

PHP et les templates

Réalisation d'un template
Niveau débutant

Sommaire

- Introduction
- Notion de template
- Comment ça marche ?
- Présentation de l'exemple
- Description en détails des fichiers
- Conclusion

Introduction

Cet article a pour but de vous initier à la création d'un template grâce à PHP. Il est principalement destiné aux débutants. Vous avez besoin de quelques connaissances de base sur les langages HTML et PHP. Si vous n'avez aucune connaissance ni en HTML, ni en PHP, je vous conseille de commencer par lire quelques cours et tutoriels sur HTML et PHP pour vous familiariser avec ces langages. L'exemple complet est disponible au téléchargement.

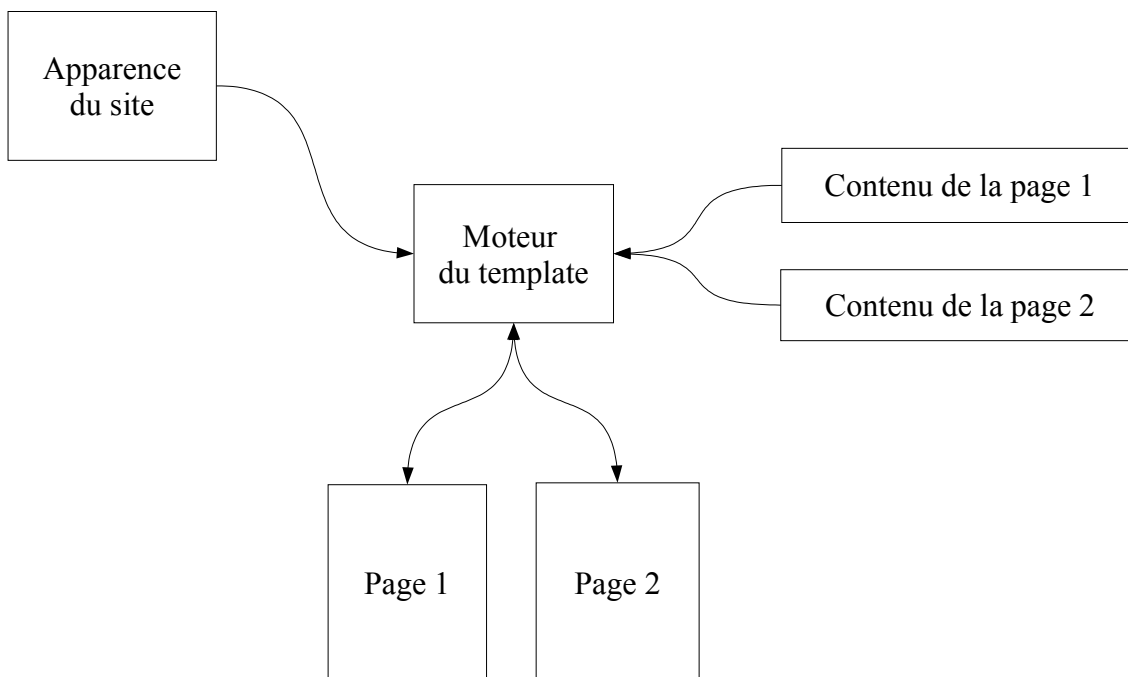
Notion de template

Si l'on devait définir le template, on commencerait tout simplement par une traduction. «Template» est traduit en français par «patron». On prendra évidemment «patron» dans le sens de «modèle». Ce concept de template est né du soucis de faciliter la création des documents HTML et de l'envie de pouvoir modifier l'apparence d'un site Web, je dirais presque, un peu comme l'on change de chemise.

C'est là où le concept de template rejoint sa traduction en modèle. On va séparer la présentation du site de son contenu. Le contenu d'une page, qui peut varier à gré (on dira dynamique), sera affiché dans un modèle d'apparence, qui lui restera fixe (on dira statique). L'utilisateur ne voit pas la différence à l'affichage de la page et la maintenance du site en est grandement facilitée.

Comment ça marche ?

Un petit schéma sera plus explicite qu'un long discours :



Pour afficher la page 1, on fait appel au moteur du template, qui va aller chercher le contenu de la page 1 et l'apparence du site, puis réaliser le mélange et retourner le tout au client. De même pour la page 2 et d'autres pages.

On s'aperçoit alors que les applications du template sont multiples.

Les plus simples seront :

- modifier l'apparence de la totalité du site, sans devoir éditer chaque page
- ajouter, modifier ou retirer une page en agissant uniquement sur son contenu

Pour les plus complexes, on pourra aller jusqu'à :

- proposer à l'utilisateur de personnaliser le site en lui laissant le choix parmi plusieurs apparences
- proposer des contenus traduits en plusieurs langues
- créer des apparences qui pourraient varier en fonction de la période de l'année (pourquoi pas une apparence «spécial Noël»)

Présentation de l'exemple

Pour illustrer le template et son utilisation, nous allons créer un mini-site pour une entreprise fictive : la S.A Jacky&Co. Nous utiliserons un template pour la structure du site et PHP pour le moteur du template.

Le plan du site, vu par le client, se découpe ainsi :

- Accueil (la page par défaut)
- Présentation de l'entreprise
- Catalogue
- Contacts

Description en détails des fichiers

Le site se compose d'un index (index.php), du moteur (moteur.php), de l'apparence (skin.html) et des contenus des 4 pages (accueil.inc, presentation.inc, catalogue.inc, contact.inc).

skin.html

Le fichier skin.html constitue l'apparence du site. Il contient du code HTML ainsi que des balises situées aux emplacements choisis pour afficher le contenu ou diverses informations (date du jour, etc...) et à la syntaxe prédéfinie.

Pour les balises de notre exemple, nous choisirons la syntaxe suivante :

- «[CONTENU]» pour l'emplacement du contenu de la page
- «[DATE]» pour l'emplacement de la date du jour

Ce qui nous donne :

```
<HTML>
<HEAD>
<TITLE>Bienvenue sur le site web de la S.A. Jacky&Co</TITLE>
</HEAD>
<BODY>
<TABLE WIDTH=100% BORDER=0 CELLPADDING=0 CELLSPACING=0>
  <TR>
    <TD COLSPAN=3 WIDTH=75% ALIGN=LEFT>
      <FONT SIZE=7>S.A. Jacky&Co</FONT>
    </TD>
    <TD WIDTH=25% ALIGN=RIGHT>
      <FONT SIZE=4>[DATE]</FONT>
    </TD>
  </TR>
  <TR>
    <TD WIDTH=25% HEIGHT=100 ALIGN=CENTER>
      <A HREF="index.php/"><FONT SIZE=6>Page d'accueil</FONT></A>
    </TD>
    <TD WIDTH=25% ALIGN=CENTER>
      <A HREF="index.php?page=presentation"><FONT SIZE=6>Pr&eacute;sentation de
Jacky&Co</FONT></A>
    </TD>
    <TD WIDTH=25% ALIGN=CENTER>
      <A HREF="index.php?page=catalogue"><FONT SIZE=6>Notre catalogue</FONT></A>
    </TD>
    <TD WIDTH=25% ALIGN=CENTER>
      <A HREF="index.php?page=contact"><FONT SIZE=6>Nous contacter</FONT></A>
    </TD>
  </TR>
  <TR>
    <TD COLSPAN=4 WIDTH=100% ALIGN=CENTER>
      <BR><BR><FONT SIZE=4>[CONTENU]</FONT><BR>
    </TD>
  </TR>
  <TR>
    <TD COLSPAN=4 WIDTH=100% HEIGHT=40 ALIGN=CENTER>
      Copyright &copy; 2004 S.A. Jacky&Co – Tous droits r&eacute;serv&eacute;s.
    </TD>
  </TR>
</TABLE>
</BODY>
</HTML>
```

moteur.php

Le fichier `moteur.php` contient le script PHP du moteur du template, que nous allons décrire ici. Nous allons créer une fonction nommée `template()` à qui nous ferons passer le nom de la page à afficher en paramètre.

La fonction va :

- S'occuper du skin : on ouvre le fichier `skin.html` (fonction `fopen()`), on extrait le code HTML (fonction `fread()`), on le stocke dans la variable `$affichage` et on referme le fichier (fonction `fclose()`).
- S'occuper du contenu de la page : on contrôle la variable `$page` (structure de contrôle `switch()`), on assigne le chemin du contenu à inclure à la variable `$include`, on ouvre le fichier contenant l'`include` (`accueil.inc`, `presentation.inc`, `catalogue.inc` ou `contact.inc`), on extrait le code HTML, on le stocke dans la variable `$contenu` et on referme le fichier.
- Déterminer la date du jour : on utilise la fonction `date()` et on stocke la valeur obtenue dans la variable `$date`.
- Effectuer le remplacement des variables : avec la fonction `ereg_replace()`, on détecte les variables `[CONTENU]` et `[DATE]` grâce à une expression régulière et on les remplace par les valeurs respectives de `$contenu` et `$date`.
- Retourner la chaîne `$affichage`.

```
<?php
function template($page) {
// on s'occupe du skin
$skin = "skin.html"; // chemin du fichier
$souvrir = fopen ($skin, "rb"); // ouverture du fichier
$affichage = fread ($souvir, filesize ($skin)); // lecture du fichier et stockage du skin dans une
// variable

fclose ($souvir); // fermeture du fichier

// on s'occupe du contenu
switch ($page) { // contrôle du contenu demandé par la page
case "presentation": $include = "presentation.inc"; break;
case "catalogue":   $include = "catalogue.inc"; break;
case "contact":    $include = "contact.inc"; break;
default:           $include = "accueil.inc";
}
$souvrir = fopen ($include, "rb"); // ouverture du fichier
$contenu = fread ($souvir, filesize ($include)); // lecture du fichier et stockage du contenu dans une
// variable

fclose ($souvir); // fermeture du fichier

// détermination de la date du jour au format JJ/MM/AA
$date = date("j/n/y");

// expression régulière pour le remplacement des variables
$affichage = ereg_replace("\[CONTENU]", $contenu, $affichage); // remplacement de [CONTENU]
// par le contenu demandé par la page
$affichage = ereg_replace("\[DATE]", $date, $affichage); // remplacement de [DATE] par la date du
// jour

// on retourne le code à afficher
return $affichage;
}
?>
```

moteur.php

index.php

Le fichier index.php est appelé par le client. Il contient un script PHP que nous allons décrire.

Tout d'abord, on vérifie si le paramètre «page» a été passé par l'URL. Pour cela, on contrôle l'existence de la variable \$_GET["page"]. Si elle existe, on assigne sa valeur à la variable \$page, sinon la variable \$page prend une valeur nulle.

Ensuite, on inclut moteur.php et on fait appel à la fonction template() en lui passant \$page en paramètre.

Enfin, on affiche la valeur retournée par template(), c'est à dire le code HTML de la page demandée.

```
<?php
if (!isset($_GET["page"])) { $page = ""; }else{ $page = $_GET["page"]; } // contrôle de $_GET["page"]
                                                    // et assignation d'une valeur à $page

include("moteur.php"); // inclusion de moteur.php
echo template($page); // appel à la fonction template() et affichage du résultat
?>
```

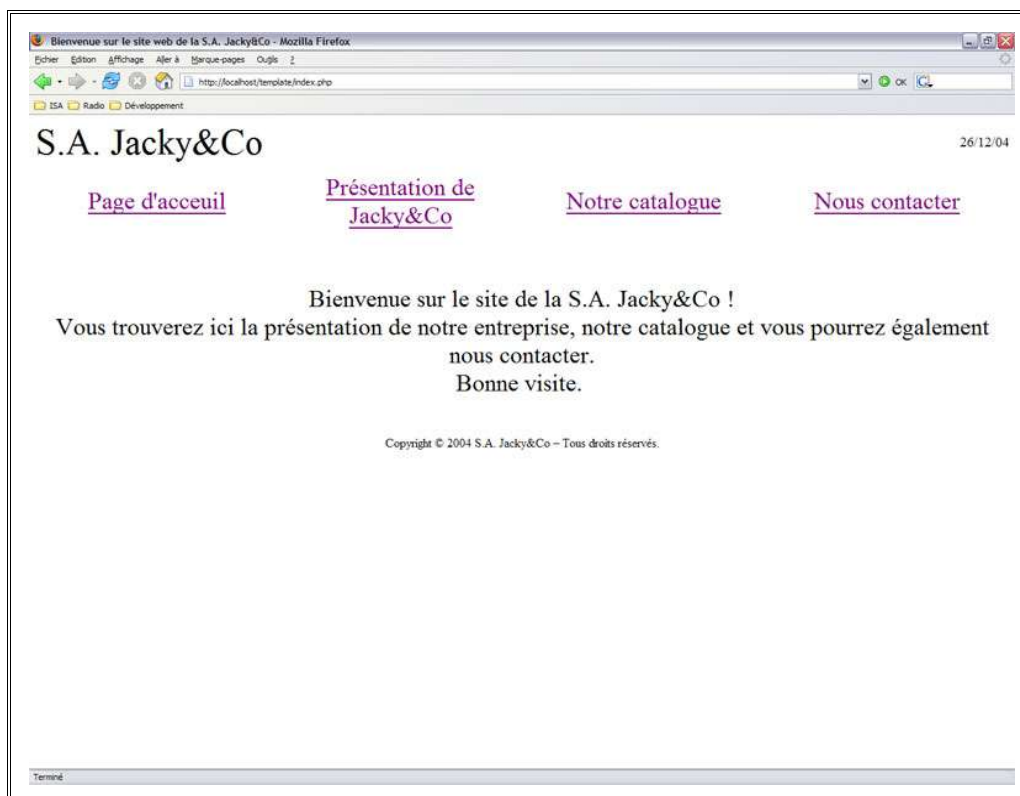
moteur.php

Une fois le tout enregistré, on appelle la page index.php et l'on peut constater en comparant avec le fichier skin.html que [DATE] a été remplacé par la date du jour et [CONTENU] par le contenu de la page d'accueil.

accueil.inc, presentation.inc, catalogue.inc, contact.inc

Ces fichiers contiennent les contenus des pages Accueil, Présentation de l'entreprise, Catalogue et Contact sous forme de code HTML. La modification de ces pages se fait simplement en mettant à jour le fichier correspondant.

Au final, l'appel du fichier index.php dans notre navigateur préféré ;-) nous donne ceci :



Conclusion

Le concept de template est utilisé très fréquemment sur internet. Le but de cet article était de vous faire comprendre le principe de fonctionnement du template à travers un exemple très simplifié. A vous maintenant d'écrire votre propre système de template, en vous inspirant de cet exemple, et de le perfectionner.

Si vous êtes intéressé par le concept du template pour votre site mais que vous ne vous sentez pas capables de développer votre propre système, je vous signale l'existence de solutions de templates utilisant PHP. Elles sont nombreuses sur le web, plus ou moins complexes dans leur fonctionnement, plus ou moins faciles à mettre en place et à utiliser. En voici une liste non exhaustive des plus connus :

- Fast Template
- PHPLib
- Vtemplate
- Smarty
- ModeliXe

Pour tous renseignements, questions ou suggestions : christophe *at* redaction-developpez *dot* com

Article publié sur :



Les sources présentées sur cette page sont libres de droits, et vous pouvez les utiliser à votre convenance. Par contre cette page de présentation de ces sources constitue une oeuvre intellectuelle protégée par les droits d'auteurs.

Copyright Sion Christophe

Aucune reproduction, même partielle, ne peut être faite de ce document et de l'ensemble de son contenu : textes, images, etc sans l'autorisation expresse de l'auteur. Sinon, vous encourez selon la loi jusqu'à 3 ans de prison et jusqu'à 300 000 E de dommages et intérêts.