

HTML, CGI, JavaScript e Database

Lucio Miranda

Lucio.Miranda@inetitalia.com

www.inetitalia.com

copyright © 2002. Tutti i diritti riservati. Vietata la riproduzione, anche parziale, senza il consenso scritto dell'autore.

Un'altra sezione dedicata al "fai da te" di Internet. Questa volta, invece degli strumenti, prendiamo in considerazione i linguaggi (o meglio, gli script) che costituiscono la struttura portante della rete.

Nel capitolo verranno citate risorse online specifiche per ognuno degli argomenti trattati. Due siti, pero', hanno una validita' generale in quanto trattano tutti gli aspetti di programmazione per Internet:

- <http://www.irt.org/>
In inglese. Letteralmente centinaia di articoli e lezioni su HTML, JavaScript, CGI, ColdFusion etc.
- <http://www.html.it/>
L'equivalente italiano di quanto sopra.

HTML

Le pagine web si scrivono in HTML. L'avvento dei programmi descritti nel capitolo precedente ha liberato dalla necessita' di conoscere l'HTML per mettere insieme un sito: ci pensa il programma. Il mio consiglio e' comunque quello di imparare un po' di HTML per saper intervenire per controllare il risultato finale del lavoro di programmi quali NOF, VisualPage o DrumBeat2000.

"Spiare" l'HTML delle pagine che ci piacciono e' una fonte di apprendimento enorme ma le risorse online non mancano:

- <http://www.virgilio.it/canali/computer/itinerari/011/>
Italiano. Raccolta di siti sull'HTML. Ce n'e' per tutti i gusti.

JavaScript (JS)

JavaScript e' uno script sviluppato da Netscape. E' ormai arrivato alla versione 1.3 ma consiglio di non andare oltre la 1.2 per motivi di compatibilita' tra browser (LINK: Compatibilita' tra Browser). JavaScript si integra con l'HTML, infatti i codici in JavaScript vengono scritti frammischiandoli con quelli dell'HTML, e fanno fare al browser azioni che non sarebbero possibili usando solo l'HTML. I Roll-Over (icone che cambiano colore quando ci si passa sopra con il mouse) sono uno degli esempi classici della funzionalita' di JS. JS puo' fare pero' cose molto piu'

complicate, come "servire" i cosiddetti "cookies" o validare i campi di un formulario elettronico.

Vale la pena impraticarsi con le nozioni di base di JS in modo tale da poter sfruttare gli script che sono a disposizione gratis online (ovvero: se lo script non e' esattamente come lo voglio, almeno so metterci le mani per adattarlo).

Alcune risorse online oltre a quelle citate in apertura sono:

- <http://www.virgilio.it/newsgroup/cgi/list?group=it/comp/lang/javascript> Italiano. Gruppo di discussione.
- <http://webreference.com/js/> Inglese. Collezione di script preconfigurati. Da visitare dopo aver acquisito le nozioni di base.
- <http://javascript.internet.com/> Inglese. Include un gruppo di discussione e vari libri sull'argomento che si possono acquistare online.

Un sottoprodotto di JS sono i Bookmarklet, script in JS molto semplici e di uso immediato, che svolgono diverse funzioni pratiche (ad esempio mandare l'indirizzo della pagina via email cliccando un link; oppure rimuovere lo sfondo, cambiare colore allo sfondo, cambiare colore al testo etc.). Per saperne di piu' visitare: <http://www.bookmarklets.com/>

CGI/Perl

L'HTML non puo' fare tutto e spesso servono dei programmi che siano fatti partire dal visitatore del sito ma che vengano eseguiti sul server dove il sito e' ospitato. Alcuni linguaggi vanno meglio di altri per scrivere programmi del genere. Perl, in particolare, e' uno script che risponde benissimo agli input generati da una pagina web (ad esempio un formulario) perche' elabora facilmente le stringhe di caratteri (che e' poi quello che un formulario elettronico invia al server).

CGI (Common Gateway Interface) e' il protocollo, ovvero l'insieme di regole, che definisce come i server Web eseguono e scambiano dati con programmi esterni. CGI non e' quindi un programma, ma detta le regole cui i programmi (Perl, ad esempio) devono sottostare per lavorare con un server.

Sia Perl che CGI non sono letture serali e non e' il caso di mettersi a studiarli (a meno che non siate determinati a diventare dei programmatori). Come per tutto il resto, pero', vale la pena acquisire familiarita' con i concetti base per capire come funzionano le cose e come intervenire sugli script disponibili online nel caso che non siano al 100% cio' di cui abbiamo bisogno.

Come per JavaScript, l'obiettivo e' di poter capire script gia' esistenti per modificarli a secondo delle nostre esigenze (attivita' estremamente piu'

facile che non quella di scrivere degli script da zero). In rete esistono centinaia di siti che offrono script già pronti e lezioni su CGI, Perl etc.

- <http://www.cgi-resources.com/>
Inglese. Un mostro sacro del CGI online con quasi 2500 script a disposizione.
- <http://www.worldwidemart.com/scripts/>
Matt's Script Archive, uno dei primi siti ad offrire scripts gratis. In particolare si può scaricare FormMail, lo script che permette di ricevere via email i dati inseriti in un formulario elettronico. Pronto per l'installazione e con chiare spiegazioni (in inglese) per l'installazione.
- <http://www.cgi-resources.com/Programmers/>
Una lista di 6 pagine di programmatori pronti ad aiutare!

Directory CGI

E' la directory su cui tipicamente avete la liberta' di installare tutti (quasi) i programmi che volete per farli elaborare dal server.

Se il vostro ISP non vi da' accesso ad una directory CGI, mentre ne cercate un altro, potete ricorrere a delle societa' online che gratuitamente mettono a disposizione il loro server dove sono stati installati dei programmi per assolvere diverse funzioni:

- www.cgi-free.com
 - Rotazione immagini. La pagina mostra un'immagine diversa ogni x ore.
 - Menu' di navigazione a tendina. Il visitatore sceglie una pagina dal menu' a tendina (il drop-down menu in inglese ..) e viene ridiretto sulla pagina scelta.
- <http://www.response-o-matic.com/refer2.htm>
 - Elaborazione dei formulari elettronici. Il visitatore riempie un formulario sul vostro sito, lo invia e voi ricevete tutti i dati via email.

Database

Qualsiasi operazione di EC prima o poi dovrà integrare l'uso di un database con quello del proprio sito. Direi che nel lungo periodo qualsiasi operazione di EC di successo avrà un database connesso al proprio sito permettendo così di fare il salto di qualità da sito statico a sito dinamico, ovvero un sito che cambia a seconda degli input che riceve. Alcuni esempi:

- Il database è collegato al magazzino e quando le scorte di un certo prodotto sono basse elimina la pagina internet dove quel prodotto è venduto.
- Il visitatore di un sito è interessato ai concessionari di zona per comprare un certo prodotto e digita la provincia di suo interesse. Il database crea "al volo" una pagina contenente solo i concessionari di zona della provincia richiesta.
- Un catalogo di parti di ricambio per auto online usa un database per permettere al visitatore di individuare il pezzo giusto tra le decine di migliaia di pezzi a magazzino. Il cliente digita online la marca dell'auto, il modello, l'anno e la categoria di prodotto (impianto elettrico, trasmissione, motore, impianto frenante, impianto di raffreddamento, illuminazione etc.) e il database gli presenta tutti e solo i prodotti pertinenti alla sua scelta.

Le applicazioni sono illimitate e si adattano ai bisogni del momento. ColdFusion (www.allaire.com) e ASP (Microsoft) sono due tra i migliori script che aiutano a gestire i database collegati ad un sito. In pratica rendono il lavoro molto più veloce e facile da concepire. Imparare a scrivere in ASP o ColdFusion non è cosa da poco e non consiglio di provarci a meno che non ne vogliate fare la vostra professione. Ciò che vorrei invece fare è dare qualche consiglio su come lavorare con i programmatori chiamati a costruire un'applicazione database da integrare al vostro sito:

- Partite dal risultato. ("Voglio che il database presenti al visitatore solo le parti di ricambio che sono disponibili a magazzino in quel momento per il modello di auto richiesto").
- Specificate chiaramente quali sono le variabili in gioco. Questo richiede un vostro lavoro preventivo di analisi. (Come si scelgono le parti di ricambio di un'auto: prima si individua la marca, poi il modello, poi l'anno, poi la categoria di prodotto. Quanti marche ho? Quanti modelli per ogni marca? Quante categorie per ogni modello? Ogni modello presenta le stesse categorie? etc.). Si tratta di evidenziare il flusso logico della richiesta pensando a tutte le possibili variazioni e ed eccezioni.

- Elencare minuziosamente tutte le variabili con le scelte ad esse associate (Marca FIAT -> Modello: 124 – 125 - 126 - 128 – 131 etc).
- Disegnare (a mano) un digramma di flusso delle scelte, illustrarlo e discuterlo con il programmatore.
- Fare la stessa cosa per il risultato. Come volete che sia presentato il risultato? Quali variabili/informazioni dovrà riportare?
- Specificare se ci sono accessi riservati ai dati (sia per consultarli che per introdurli nel database) e quali sono le condizioni di accesso (parola d'ordine + userId; solo ID; chi rilascia le parole d'accesso; come fare se un utente dimentica la sua etc.).
- Tenere presente che la fase piu' critica e' quella del disegno del database. Se in questa fase qualche aspetto viene dimenticato o sottovalutato correggere la situazione e' costoso sia in termini di tempo che di costo.