Le site ouaibe de la caravane permanente

intrigeri [intrigeri@squat.net]

3 novembre 2003

Table des matières

Résumé

Pas mal de genTEs ont déjà passé un bout de temps à réfléchir ensemble sur le site ouaibe de la Karavane, et plus précisément, sur l'articulation entre *la façon dont le site apparaîtra pour les visiteureuses* et *comment qu'on fait, techniquement, pour que ça donne ça.*

J'ai ensuite réalisé une maquette de site appliquant les idées issues de ces réflexions.

Le présent texte se veut une synthèse de ces discussions collectives et de ce boulot ; il est donc, pour partie, technique. J'ai néanmoins essayé d'expliquer clairement tout ce qui ne relève pas du détail technique ; si vous ne comprenez pas un truc, demandez moi de ré-expliquer plus clairement, sans hésiter.

Il n'en reste pas moins qu'il est bien difficile dans un tel texte de satisfaire tant les débutantEs en la matière que les spécialistes de la question.

0.1 Historique de ce document

- 2003 08 23 : version initiale (compte-rendu de la première réunion sur la question)
- 2003 11 03 : fusion de la version initiale et des notes prises lors de la réalisation de la maquette de site

0.2 À faire...

- le site ouaibe et la Gazette (papier) : quelles relations ?
- étudier quels boutons de navigation on doit mettre, en fonction de la vue considérée

1 Un site ouaibe dynamique, pourquoi ?

Il y a deux types de sites ouaibes : les sites *statiques* et les sites *dynamiques*.

Dans le premier cas (statique), CHAQUE page du site doit être créée et stockée individuellement sur le serveur, une bonne fois pour toutes ; le contenu (ex : texte) et la forme (ex : 2 colonnes plus un bandeau en haut) sont donc stockées ensemble ; si l'on souhaite changer la forme (la gueule du site, quoi), il faut aller retoucher chaque page du site. Pfff. Si l'on veut une page qui donne la liste des 20 dernières niouzes de la Karavane, il faut la faire à la main, puis la faire évoluer à la main à chaque nouvelle niouze. Galère.

Dans le second cas (dynamique), on dissocie forme et fond : le contenu est stocké dans une base de données, indépendamment des fichiers qui décrivent la forme du site ; on peut donc changer la gueule de tout le site en modifiant UN fichier et non 375 fichiers ; de plus, si l'on veut une page qui donne la liste des 20 derniers trucs parus à propos de la carpothèque, hop, suffit de la programmer une fois, et au fur et à mesure du temps, le site la recalculera tout seul. Magique.

Évidemment, la seconde solution est préférable. Ça tombe bien, le truc qu'on a choisi pour faire le site (SPIP - http ://uzine.net/spip) est fait pour ça.

2 Des données et des vues

2.1 Plusieurs vues pour les mêmes données

Tout d'abord, l'unité de base du site de la Karavane, pour des raisons bassement techniques, est *l'article* ; un article, c'est du texte qui a un titre, unE auteurE, avec, en option, des images, vidéos, sons, etc.

Quand nous avons commencé à réfléchir à ce site, s'est immédiatement posée la question : "doit-on classer/rubriquer les articles par date, par lieu, ou par type d'activité ?" ; question sans réponse, évidemment. Ce dont nous avons besoin, c'est de pouvoir trier et afficher les articles, au choix, selon n'importe lequel de ces critères.

Et c'est là que se révèle à nous toute la puissance d'un site ouaibe dynamique, qui permet ça. Comment ? Ben voilà.

Grosso modo, on stocke tous les articles ensemble, en vrac. Pour chacun de ces articles, on donne quelques précisions : date, lieu, type d'activité, etc. Tout ça, la visiteureuse ne le voit pas.

Mais ensuite, on crée des pages de programmation qui savent extirper, de ce tas d'articles en vrac, ceux qui nous intéressent, pour en faire une zoulie page ouaibe que le visiteureuse verra. Je nommerai désormais une *vue* chaque façon de présenter une liste d'articles : par exemple, une vue par lieu, une vue par date, une vue par atelier, etc. Vous l'avez maintenant compris, un même article peut apparaître sur différentes vues : l'article sur le spectacle donné à Colombey-les-deux-églises le 11 mars 2007 apparaîtra dans 3 vues, par exemple : celle qui liste les activités de la Karavane dans ce charmant patelin, celle qui liste les activités de la Karavane en mars 2007, juste avant la révolution, et celle qui liste les spectacles donnés par la Karavane. Z'avez compris ?

Un autre avantage de procéder ainsi, c'est qu'une fois tout ce bazar programmé, ça roule tout seul, y'a plus qu'à remplir le site, en ajoutant des articles via une interface de publication simple style Indymedia. Puis on voit nos différentes vues s'allonger, comme par magie, à chaque nouvelle parution d'article. Et comme ça, hop, on peut faire autre chose que de l'ordinateur, youpi.

2.2 Quelles vues, quelles données ?

Bon, c'est pas tout, maintenant que vous avez compris le principe, hop hop, du concret. Quelles vues souhaite-t-on ?

2.2.1 Les trois vues de base et leurs corollaires

Les trois vues de base sont :

- une vue *Temps* (tri par date décroissante, avec des moyens de navigation vers le passé et vers le futur)
- une vue *Lieux* (tri par lieu)
- une vue *Thèmes* (tri par type d'atelier / structure de la Karavane / thématique ; ex : soudure, anti-psy, infokiosque)

Concrètement, on arrive sur le site, on clique sur "Où ?", on arrive sur une page donnant la liste des lieux/étapes de la Karavane, on clique sur "Dijon" et on arrive sur une page donnant la liste des trucs que la Karavane a fait à Dijon.

Nous avons donc besoin de deux vues corollaires des précédentes :

- une vue *Lieu* listant les articles relatifs au lieu choisi
- une vue *Thème* listant les articles relatifs au thème choisi

2.2.2 Des vues particulières pour des articles particuliers

De plus, on veut aussi pouvoir trier selon d'autres critères, donc on ajoute une classification supplémentaire, selon le type d'article : "ponctuel" (ex : une niouze), "intemporel" (ex : une fiche pratique), ou "interne" (ex : un compte-rendu de réunion). Ce qui nous donne d'autres vues :

- une vue *Documents* rassemblant les trucs "intemporels"
- une vue *Interne* rassemblant les compte-rendus et autres trucs d'organisation interne

Cette classification supplémentaire permet de plus, pour les pages des ateliers, de mettre dans une première colonne les fiches pratiques, et dans une seconde les niouzes. Par exemple. Histoire de ne pas avoir à chercher la fiche pratique de la soudure tout en bas d'une page listant les diverses créations métalliques réalisées dans tous les lieux où la Karavane est passée.

Ensuite, on a étudié le cas des articles qui ne rentreraient pas dans un atelier particulier, par exemple les articles parlant de la Karavane en général. Après réflexion, la solution la plus efficace, est d'avoir, en amont, une classification étanche entre les articles relatifs à des ateliers particuliers et les articles plus généraux. M'enfin, ça c'est de la bidouille technique, ça n'aura pas d'incidence sur la gueule du site (pour les amateurices, on peut faire ça, en SPIP, ou bien avec un groupe de mots-clés supplémentaires, ou bien en faisant deux rubriques distinctes, au lieu de mettre tous les articles dans le même tas fourre-tout).

2.2.3 La page d'accueil : une vue à elle toute seule

On n'a pas eu le temps de réfléchir à la tronche de cette page, mais tout de même celui de lister les liens et infos qui devraient s'y trouver :

- une micro-présentation de la Karavane, avec un lien vers des textes plus complets
- des liens directs vers nos différentes vues ("Où est la Karavane ?", "Que fait la Karavane ces temps ci ?", "Quelles activités sur la Karavane ?", "Fiches pratiques et autres trucs intemporels", "Interne")
- un lien vers la rubrique "Gazette"
- un lien "Soutenez la Karavane, sortez votre carnet de chèques !" ou assimilé
- une case pour taper un mot à rechercher sur le site
- un lien vers la liste de matos
- un lien vers le wiki de préparation d'étapes
- un lien "Publier" pour nous autres Karavaneureuses

3 L'implémentation de tout ce bazar

Du concret, du concret ! *Implémenter* signifie *construire une réalité informatique à partir de ce qu'on a modélisé sur le papier.*

3.1 L'hébergement

Comme expliqué dans le texte *Ce qui se cache derrière votre URL* [1], un site ouaibe est hébergé sur un serveur. J'ai bossé, pour l'instant, sur un serveur autogéré (lautre.net), sur lequel je peux créer des comptes pour bosser sur le site, plus aisément – pour l'instant – que sur squat.net, qui, *a priori*, hébergera le site une fois qu'il sera lancé.

L'adresse du truc que j'ai fait est http ://k.chutelibre.org.

3.2 La structure de la base de données

Les articles sont triés selon deux catégories, matérialisées par des *rubriques* SPIP :

- les articles relatifs à la caravane en son ensemble
- les articles relatifs à un atelier/thème particulier

On attribue de plus à chaque article, lors de sa publication, quelques mots-clés classés par groupes :

- un et un seul mot-clé du groupe "Lieux", sauf pour les articles intemporels qui n'en ont pas ;
- un mot-clé, au moins, du groupe "Thèmes", sauf pour les articles relatifs à la caravane dans son ensemble, qui n'en ont pas ;
- entre zéro et deux mots-clés du groupe "Types spéciaux d'articles", parmi "Intemporel" (fiches techniques, etc.) et "Interne" (pas encore utilisé).

3.3 Séparer le code de l'apparence : comment ça marche sous SPIP ?

De même qu'il est intéressant, comme on l'a vu dans le paragraphe 2.1, de séparer les données des programmes qui les présentent à nos yeux, il est fort utile de séparer les squelettes des feuilles de styles. Oula, cékoiça ?

[1] Darkveggy, *Ce qui se cache derrière votre URL*, 2003, http ://print.squat.net/docs/drkvg-derriere_url.txt.

Les *squelettes* sont, dirons-nous, les programmes qui vont chercher les informations dans la base de données ; en gros, ils contiennent des trucs du genre "Affiche les titres des dix derniers articles", "Affiche le menu de navigation", ces "ordres" étant bien sûr traduits dans le langage de SPIP.

Les *feuilles de styles* sont les fichiers qui décrivent la forme que doivent avoir les pages et le texte obtenus ; par exemple "Les titres seront en police de 16, gras" ou "Le menu de navigation aura un fond rose et sera placé à gauche de la page". Le langage ici employé sera celui des CSS (*Cascading Style Sheets*, qui signifie "feuille de style" en anglais).

Séparer ces deux trucs a de nombreux avantages :
- on peut modifier entièrement l'apparence du site en changeant uniquement les feuilles de styles ; cf. http ://csszengarden.com pour une petite démonstration ;
- le code source des squelettes en est considérablement allégé, ce qui le rend beaucoup plus lisible ;
- le site est compatible avec les outils tels qu'écrans braille et dictée vocale, pour les non-voyantEs.

3.4 Les squelettes

Les squelettes sont écrits en XHTML 1.0 Strict, mâtiné de langage SPIP et d'un poil de PHP ; cette décision est *uniquement* issue de mon cerveau stalinien. Concrètement, ça a un avantage : ça oblige à bien séparer squelettes et feuilles de styles.

On stocke ces fichiers, portant l'extension `.html`, dans le répertoire `squelettes` du site, ce qui permet d'utiliser différents jeux de squelettes. Pour que ce soit possible, il a fallu modifier le fichier `mes_fonctions.php3`.

Comme vous pourrez le constater en visitant le site, ce que j'ai réalisé n'est qu'une maquette implémentant ce qui avait déjà été discuté, à savoir les vues principales.

3.4.1 La page d'accueil

Pas entamé... car pas encore discuté collectivement.

3.4.2 La vue "interne"

Pas entamé... ça pourrait être un bon exercice pour unE débutantE en SPIP, tiens, d'ailleurs.

3.4.3 La vue chronologique

Le squelette `temps.html` liste tous les articles du site par ordre chronologique inverse.

3.4.4 La vue des lieux

Le squelette `lieux.html` liste les lieux (et oui), et donne les liens vers les pages spécifiques à chaque lieu (http ://SITE/lieu.php3 ?id_mot=#ID_MOT) car chaque lieu est un mot-clé).

3.4.5 La vue des thèmes

Le squelette `themes.html` liste les thèmes, et donne le lien vers la page spécifique à chaque thème (http ://SITE/theme.php3 ?id_mot=#ID_MOT) car chaque thème est un mot-clé).

3.4.6 La vue pour *un* lieu

Le squelette `lieu.html` liste les articles relatifs à ce lieu.

3.4.7 La vue pour *un* atelier/thème

Le squelette `theme.html` liste les articles relatifs à ce thème, en séparant les articles "intemporels" des autres (le fil de news).

3.4.8 La vue des trucs intemporels

Le squelette `documents.html` liste tous les articles du site ayant le mot-clé "intemporel" (fiches techniques, etc.)

3.4.9 La vue des trucs qui concernent la caravane en général

Le squelette `karavane.html` liste tous les articles relatifs non à un thème/atelier particulier, mais à la Karavane dans son ensemble (ie. les articles de la rubrique 3), en séparant les articles "intemporels" des autres (le fil de news).

3.4.10 La navigation entre les pages d'une même vue

On peut prévoir que certaines pages sont amenées à grandir ; la vue chronologique, par exemple, listera très vite un grand nombre d'articles. Il faut donc la scinder en plusieurs pages : la première liste les dix articles les plus récents, etc.

Pour réaliser cela, SPIP ne suffit plus et il faut faire appel à quelques lignes de code en PHP (le langage de programmation dans lequel est écrit SPIP). L'implémentation de cette fonction réside dans le fichier `mes_fonctions_cache.php3` ; cf. B.1 pour le pourquoi du comment de ce choix.

Pour l'instant, seul le squelette `temps.html` utilise cette fonctionnalité. Reste à l'ajouter aux squelettes `lieu.html` et `theme.html`.

NB : On ne pourra pas appliquer cette méthode aux squelettes `lieux.html` et `themes.html`, car elle nous rend tributaire d'un affichage par ordre chronologique inverse ; or, dans ces vues, nous souhaitons mettre en valeur des items (lieu ou thème) choisis *au hasard*. Et de toute façon, il y a peu de chances (?) que ces deux squelettes génèrent un jour trop de données pour que ça tienne sur une page...

3.4.11 Le menu de navigation

Afin de pouvoir bosser aisément sur le projet de site, j'ai créé `menu-dev.php3`, un menu incluant des liens vers les différentes vues et vers les tests de conformité des pages ; il est affiché sur certaines pages, p't'être pas sur toutes... à voir. C'est de toute façon temporaire.

3.5 Les feuilles de style

Ces fichiers portent l'extension `.css`. On les appelle, dans les squelettes, par leur adresse à partir de la racine du site ; il y en a quatre pour l'instant :

- `squelettes/k/habillage.css`
- `squelettes/k/impression.css`
- `squelettes/k/spip_style.css`
- `squelettes/k/typographie.css`

Je suis parti des feuilles de styles livrées avec SPIP, et je me suis borné à les nettoyer : j'ai viré tout ce qu'on n'utilisera pas (forums, brèves, plan, résumé...) afin de simplifier leur compréhension.

Maintenant, au tour de "on" ! Il faut que "on", grâce à de pertinentes feuilles de styles, fasse, de cette maquette mal dégrossie, un zouli site.

4 Des outils en complément de SPIP

4.1 Le wiki

On s'est dit que l'outil le plus adapté, pour les trucs censés être modifiés fréquemment et par de nombreuses personnes différentes, était le Wiki. Pour celleux qui ne savent pas ce que c'est, cf. http ://wiki.chutelibre.org/ :)

Pour la liste de matos, on a pensé à 3 pages Wikis :

- le matos qu'on a sur la Karavane ;
- le matos qu'on recherche ;
- le matos qu'on a à notre disposition à tel endroit".

Reste à ce que "on"[2] les mette en place...

De même, pour ce qui est de la préparation d'étapes de la caravane, un Wiki conviendra à merveille.

4.2 Listes de discussion/diffusion

À réfléchir collectivement.

[2] C'est qui, "on" ?

A Modifications du code source de SPIP

A.1 Pourquoi c'est mal, et pourquoi on le fait quand même (un peu)

L'idéal serait de pouvoir se passer de modifier le code de SPIP ; en effet, si l'on modifie le code de SPIP, une future mise à jour de SPIP écrasera nos modifications et nous aurions à les réintégrer ensuite en son sein. On essaiera donc de limiter nos modifications au strict nécessaire.

Toutefois, la version de SPIP utilisée ici (1.6) génère parfois du code non standard à la norme XHTML 1.0 Strict, et pour corriger ce fait, il faut modifier le code source de SPIP.

A.2 Liste des modifications effectuées

– `inc-calcul-squel.php3`, ligne 1478, dans le traitement de #FORMULAIRE_RECHERCHE : mis l'élément <input> dans un <p>

B Extensions apportées à SPIP

B.1 `mes_fonctions.php3` et `mes_fonctions_cache.php3`

Il est possible d'étendre SPIP par des fonctions écrites en PHP dans le fichier `mes_fonctions.php3`. Or les définitions de fonctions écrites dans `mes_fonctions.php3` ne sont prises en compte que lors du premier calcul d'une page par SPIP vers son cache, mais pas lors de l'interprétation PHP du cache de SPIP ; une solution est d'écrire les fonctions qu'on veut voir exécuter dans ce second temps dans un fichier `mes_fonctions_cache.php3`, inclus dans le squelette où l'on veut utiliser ces fonctions, grâce à la ligne :

```
<? include "mes_fonctions_cache.php3"?>
```

C'est ce qui a été fait pour le script permettant de naviguer entre les différentes pages listant les articles appartenant à une même vue.