

CSS2: lo stile del Web si evolve

Migliorare l'accessibilità delle pagine del proprio sito, separandone contenuto e presentazione? Si può! Basta rispettare le nuove specifiche relative ai nuovi fogli stile

In origine il Web era senza stile. Era un luogo di scambio di idee scientifiche e di incontro tra ricercatori e studiosi appartenenti a diverse università e comunità scientifiche. In accordo con tali scopi e con il tipo di documenti scambiati, l'HTML, fino alla versione 2 datata 1995, si limitava a fornire gli strumenti per evidenziare la funzione logica dei contenuti: titoli, paragrafi, di testo, elenchi, enfasi (elemento **EM**), forte enfasi (**STRONG**), riferimenti bibliografici (**CITE**), citazioni (**BLOCKQUOTE**), indirizzi (**ADDRESS**), codice (**CODE**), testo immesso dall'utente (**KBD**), variabili (**VAR**), e poco altro. Fondamentalmente, il dissidio tra contenuti e presentazione dei contenuti non esisteva. Non c'era ancora l'esigenza di dare stile e personalità grafica ai documenti: bastava che fossero logicamente ben strutturati. Poi qualcosa cambiò. Il Web si allargò e raggiunse il grande pubblico. A quel punto, chiunque aveva una connessione a Internet e programmi per navigare e scrivere codice HTML poteva pubblicare un proprio sito. E fu proprio allora che cominciò il dissidio fra il contenuto e la sua rappresentazione grafica o, per usare le parole di Nietzsche «lo strano contrasto di un'interiorità a cui non corrisponde nessuna esteriorità».

NUOVE ESIGENZE STILISTICHE

Gli autori volevano controllare l'aspetto delle loro pagine, volevano impostare lo sfondo, il colore del testo, e volevano posizionare i blocchi sulla pagina in modo preciso ed accattivante. Il linguaggio HTML, pensato per dare forma logica ai contenuti ma non per renderli esteticamente gradevoli, non era adatto allo scopo. A quel punto le società produttrici di browser, cogliendo le esigenze del pubblico, cominciarono ad implementare elementi e attributi proprietari, che garantivano funzioni di stile, ma allo stesso tempo snaturavano lo scopo originario del linguaggio di marcatura. Si arrivò così nel 1997 alla versione **3.2** di **HTML**, che recepiva le nuove esigenze stilistiche, introducendo elementi come **FONT** e attributi come **bgcolor**, nel tentativo di fornire agli autori uno standard condiviso e di fermare la cosiddetta «guerra dei browser», combattuta a colpi di funzioni proprietarie da *Microsoft* e *Netscape*. In quello stesso periodo, poiché gli autori non riuscivano ancora ad avere un controllo preciso sul posizionamento dei blocchi nella pagina, si sviluppò l'abitudine di usare le tabelle (l'elemento **TABLE**) per creare griglie d'impaginazione. Fu una sorta di rivoluzione: in breve il Web si riempì di siti dalla complicata struttura grafica, che non avevano più nulla della spartana semplicità delle origini. Ma i più attenti colsero gli svantaggi evidenti che tale pratica comportava: il peso eccessivo delle pagine, la mancanza

di logica e di semantica nel codice, il caotico groviglio di contenuti e stili grafici rappresentavano, e rappresentano, una barriera all'accesso per alcune tipologie di utenti, come i non vedenti, che possono sapere di un sito Web solo ciò che il loro *screen-reader* è in grado di leggergli.

LO STANDARD CSS1

Già da tempo si avvertiva l'esigenza di un linguaggio di stile che permettesse di gestire separatamente la presentazione dei documenti Web, sottraendo tale funzione all'HTML e restituendogli il compito esclusivo di strutturare i contenuti. Della redazione di un simile linguaggio si occupò il **W3C** (*World Wide Web Consortium*, www.w3.org) che, fondato nel 1994, aveva cominciato ben presto la sua opera di standardizzazione di linguaggi per il Web. Nel 1996 videro così la luce le specifiche **CSS1** (*Cascading Style Sheets Level 1*, "Fogli di

SUL
CD

Disponibile la traduzione in italiano dello standard CSS del W3C, in formato pdf ed in formato html (più aggiornata)



Figura 1
L'originale specifica per un linguaggio a fogli di stile proposta di **Håkon Wium Lie**: era il **10 ottobre 1994**



stile a cascata di livello 1"), ad opera di Bert Bos, www.w3.org/People/Bos/, e Håkon Wium Lie, <http://people.opera.com/howcome/> (Figura 1). Tuttavia il supporto iniziale dei browser ai CSS era estremamente limitato e del pari limitato fu il loro successo. Ma le cose stavano rapidamente cambiando: Microsoft e Netscape erano entrati a far parte del W3C e i loro browser avevano cominciato a supportare i fogli di stile; nuovi browser come Opera erano apparsi sulla scena del Web. Vi erano tutti i presupposti per continuare sulla strada intrapresa. Nel maggio 1998 furono quindi pubblicate le specifiche CSS2. Lo "strano contrasto" poteva considerarsi risolto (Figura 2).

L'EVOLUZIONE DEI CSS1

I CSS2, disponibili anche in traduzione italiana all'indirizzo www.diodati.org/w3c/css2/cover.html, ampliano notevolmente le caratteristiche e le funzionalità dei CSS1. Vengono definiti i concetti di conformità dei programmi utente (tra cui i browser) alle specifiche, nonché le regole a cui tali programmi devono obbedire nell'applicare proprietà di stile a una data pagina Web. Vengono definiti nuovi tipi di media (come i *media acustici*) e vengono loro attribuite specifiche proprietà. Viene definito il concetto di "modello di formattazione visuale", con numerose proprietà per gestire il posizionamento dei contenuti sulla pagina, le loro dimensioni e l'aspetto grafico. Viene definito il concetto di "modello per riquadrati" (box model), che specifica le proprietà di stile che possono essere attribuite a ciascun blocco sulla pagina: bordi, margini, ecc (Figura 3). Si ampliano le proprietà per la formattazio-

ne del testo (colore, allineamento, tipo di carattere, spaziatura tra lettere e parole, gestione degli spazi bianchi ecc.); si definiscono stili per le tabelle, con la speranza che possano tornare a svolgere la funzione loro destinata dal linguaggio HTML, ossia la strutturazione di dati tabulari. Si definiscono inoltre nuovi tipi di selettori e si rivedono alcuni concetti di base, quali le *pseudo-classi* e gli *pseudo-elementi*, nonché la possibilità di generare contenuti tramite apposite proprietà. Ad uno sguardo d'insieme i CSS2 appaiono un lavoro poderoso e profondamente innovativo. Risaltano in particolare le caratteristiche pensate per potenziare l'accessibilità dei documenti Web. Separando il contenuto dalla presentazione, ossia gli stili dalla marcatura, gli autori possono semplificare e snellire il codice HTML dei loro documenti, rendendoli più accessibili.

LA RIDUZIONE DEGLI ABUSI NEI CSS2

I CSS2 consentono finalmente un controllo preciso sulla spaziatura, l'allineamento ed il posizionamento degli elementi. Gli autori possono quindi evitare il cosiddetto "abuso di tag", ossia l'uso improprio di elementi

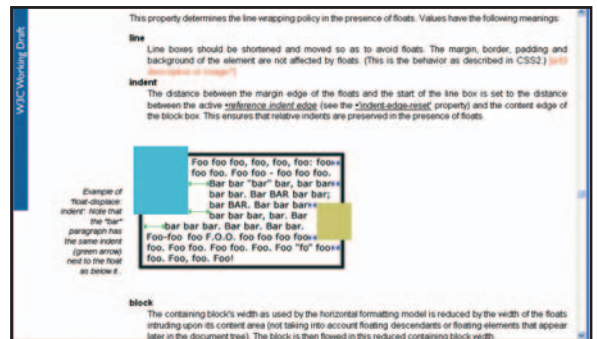


Figura 3 Estratto del modulo CSS3 dedicato al modello per riquadrati (box model)

Figura 2 Le specifiche sui CSS in corso d'elaborazione presso il W3C

High Priority	Current	Upcoming
CSS Level 2 Revision 1	Working Draft	Last Call
Selectors Level 2	Last Call	Last Call
CSS Module Profile 1.0	Candidate Recommendation	Last Call
Medium Priority	Current	Upcoming
CSS Namespaces	Working Draft	Last Call
Page Media Level 2	Candidate Recommendation	Candidate Recommendation
CSS Print Profile 1.0	Candidate Recommendation	Candidate Recommendation
CSS Values and Units Level 2	Working Draft	Working Draft
Quoting and Escaping	Working Draft	Working Draft
CSS Text Level 2	Working Draft	Working Draft
CSS Text Level 2 Level 2	See Text	Working Draft
CSS Ruby	Candidate Recommendation	Working Draft
CSS Generated Content for Paged Media	Working Draft	Working Draft
CSS Backgrounds and Borders Level 2	Working Draft	Working Draft
CSS Forms Level 2	Last Call	Working Draft
CSS Box Model Level 2	Working Draft	Working Draft
CSS Multiple Layouts	Working Draft	Working Draft
CSS Advanced Layouts	Working Draft	Working Draft
Media Queries	Candidate Recommendation	Proposed Recommendation
CSS Speech Level 2	Working Draft	Working Draft

MAGGIORE CONTROLLO SUI CONTENUTI

I CSS2 forniscono un controllo più preciso sulla visualizzazione di contenuti alternativi rispetto al semplice HTML. I selettori CSS2 danno infatti accesso ai valori degli attributi, spesso usati per fornire contenuti alternativi o integrativi. Nei CSS2, i valori degli attributi possono così essere resi in un documento insieme con il contenuto primario degli elementi. Ma i CSS2 sono anche un potente strumento di design. Ne è una prova il progetto CSS Zen-

garden (www.csszengarden.com), basato interamente sull'impaginazione senza tabelle. I risultati visivi sono strabilianti, ma altrettanto evidenti sono i risultati in termini di accessibilità dei contenuti: disabilitati i CSS nel browser, tutti i progetti di CSS Zengarden rivelano un contenuto ancora perfettamente fruibile, perché organizzato correttamente, nel rispetto della semantica, con un codice HTML lineare e pulito (Figura 4)

strutturali per i loro previsti effetti stilistici (come ad esempio usare BLOCKQUOTE per ottenere un testo rientrato e TABLE per il posizionamento e l'allineamento). Grazie ai fogli di stile, gli autori possono evitare anche il cosiddetto "abuso di immagini", per esempio l'uso di immagini invisibili di 1 pixel per posizionare con precisione i contenuti. Ciò è importante per l'accessibilità: entrambi gli abusi possono avere infatti conseguenze deleterie per chi naviga con un sintetizzatore vocale, perché rischiano di rendere caotica e incomprensibile la struttura dei contenuti. I CSS2 significano anche flessibilità e personalizzazione: permettono agli utenti di sovrascrivere con precisione gli stili definiti dall'autore, sostituendoli con un foglio di stile utente che impone ai documenti i caratteri e i colori preferiti dal lettore (un'ancora di salvezza per chi ha difficoltà a leggere una pagina con i caratteri e i colori scelti dall'autore). I CSS2 inoltre supportano sia i contenuti generati automaticamente (come numeri, testo, marcatori ecc.), che possono aiutare gli utenti a non perdere l'orientamento all'interno della pagina, specie in caso di contenuti disposti in lunghi elenchi di struttura ripetitiva, sia i fogli di stile acustici (*aural cascading style sheets*), che definiscono come un documento verrà reso in forma parlata. Essi permettono all'autore di specificare il volume del contenuto parlato, i suoni in sottofondo, le proprietà spaziali per il suono, ed altre proprietà che possono aggiungere effetti alla parola sintetizzata, così da fornire un corrispettivo acustico delle varie soluzioni di stile attribuibili al testo in modalità grafica.

I LIMITI DEI CSS2

Tuttavia vi sono dei limiti nei CSS2 che non possono essere taciuti. Molte proprietà dei



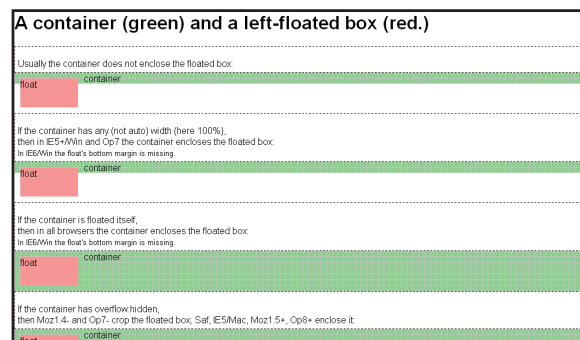
Figura 4 - Uno dei layout, disponibili sul sito **CSSZenGarden** (www.csszengarden.com), che dimostrano le capacità di rappresentazione grafica dei fogli di stile

CSS2, come alcuni valori della proprietà **display** e alcune proprietà relative ai caratteri, sono state rimosse nella successiva revisione **2.1** proprio per la mancata implementazione da parte dei browser (vedi box *Fogli stile e Browser*), oppure vengono mantenute ma senza un reale supporto (come le proprietà per i fogli di stile acustici). Un limite più sottile dei CSS2, e anche della loro revisione, i **CSS2.1**, sta nella mancanza di proprietà specifiche per gestire impaginazioni multicolonna: non vi sono strumenti per dire a un browser di suddividere una pagina in due, tre o più colonne. Di qui la necessità di «arrangiamenti» ad hoc da parte degli sviluppatori: per esempio l'uso e l'abuso della proprietà **float**, nata per inserire immagini fluttuanti in un blocco di testo, che è stata piegata a generare colonne, con il pericolo di cadere in un errore analogo a quello dell'abuso di tabelle d'impaginazione nel codice HTML. Interessanti in tal senso gli interventi di *David Baron* (<http://dbaron.org/log/2005-12>), sviluppatore di Mozilla, e di *Bruno Fassino* (<http://gabrielromanato.altervista.org/css/intervista-a-bruno-fassino/>), esperto italiano di CSS, che ci ricordano come i fogli di stile, allo stadio attuale, sono ancora lontani dall'essere una tecnologia matura, e che non bisogna perdere di vista l'obiettivo primario, che è quello di fornire ad autori ed utenti un controllo sempre maggiore sui documenti Web (**Figura 5**). Questi limiti hanno spinto molti autori a continuare ad usare le tabelle a scopo di impaginazione. Una precisazione qui è d'obbligo: l'uso delle tabelle per il posizionamento non è un tabu, ma è l'abuso, cioè il ricorso a decine di tabelle annidate in una sola pagina, ad es-

sere seriamente sconsigliato. Le tabelle non sono state affatto "bandite" dal W3C: semplicemente, è stato ricordato come il loro uso debba limitarsi all'impaginazione di dati tabulari; non servono per creare colonne, menu di navigazione o cose simili. Se proprio si vuole continuare ad usare le tabelle per questo scopo, il W3C raccomanda che tali tabelle conservino significato se rese in modo sequenziale.

IL FUTURO DEI FOGLI DI STILE

Dopo la revisione dei **CSS2**, riformulati nei **CSS2.1** (www.w3.org/TR/CSS21/), che tuttavia non hanno ancora raggiunto lo stadio finale di Raccomandazione ufficiale, il **W3C** si è rimesso al lavoro su una nuova versione dei fogli di stile: i **CSS3**. La novità maggiore rispetto ai CSS2 è che le specifiche, una volta raggiunto il grado di raccomandazione ufficiale, saranno suddivise in moduli. Questa suddivisione è stata fatta pensando ai diversi tipi di browser esistenti (visuali, acustici, ecc.). Per esempio il modulo acustico interesserà i browser di tipo acustico, mentre quello sui colori o sui bordi quelli di ti-



FOGLI STILE E BROWSER

I browser oggi più diffusi non implementano ancora completamente i **CSS2**, anche se il numero delle proprietà supportate è ormai considerevole. L'implementazione continua a progredire, ma restano differenze non da poco tra browser e browser. Ecco dunque un limite dei **CSS2**: l'implementazione e il supporto da parte dei browser. La mente corre subito a Internet Explorer, che oltre ad essere il browser più diffuso è anche quello che dà più problemi agli sviluppatori nell'applicazione dei fogli di stile. A parte i bachi nell'implementazione, che sono presenti anche negli altri browser, c'è da dire che nello sviluppo di *Internet Explorer* si sono finora fatte coesistere le esigenze delle specifiche standard con quelle delle tecnologie proprietarie. Una di queste caratteristiche proprietarie è ad esempio la proprietà **hasLayout** (si veda il documento "On Having Layout", disponibile in traduzione italiana su <http://gabrielromanato.altervista.org/css/onhavinglayout.html>), che influenza pesantemente la resa finale di un documento a cui sono stati applicati degli stili con i **CSS**, soprattutto per quello che riguarda il posizionamento. L'uscita di *Internet Explorer 7* fa sperare in un migliore supporto ai fogli di stile, nonché in una risoluzione dei problemi più gravi che affliggono il browser di casa Microsoft.

Figura 5
Esperimento dimostrativo delle proprietà dei fogli di stile, tratto dal sito di **Bruno Fassino** (<http://brunildo.org/test/>)

po visuale. I browser supporteranno così solo i moduli che servono realmente alla rappresentazione dei contenuti per i quali sono specializzati. Per ogni modulo scelto, il supporto dovrà però essere totale. I **CSS3** risolveranno alcuni dei problemi più comuni per i designer, come la possibilità di inserire immagini di sfondo multiple, di dimensionare tali immagini, di arrotondare i bordi senza far ricorso alle immagini, di creare ombre per i riquadri (**box**) e, finalmente, di avere delle proprietà per la gestione di documenti multicolonna. In questo modo il controllo sulla pagina sarà totale... anzi, potrebbe essere totale. Ancora una volta, infatti, come già per i **CSS1** e i **CSS2**, il problema principale sarà il supporto dei browser ai moduli e alle proprietà dei **CSS3**. Attualmente alcuni browser (come *Opera*) hanno già cominciato a supportare alcune proprietà dei **CSS3** e questo fa ben sperare per il futuro. Quello che si era aperto come un contrasto, potrebbe risolversi in un felice connubio.

Michele Diodati
Gabriele Romanato