial*.

# Preferencia en la cascada

Las hojas de estilo pueden tener 3 orígenes: autor, usuario, agente de usuario.

Las reglas tiene un peso, si dos reglas se contradicen la de mayor peso tiene preferencia

- Las reglas del autor tienen más peso que las del usuario
- Las del usuario más que la del agente de usuario.
- La preferencia se puede invertir con la directiva *!important*.

## El orden de la cascada

- 1 Encontrar todas la reglas que se pueden aplicar a un determinado elemento.
- 2 Ordenar por orden de importancia (creciente):
  - 1 reglas del agente de usuario
  - 2 reglas normales del usuario
  - 3 reglas normales del autor
  - 4 reglas importantes del autor
  - 5 reglas importantes del usuario
- 3 Ordenar por *concreción de la regla*. Las más específicas tiene mayor peso.
- 4 Si tras todo hay dos con el mismo peso, la que aparece más tarde (en el fichero) tiene prioridad.

## Selectores CSS I

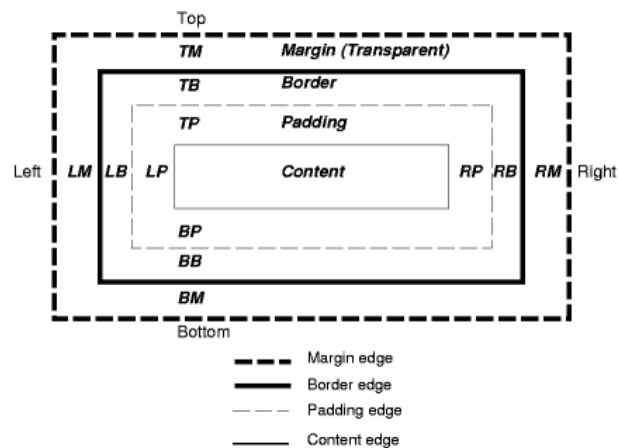
- 1 \* Matches any element. [Universal selector](#)
- 2 E Matches any E element (i.e., an element of type E). [Type selectors](#)
- 3 E F Matches any F element that is a descendant of an E element. [Descendant selectors](#)
- 4 E > F Matches any F element that is a child of an element E. [Child selectors](#) . ¡¡¡¡!E6 no lo entiende!!!! se tiene que simular con JavaScript
- 5 E#myid Matches any E element with ID equal to "myid". [ID selectors](#)



## Selectores CSS II

- 6 **E:first-child** Matches element E when E is the first child of its parent. **!!!!IE6 no lo entiende!!!**
- 7 **E:link**, **E:visited** Matches element E if E is the source anchor of a hyperlink of which the target is not yet visited (:link) or already visited (:visited). **The link pseudo-classes**
- 8 **E:active**, **E:hover**, **E:focus** Matches E during certain user actions. **The dynamic pseudo-classes** **!!!!en IE6 sólo funciona con etiquetas <a>...</a>!!!! se tiene que simular con JavaScript**

## Modelo de caja I



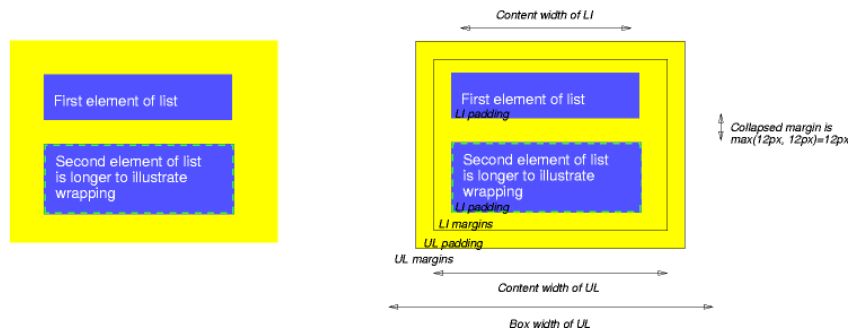
## Modelo de caja II

```
ul {  
  background: yellow;  
  margin: 12px 12px 12px 12px;  
  padding: 3px 3px 3px 3px;  
}  
li {  
  color: white;  
  background: blue;  
  margin: 12px 12px 12px 12px;  
  padding: 12px 0px 12px 12px;  
  list-style: none  
}  
li.withborder {  
  border-style: dashed;  
  border-width: medium;  
  border-color: lime;  
}
```

1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17

## Modelo de caja III

<http://antares.sip.ucm.es/~luis/accesibilidadWEB05-06/ejemplos/css/ejemploLI.html>



## Modelo de caja IV

**padding** del color del fondo de la caja, no colapsa

**padding**, **padding-top**, **padding-right**,  
**padding-bottom**, **padding-left**

**margin** transparente (de color de la caja contenedora),  
colapsa.

**margin**, **margin-top**, **margin-right**,  
**margin-bottom**, **margin-left**

## Modelo de caja V

Tamaños:

**longitudes** la unidad es obligatoria (salvo si es 0):

- absolutas (a evitar), puntos **pt**, milímetros **mm**, centímetros **cm**.
- relativas a la fuente actual **em**, a la resolución del dispositivo **px**.

**porcentaje** %, relativo a la caja contenedora.

# Modelo de caja VI

Global: **border**.

Anchura del borde: **thin**, **medium**, **thick**, o longitud explícita.

**border-top-width**, **border-right-width**,

**border-bottom-width**, **border-left-width**, y **border-width**.

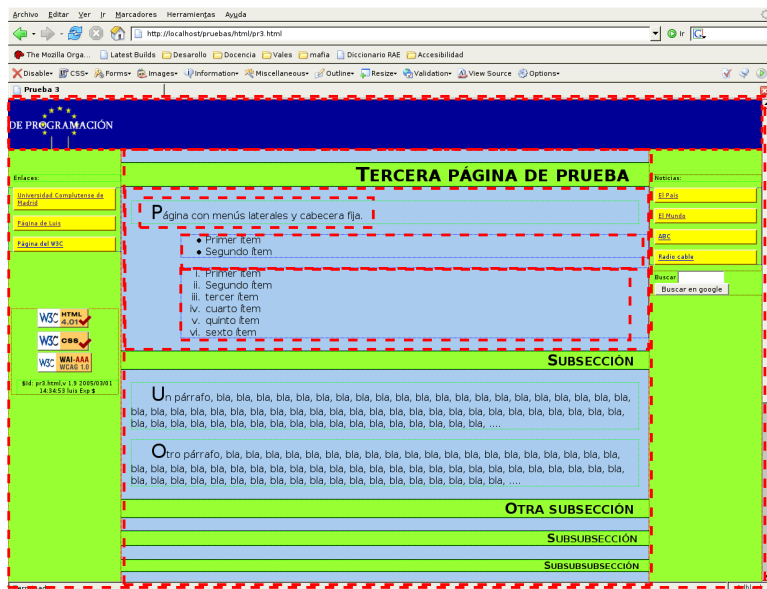
Estilo del borde: **none**, **hidden**, **dotted**, **dashed**, **solid**, **double**, **groove**, **ridge**, **inset**, **outset**.

**border-top-style**, **border-right-style**,

**border-bottom-style**, **border-left-style**, y **border-style**.

Color del borde: **border-top-color**, **border-right-color**,

**border-bottom-color**, **border-left-color**, y **border-color**.



`display` valores `block`, `inline`, `none`, ....

`position` Propiedades relacionadas `top`, `right`, `bottom`, and `left`:

`static` posicionado *normal*, no tiene efecto.

`relative` Se posiciona de forma relativa con la posición normal, reserva el hueco con respecto al posicionamiento normal

`absolute` Se posiciona de forma absoluta, con respecto al antecesor que no sea `static`. No reserva hueco con respecto al posicionamiento normal.

`fixed` Igual que `absolute`, pero no se mueve con el *scrolling*. **AVISO** NO lo entiende IE6.

```
div#superior {
  position: absolute;
  left: 0; top: 0;
  height: 17%; width: 100%;
}
div#izquierda {
  position: absolute;
  left: 0; top: 17%;
  height: 83%; width: 15%;
  margin: 0;
}
div#centro {
  position: absolute;
  top: 17%; left: 15%;
  width: 70%;
}
div#derecha {
  position: absolute;
  left: 85%; top: 17%;
  width: 15%; height: 83%;
}
```

1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21

# Anchuras y alturas

- `width`
- `height`

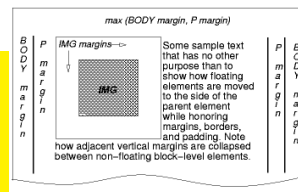
Se aplican a cajas de bloque.

Valores: `<length>` | `<percentage>` | `auto` | `inherit`

Se aplican al contenido, al tamaño total hay que añadir el *padding* y el *margin*.

## Posicionamiento flotante: Float

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0//EN">
<HTML>
  <HEAD>
    <TITLE>Float example</TITLE>
    <STYLE type="text/css">
      IMG { float: left }
      BODY, P, IMG { margin: 2em }
    </STYLE>
  </HEAD>
  <BODY>
    <P><IMG src=img.png alt="This image will illustrate floats">
      Some sample text that has no other...
    </BODY>
  </HTML>
```



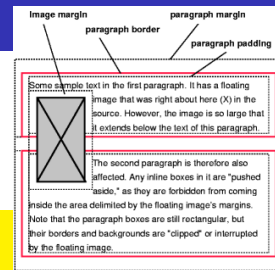
8  
9  
10  
11  
12  
13  
14

# Posicionamiento flotante: Float

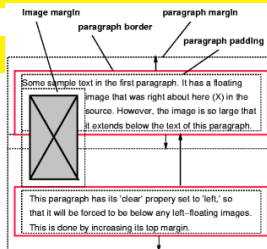
Valores: `code`, `auxleft`, `right`, `none`

```
<BODY>
<P>Some sample text
<IMG src=img.png">
that has no other...
</BODY>
```

```
p { clear: left }
```



1  
2  
3  
4  
5



1

## Efectos visuales

`overflow` : `visible`, `hidden`, `scroll`, `auto`

`clip` : recorte, se puede especificar una forma.

`visibility` `visible`, `hidden`

`z-index` : indica la capa en la que se coloca, se puede poner encima (número mayor), o debajo (número más pequeño).

# Fuentes

**font-family** Debe incluir una de las siguientes: **serif**,  
**sans-serif**, **cursive**, **fantasy**, **monospace**.

```
body { font-family: "New Century Schoolbook", serif } 1
```

**font-style** normal, italic, oblique

**font-variant** normal, small-caps

**font-weight** normal, bold, bolder, lighter

**font-size** Porcentaje (con respecto al padre), en puntos y  
**xx-small**, **x-small**, **small**, **medium**, **large**,  
**x-large**, **xx-large**,

# Colores

```
em { color: #f00 } /* #rgb */ 1
em { color: #ff0000 } /* #rrggbb */ 2
em { color: rgb(255,0,0) } 3
em { color: rgb(100%, 0%, 0%) } 4
```

maroon #800000	red #ff0000	orange #ffa500	yellow #ffff00	olive #808000
purple #800080	fuchsia #ff00ff	white #ffffff	lime #00ff00	green #008000
navy #000080	blue #0000ff	aqua #00ffff	teal #008080	
black #000000	silver #c0c0c0	gray #808080		



# Listas

`list-style-type` valores `disc`, `circle`, `square`, `decimal`,  
`decimal-leading-zero`, `lower-roman`,  
`upper-roman`, `lower-latin`, `upper-latin`.

`list-style-image`

```
ul { list-style-image: url("http://png.com/ellipse.png") }
```

1

`list-style-position` valores `inside`, `outside`.

- first list item  
comes first

- second list item  
comes second

---

- first list  
item comes first

- second list  
item comes second