

Il mod_rewrite e la magia di riscrivere gli URL (prima parte)



In questo articolo vedremo le basi di una tecnica molto diffusa ed utile ovvero *l'url rewrite*. Si tratta di una funzionalità di Apache resa possibile dal modulo *mod_rewrite*, implementabile e configurabile tramite una nostra vecchia conoscenza: i files *.htaccess*. Il maggiore beneficio di riscrivere gli url é una **migliore indicizzazione da parte dei motori di ricerca**. Pare infatti che gli [spider](#) siano "disturbati" dalle query string (*?id=34&type=5&mod=user*), in particolare se passano molti parametri.

Sei pronto dunque a calarti in questo misterioso ed affascinante campo del web?

Come avviene la traduzione dell'url?

Poniamo che nel nostro sito, la pagina *prodotti.php* mostri un prodotto in base all'id passato nell'url.

Questa pagina mostra i dettagli del prodotto salvato nel database con id=79.
Con una semplice regola, possiamo riscrivere l'url in questo modo:

Quello che dobbiamo fare é *dire* ad Apache:

Quando trovi nell'url *prodotto* seguito da un *trattino* seguito da un *numero* seguito da *.html*, richiedi la pagina *prodotti.php?id=quelNumero*

Come avrai intuito, bisognerà lavorare con le [espressioni regolari](#). Se ne sei a digiuno, non ti

preoccupare. [Giustino](#) ha scritto [un'ottima guida](#) al riguardo.

Come dichiarare una regola?

Il modello generale per dichiarare una regola di rewrite é il seguente:

In primo luogo va inizializzato il motore del mod_rewrite con la dichiarazione *RewriteEngine On*. In seguito va dichiarata la regola (o le regole).

Ma passiamo alla pratica ed implementiamo quanto proposto nel paragrafo precedente. Iniziamo a scrivere la pagina *prodotti.php* che realizzeremo semplicemente così:

Tanto basterà per verificare il corretto funzionamento della nostra regola.

Ora vediamo il file *.htaccess* che dovrà essere salvato nella stessa (o gerarchicamente superiore) cartella del file *prodotti.php*.

Quanto dice questa regola, é esattamente quello che abbiamo visto in precedenza: Quando trovi nell'url qualcosa che inizia con (^ significa inizio della stringa) *prodotto* seguito da un *trattino* seguito da un *numero* di una o più cifre (+ significa uno o più) e che termina (\$) significa fine della stringa) con *.html*, richiedi la pagina *prodotti.php?id=quelNumero*

\$1 si riferisce infatti a quanto é stato trovato dalla prima (e in questo caso unica) sottostringa.

Ora impostando come url:

Dovresti visualizzare a video la seguente scritta:

"Questa pagina mostra il prodotto numero 34."

Un esempio con due parametri

Poniamo ora che la nostra pagina *prodotti.php* dipenda da due parametri: l'id e la categoria del prodotto. L'URL originale potrebbe essere questo:

Mentre riscritto potrebbe avere questo aspetto:

Vediamo dunque una possibile regola per poi commentarla:

Cosa dice questa regola?

Se trovi una stringa che inizia con uno o più *caratteri* seguiti da / seguiti da *prodotto* seguito da un *trattino* seguito da un *numero* di una o più cifre seguito da *.html*, richiedi la pagina `prodotti.php?category=quelloCheHaiTrovatoNellaPrimaSottostringa&id=quelloCheHaiTrovatoNellaSecondaSottostringa`.

Modifichiamo il file *prodotti.php* in modo da poter verificare il funzionamento anche di questo secondo esempio in questo modo:

Ora digitando l'url:

Visualizzerai a video la seguente scritta

Questa pagina mostra il prodotto numero 12 appartenente alla categoria bevande.

Conclusione

In questo articolo abbiamo visto le basi dell'url rewrite. Ti sarai reso conto di quanto sia importante una buona conoscenza delle espressioni regolari per poter padroneggiare questa tecnica. Nel prossimo articolo, vedremo un'applicazione pratica molto utile in campo SEO, ovvero un rewrite **simile a quello che implementa wordpress con i *permalink***.

E tu, hai degli esempi di url rewrite da mostrare?