

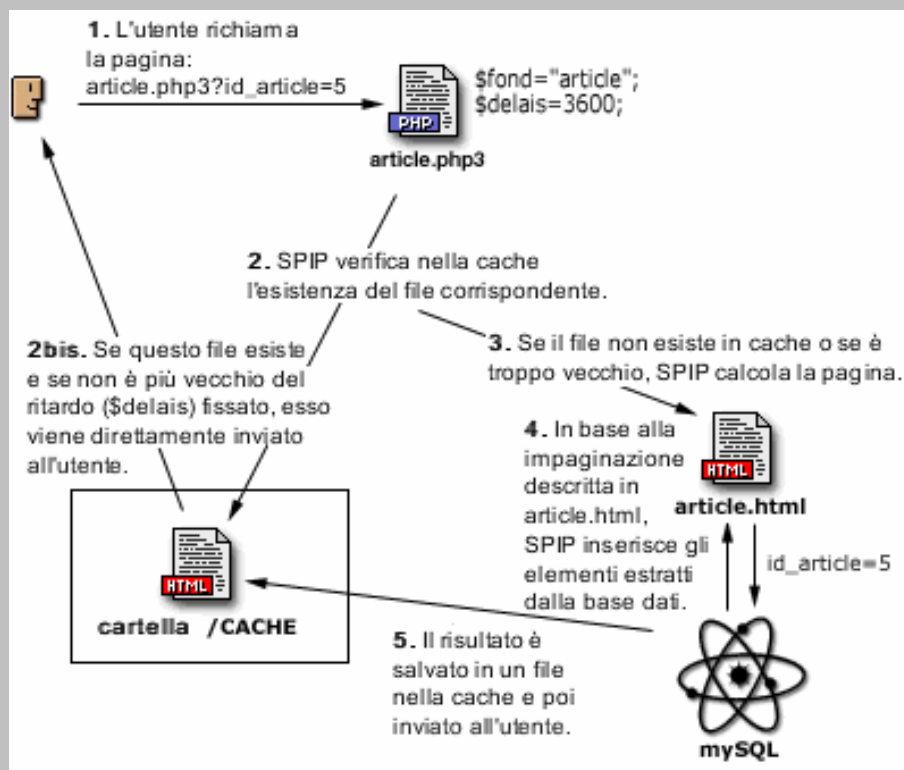
L'approccio di SPIP alla creazione dei modelli

- **SPIP** adotta un semplice linguaggio di impaginazione che consente di poter creare modelli di elevata complessità **senza conoscenze specifiche di PHP o SQL**, innalzando la curva di apprendimento ad un livello accettabile dalla maggior parte dei webmaster
- **I modelli sono riutilizzati** per tutti gli elementi (articoli, rubriche, brevi) del sito.
- SPIP adotta **un sistema di cache** che consente di abbattere il carico di lavoro sul server e permette al sito di essere raggiungibile anche in caso di blocco del server mySQL

Il funzionamento della cache

Due file per ogni modello: un file PHP ed uno HTML

2. Il file PHP ha il compito di selezionare il file HTML corrispondente e di determinarne la durata in cache
3. Il file HTML contiene il modello vero e proprio



Il contenuto del file PHP

Il file PHP è estremamente semplice.

Sono 2 le variabili importanti da conoscere:

3. `$fond`, corrisponde al nome del file HTML del modello corrispondente
4. `$delais`, la durata in secondi della pagina in cache

Ecco un esempio pratico (in rosso le parti da modificare):

```
<?php
    $fond = "article";
    $delais = 24 * 3600;

    include ("inc-public.php3");
    ?>
```

Il contenuto del file HTML

Perché un altro linguaggio?

Perché è più semplice che imparare il PHP e l'SQL e perché vi farà guadagnare moltissimo tempo.

Si tratta di un **linguaggio a marcatori**, simile all'HTML, che ha il compito di selezionare gli elementi dal database e di inserirli nel Proprio codice HTML

Riutilizzare i modelli, ma non sempre

Poter creare un **modello universale**, adatto a più pagine, è sicuramente utile.

SPIP estende questo concetto e permette di richiamare modelli specifici per determinate rubriche o rami di rubriche.

Il modello standard delle rubriche: **rubrique.html**

Il modello per la rubrica 2 e le sue sottorubriche: **rubrique-2.html**

Il modello per la sola rubrica 2 : **rubrique=2.html**

I cicli

E' l'elemento fondamentale della sintassi di **SPIP**.

Affinché **SPIP** possa pubblicare il contenuto è necessario estrarre dal database dei dati. *Il **ciclo** ha il compito di selezionare questi dati (una serie di articoli, rubriche etc.) e di inserirli nel modello secondo i bisogni del webmaster, uno alla volta, iterando tra gli elementi corrispondenti.*

Inoltre il ciclo permette di *specificare una visualizzazione alternativa* nei casi in cui nessun elemento corrisponda ai criteri di selezione adottati.

(Es: Seleziona i primi 10 articoli della rubrica numero 3)

```
<BOUCLE_articoli(ARTICLES) {id_rubrique=3} {0,10}>
```

I segnaposti

Sappiamo che un articolo è composto da varie parti (titolo, sottotitolo, testo, etc..)

Dopo aver selezionato gli elementi attraverso il ciclo si deve specificare quali parti vogliamo pubblicare nel modello.

Il **segnaposto** individua, *all'interno dei cicli*, cosa debba essere pubblicato.

Es: Ciclo dei primi 10 articoli della rubrica numero 3

Per ogni articolo mostra (un segnaposto per ogni voce):

Il titolo, il sottotitolo, il logo, il corpo dell'articolo

```
<BOUCLE_articoli(ARTICLES) {id_rubrique=3} {0,10}>
```

```
#TITRE, #SOUSTITRE, #LOGO_ARTICLE, #TEXTE
```

```
</BOUCLE_articoli>
```

La sintassi dei cicli

Sintassi semplificata (BOUCLE significa ciclo in francese):

```
<BOUCLE_nome_ciclo(TIPO_ELEMENTO) {criterio1} {criterio 2}>  
    Contenuto del ciclo (HTML e segnaposti)  
</BOUCLE_nome_ciclo>
```

Cosa è necessario scrivere?

- Il **nome** del ciclo
- Il **tipo di elemento** da selezionare (articolo, rubrica, breve)
- I **criteri di selezione** (i primi dieci, l'ordine di pubblicazione, la rubrica di appartenenza, etc..)
- Il **contenuto del ciclo**
SPIP elaborerà il contenuto del ciclo per ogni elemento che corrisponde ai criteri impostati

La sintassi estesa

<B_nome_ciclo>

parte opzionale prima (HTML)

<BOUCLE_nome_ciclo(TIPO_ELEMENTO) {criterio1} {criterio 2}>

Contenuto del ciclo (HTML e segnaposti)

</BOUCLE_nome_ciclo>

parte opzionale dopo (HTML)

</B_nome_ciclo>

parte condizionale, nessun elemento trovato (HTML)

<{//B_nome_ciclo>

Esempio: Mostra il titolo dei primi 10 articoli contenuti nella rubrica 3, altrimenti mostra “Nessun articolo trovato”

<B_articoli>

<BOUCLE_articoli(ARTICLES) {id_rubrique=3} {0,10}>

#TITRE

</BOUCLE_articoli>

</B_articoli>

Nessun articolo trovato

<{//B_articoli>

Il contesto dei criteri

I **criteri** permettono di specificare *le condizioni che gli elementi devono possedere per essere selezionati*.

Se volessimo selezionare gli articoli della rubrica corrente, dovremmo necessariamente conoscere quale essa sia.

Tali informazioni vengono definite “**contesto**”

Le informazioni di contesto possono essere fornite da:

- **l'URL** es. ([rubrique.php3?id_rubrique=3](#)) La rubrica del contesto è la numero 3
- **dal contesto fornito da altri cicli**

```
<BOUCLE_rubrica(RUBRIQUES) >  
  <BOUCLE_articoli(ARTICLES) {id_rubrique}>  
    #TITRE  
  </BOUCLE_articoli>  
</BOUCLE_rubrica>
```

Il primo ciclo seleziona, una alla volta, tutte le rubriche (non è stato impostato alcun criterio di selezione). Il secondo ciclo seleziona, di volta in volta, gli articoli della rubrica presente nel contesto creato dal primo ciclo.

La sintassi dei segnaposti

I **segnaposti** pubblicano le informazioni di un determinato elemento (articolo, rubrica, breve) selezionato mediante l'uso di un ciclo. I *segnaposti sono sempre preceduti dal simbolo # e sono scritti in maiuscolo.*

La sintassi semplificata:

```
<BOUCLE_articolo(ARTICLES) {id_article=3}>  
  <h1>#TITRE</h1>  
  <div>#SOUSTITRE</div>  
</BOUCLE_articolo>
```

SPIP seleziona l'articolo numero 3 e, grazie all'uso dei segnaposti, pubblica il titolo (**#TITRE**) ed il sottotitolo (**#SOUSTITRE**)

SPIP quindi sostituisce al segnaposto il contenuto del campo specificato prelevandolo dai campi dell'elemento correntemente selezionato dal ciclo.

La sintassi avanzata dei segnaposti

Molto spesso è necessario modificare il codice HTML prodotto da **SPIP** in base all'esistenza in database di un determinato campo.

Es:

```
<B_articolo>  
<div class="articolo">  
<BOUCLE_articolo(ARTICLES) {id_articolo}>  
    #SURTITRE <br />  
    #TITRE  
</BOUCLE_articolo>  
</div>  
</B_articolo>
```

Che succede se l'autore non ha riempito il campo "occhiello" (**#SURTITRE**)?
SPIP inserirà nel codice un salto di linea non voluto. La sintassi estesa ci aiuta a formattare correttamente il codice HTML.

[testo opzionale prima (**#SEGNAPOSTO**) testo opzionale dopo]

La sintassi avanzata dei segnaposti

Riscriviamo il ciclo utilizzando la *sintassi avanzata* dei segnaposti per inserire il tag `
` sono in caso di necessità, ovvero se il campo **#SURTITRE** è stato riempito.

```
<B_articolo>
<div class="articolo">
<BOUCLE_articolo(ARTICLES) {id_articolo}>
    [(#SURTITRE) <br />]
    #TITRE
</BOUCLE_articolo>
</div>
</B_articolo>
```

Segnaposti non ambigui

Alcuni segnaposti sono supportati da più tipologie di cicli.

Il segnaposto **#TITRE** può essere inserito all'interno di cicli sulle rubriche o sugli Articoli, sulle brevi etc..

In maniera predefinita un segnaposto si riferisce ai dati selezionati dal ciclo che lo contiene. *Tuttavia è sempre possibile specificare per un segnaposto il ciclo a cui si riferisce, quando esso sia diverso da quello che lo contiene*

La sintassi è: **#_nome_ciclo:SEGNAPOSTO**

```
<BOUCLE_rubriche(RUBRIQUES) {id_rubrique}>  
  <BOUCLE_articoli(ARTICLES) {id_rubrique}>  
    #_rubriche:TITRE &gt; #TITRE  
  </BOUCLE_articoli>  
</BOUCLE_rubriche>
```

Questo ciclo mostrerà: titolo della rubrica > titolo dell'articolo

I filtri

*E' possibile trattare ulteriormente i dati provenienti dai segnaposto mediante l'uso di **filtri**, ad esempio per mostrare una stringa in maiuscolo.*

I filtri si applicano direttamente ai segnaposti in questo modo:

[**testo opzionale prima** (#SEGNAPOSTO|**filtro1**|**filtro2**) **testo opzionale dopo**]

Mostrare un titolo in maiuscolo significa scrivere:

[(#TITRE|**majuscules**)] dove **majuscules** è il filtro che opera la trasformazione della stringa in maiuscolo.

SPIP offre una vasta scelta di filtri adatti alle più comuni esigenze di impaginazione.

Alcuni filtri possono richiedere l'inserimento di parametri mediante la sintassi **filtro**{parametro}.

Es: ridimensionare un logo a 100 pixel di larghezza si scrive

[(#LOGO_ARTICLE|**reduce_image**{100,0})]

Quale contesto verrà usato?

Se un ciclo non è a sua volta contenuto in un altro ciclo, il suo contesto sarà quello fornito dall'URL. Viceversa il contesto del ciclo sarà quello stabilito dal ciclo che lo contiene.

Es: URL: www.miosito.it/rubrique?id_rubrique=3

Ciclo (mostra gli articoli della rubrica indicata nel parametro `id_rubrique` e, per ognuno di essi, mostra il titolo ed nome dei loro autori)

```
<BOUCLE_articoli(ARTICLES) {id_rubrique}> il contesto è quello dato dall'URL
  #TITRE
  <BOUCLE_autori(AUTEURS) {id_article}> il contesto è quello fornito dal ciclo _articoli
    #NOM
  </BOUCLE_autori>
</BOUCLE_articoli>
```