

Cascading Style Sheet (CSS)

Mario Bravetti

Introduzione

- ◆ Linguaggio HTML ideato con scopo di descrivere il **contenuto** di un documento e la sua **struttura**, ma **non relative caratteristiche di visualizzazione**.
- ◆ In prototipo WWW di Tim Berners-Lee (1989):
 - linguaggio di stile che consentiva **ai lettori** di **definire personalmente come presentare** i documenti HTML,
 - **prime versioni dei browser Web** (Mosaic) permettevano agli utenti di specificare **queste caratteristiche**.
 - ma prototipo di Mosaic: **pochissime opzioni** per l'utente (dimensione e nome del font da usare per i testi)

Introduzione

- ◆ Netscape 1.0 introdusse **alcuni tag** (font, center) che permettevano all'autore di specificare alcune **caratteristiche di presentazione**.
- ◆ **Successo** di HTML e del WWW introdusse, nel mondo degli **autori di pagine Web** esperti di marketing, grafici e tipografi:
 - fu **fondamentale** definire **aspetto finale delle pagine** inserendo le **informazioni di presentazione all'interno del documento** sotto forma di tag e attributi HTML.

Introduzione

- ◆ Per questo, con HTML 3.2 venne introdotta:
 - moltitudine di **nuovi tag e attributi** dedicati alla specifica di **caratteristiche tipografiche e di formattazione presentazionale**.
- ◆ Questa purtroppo **non fu una scelta felice** poiché visualizzazione di un documento HTML infatti dovrebbe essere
 - **indipendente dal dispositivo utilizzato** e non vincolata a specifiche caratteristiche imposte dall'autore.

Introduzione

- ◆ Inoltre inserendo informazioni di presentazione all'interno del documento insieme ai contenuti:
 - è necessario ripeterle su ogni pagina e non è quindi possibile riutilizzarle
 - qualsiasi successiva modifica alle caratteristiche di presentazione si traduce in un intervento massiccio dell'autore su tutte le pagine interessate
 - i marcatori di presentazione disponibili in HTML presentano numerose limitazioni
 - non consentono di controllare con precisione layout, margini, font, colori, spaziatura e posizionamento elementi.

Contenuti e presentazione

- ◆ Combinazione di fogli di stile CSS e linguaggio HTML 4 definisce un nuovo approccio per la realizzazione dei documenti che prevede la separazione dei contenuti dalla presentazione:
 - i tag disponibili nel linguaggio sono solo quelli che definiscono i contenuti e la loro strutturazione
 - le caratteristiche di presentazione dei contenuti sono invece affidate ad un foglio di stile
 - definisce i font, dei colori, degli stili, del posizionamento degli elementi e di tutto ciò che determina l'aspetto grafico della pagina;
 - è legato allo specifico dispositivo usato per visualizzare il contenuto.

Vantaggi e svantaggi

- ◆ Vantaggi (separazione contenuti/presentazione):
 - **il riutilizzo** di un medesimo schema di presentazione in documenti (e siti) diversi
 - una **maggiore facilità di manutenzione**, in quanto consente di apportare le modifiche a una sola entità (il foglio di stile) per variazioni a tutti i documenti
- ◆ Svantaggi:
 - **Livello ulteriore di complessità** nella progettazione:
 - notevole quantità di lavoro talvolta necessaria per ottenere **risultati prevedibili ed uniformi su tutti i browser.**

Cascading Style Sheets

- ◆ 1996: introdotto il **primo standard per i CSS**:
 - utilizzo di **fogli di stile multipli che agivano uno dopo l'altro in cascata** tra loro stabilendo così le caratteristiche tipografiche e di layout di un documento.
- ◆ CSS permettono quindi di avere un controllo dell'aspetto presentazionale di un documento HTML **sia da parte dell'autore che da parte del lettore.**
- ◆ Altra caratteristica vincente è quella di essere:
 - **indipendenti da specifico set di elementi ed attributi HTML**, così da rendere possibile anche il supporto di **nuove versioni di HTML** e altri linguaggi, es. **XML.**

Standard CSS

- ◆ CSS 2 (1998): evoluzione di CSS 1 (1996). che introduce il supporto per **media multipli** e un **linguaggio di layout complesso** (posizionamenti).
- ◆ CSS 2.1 (1998) è un'ulteriore evoluzione che **corregge errori** e introduce/rimuove **funzionalità già/non supportate dai browser**.
- ◆ **Supporto dei vari browser** è complesso e difficile:
 - ognuno infatti **implementa aspetti diversi ed incompatibili delle varie specifiche di CSS**, senza supportare **integralmente** nessuna di esse.
 - risulta **difficile scrivere codice "cross-browser"**

CSS 2

- ◆ Deriva da CSS 1: tutti i fogli di stile realizzati con CSS 1 sono fogli di stile CSS 2 validi.
- ◆ Supporta fogli di stile specifici per il mezzo
 - autori possono definire la presentazione dei documenti per browser visuali, periferiche acustiche, stampanti, periferiche braille, palmari, ecc.
- ◆ Inoltre supporta:
 - posizionamento dei contenuti, font scaricabili, impaginazione di tabelle, caratteristiche per l'internazionalizzazione, numerazione automatica, proprietà relative all'interfaccia utente, ...

CSS 3

- ◆ La specifica dello standard per la prima volta è divisa in **moduli in diverso stato di avanzamento**
 - oltre 50 moduli che estendono **diverse parti** di CSS2 preservando la retrocompatibilità
- ◆ Attualmente **solo 4** sono state pubblicati:
 - “Media Queries”, “Namespaces”, “**Selectors Level 3**” e “Color”
- ◆ Altre sono in stato di **candidate recommendations**
 - “**Backgrounds and Borders**” e “Multi-column Layout”
 - quindi raccomandato il supporto senza **vendor-prefix**
 - es. **border-radius** era **-moz-border-radius** in firefox

Modello di elaborazione

- ◆ Il browser opera come segue:
 - analizza il documento HTML sorgente e crea una **struttura ad albero** (gerarchia di contenimento degli elementi) chiamata **document model**
 - identifica il **tipo di media di destinazione**
 - reperisce tutti i **fogli di stile** associati al documento **specificati per il tipo di media di destinazione**.
 - per ciascun elemento dell'albero assegna un **unico valore** a ciascuna **proprietà relativa allo stile**.
 - per determinare questo valore il browser applica alle **regole CSS** meccanismi di **specificità, ereditarietà e organizzazione a cascata** (vedremo in seguito).

Modello di elaborazione

- a partire dall'albero del documento genera quindi una struttura di formattazione presentazionale (con struttura ad albero normalmente simile)
 - eventuali differenze possono essere dovute alla presenza di pseudo-elementi e contenuti generati (vedremo in seguito).
- trasferisce la struttura di formattazione generata al mezzo di destinazione
 - per stampare i risultati, mostrarli sullo schermo, riprodurli come discorso in un apparecchio di sintesi vocale, ecc...