

Nouveautés de yacs Martin

Version du 22 février 2010

Table des matières

1. Présentation générale de yacs Martin.....	5
2. Modifications importantes.....	6
2.1. Nouvelles possibilités de mise en page.....	6
2.2. Organisation flexible des composants de page.....	6
2.3. Alignement des règles de style.....	6
2.4. Visibilité des notifications.....	6
2.5. Simplification de la gestion des notifications.....	7
2.6. Liens canoniques.....	7
2.7. Informations publiques, communautaires, privées.....	8
2.8. Propriétaire de section, d'article.....	8
2.9. Centralisation de la gestion des droits.....	8
2.10. Nouvelles fonctions liées à la gestion des droits.....	9
2.11. Formulaire simplifié de saisie de contenu.....	9
2.12. Gestion des images.....	9
2.13. Ajout d'images.....	10
2.14. Notifications et mode brouillon.....	10
2.15. Extension de publication.....	10
2.16. Messages aux participants.....	10
2.17. Extension de l'interface overlay.....	10
2.18. Ouverture à des scripts fournis par des tiers.....	11
2.19. Sauvegarde de paramètres dans les conteneurs.....	11
2.20. Héritage des options.....	11
2.21. Extension des scripts inclus.....	11
2.22. Cache dans des fichiers.....	11
2.23. Taille maximum des fichiers.....	11
2.24. Support de PHP 5.....	11
3. Fichiers supprimés.....	13
4. Fonctions supprimées ou changées.....	14
4.1. are_allowed(), dans actions/actions.php.....	14
4.2. are_allowed(), dans articles/articles.php.....	14
4.3. check_default_editor(), dans shared/surfer.php.....	14

4.4. get_action_label(), dans shared/anchors.php.....	14
4.5. is_editable(), dans articles/article.php.....	14
4.6. is_interactive(), dans articles/article.php.....	14
4.7. stat_for_author(), dans articles/articles.php.....	14

A propos de l'auteur

Bernard Paques est le créateur initial du système yacs, l'architecte du logiciel, et son principal développeur. Il est aussi webmestre de plusieurs serveurs, tous construits avec yacs, sur plate-forme Linux. En véritable praticien des plate-forme web dynamiques, il a formalisé son expérience dans ce guide, ensemble de bonnes pratiques développées sur le terrain à travers de multiples projets.

Pour en savoir plus sur l'auteur : <http://www.linkedin.com/in/webnetworker>

Remerciements

Ce document n'aurait pas vu le jour sans la communauté yacsienne. Merci aux muses, aux relecteurs, et à tous ceux qui ont pris un peu de leur temps pour améliorer ce guide.

Pour rejoindre la communauté : <http://www.yacs.fr>

Copyright et licence d'utilisation

Copyright © 2010 Bernard Paques

Ce guide est distribué selon les termes de la licence Creative Commons by-nc-sa 3.0. En conséquence, vous êtes autorisé à copier, modifier, et distribuer publiquement ce document, aux conditions suivantes :

- Attribution – Vous devez reconnaître le travail de l'auteur, en préservant les marques de copyright placées dans le document.
- Non-commercial – Vous ne pouvez pas valoriser ce travail à travers une activité commerciale sans l'autorisation de l'auteur.
- Partage similaire – Si vous modifiez ou réutilisez ce travail, vous pouvez distribuer le document obtenu, mais seulement sous la même licence ou sous une licence similaire.

1. Présentation générale de yacs Martin

A définir

2. Modifications importantes

2.1. *Nouvelles possibilités de mise en page*

Comme tout gestionnaire de contenu qui se respecte, yacs fournit un grand nombre d'options pour affiner la mise en page des informations. Plusieurs layouts (c'est le nom des mises en page dans yacs) ont été introduits avec yacs Martin :

- `accordion` – Une seule page est visible à la fois. En cliquant sur un titre on fait apparaître son contenu, et le contenu de la page affiché précédemment est caché.
- `carrousel` – Avec un contrôleur en Flash qui fait tourner les images à l'écran
- `iui` – Cette mise en page spéciale est réservée aux accès depuis un mobile, genre iPhone
- `newspaper` – A la façon d'un journal, pour mettre en valeur les dernières nouvelles
- `rights` – mise en page spéciale, utilisée en interne par yacs pour afficher les rôles d'une personne (observateur, éditeur, propriétaire)
- `spray` – Une mise en page indissociable de l'overlay issue, pour construire un outil de suivi des problèmes
- `tagged` – Avec les étiquettes affichées prêt des titres, pour les informations classifiées
- `titles` – Les titres et les vignettes de page, et puis c'est tout !

Les mises en page qui existaient précédemment ont toutes été revues et optimisées, soit pour les performances, soit pour le rendu visuel, soit pour préciser l'usage des overlays, soit pour une combinaison de tout cela.

2.2. *Organisation flexible des composants de page*

Les composants de page sont à présent rangés en mémoire avant la construction des pages dynamiques. Un panneau de configuration permet de modifier l'ordre d'apparition des composants dans les différentes zones de la page. Il est même possible d'ajouter une page nommée dans la liste des composants de page, si besoin.

2.3. *Alignement des règles de style*

Les dénominations des règles de style sous-jacents à yacs, et des images, ont été harmonisées pour faciliter le travail de design. Lorsque cela est possible, les règles de style sont nommées d'après les mises en page qui les utilisent.

De plus, le fichier CSS principal de yacs, c'est-à-dire le fichier `skins/_reference/yacs.css`, contient à présent l'essentiel des styles générés par yacs. Cette consolidation est destinée autant à simplifier les évolutions futures qu'à limiter au strict minimum le contenu des thèmes développés par les web designers.

2.4. *Visibilité des notifications*

Les notifications ont été ajoutée au logiciel depuis longtemps. La nouveauté, c'est que yacs Martin indique au contributeur la liste des personnes notifiées après un ajout. Cette fonction donne beaucoup de visibilité aux auteurs, qui peuvent ainsi déterminer s'ils doivent inviter de nouveaux participants, etc.

2.5. Simplification de la gestion des notifications

Pour aider les usagers bombardés de notifications yacs Martin offre plusieurs mécanismes complémentaires :

Les messages 'Oublier cette page' et 'Oublier cette section' ont été remplacés par un simple 'Arrêter les notifications', c'est plus clair !

2.6. Liens canoniques

L'accès à l'information est flexible, puisque yacs affiche une page dès qu'il en connaît le type et le numéro. Cette flexibilité permet de s'accommoder à la fois des changements de titre, et des imprécisions dans les liens fournis par les internautes.

Pour yacs, tous les liens suivants conduisent à la même page :

- <http://www.yacs.fr/article-6783-en-2010-yacs-relance-la-matrice->.
- <http://www.yacs.fr/article-6783-en-2010>
- <http://www.yacs.fr/article-6783-ancien-titre-eventuel>
- <http://www.yacs.fr/article-6783>
- <http://www.yacs.fr/articles/view.php/6783>
- <http://www.yacs.fr/articles/view.php?id=6783>

Cette façon de procéder convient aux êtres humains, imparfaits et fantasques, que nous sommes. Mais pour les ordinateurs, et notamment pour les moteurs de recherche, cette souplesse induit beaucoup de confusion. En particulier, ils n'aiment pas recevoir le même contenu par des liens différents. Chez Google comme chez d'autres, la duplication des informations peut nuire à la visibilité d'un site.

La version yacs Martin ajoute une nouvelle bonne pratique à la souplesse précédente. Si le lien reçu n'est pas le lien canonique d'une page, yacs redirige le visiteur vers le bon lien. Un saut de puce qui restera indolore pour la plupart des internautes, mais qui fera toute la différence pour les moteurs de recherche.

Pour reprendre l'exemple précédent, le lien canonique est :

<http://www.yacs.fr/article-6783-en-2010-yacs-relance-la-matrice->.

Si l'un des liens suivants est fourni à yacs, l'internaute sera redirigé vers l'adresse canonique :

- <http://www.yacs.fr/article-6783-en-2010>
- <http://www.yacs.fr/article-6783-ancien-titre-eventuel>
- <http://www.yacs.fr/article-6783>
- <http://www.yacs.fr/articles/view.php/6783>
- <http://www.yacs.fr/articles/view.php?id=6783>

La redirection s'effectue par le biais d'un code HTTP 301, pour s'assurer que les moteurs de recherche mémorisent seulement les liens canoniques.

2.7. Informations publiques, communautaires, privées

Ces libellés ont été généralisés dans tout le code pour simplifier la gestion des droits d'accès. En résumé :

- Public – tout internaute, et les moteurs de recherche
- Communautaire – pour les personnes authentifiées (membres et souscripteurs)
- Privé – pour les personnes (membres et souscripteurs) assignées explicitement, et les associés

Les restrictions d'accès se règlent indépendamment sur chaque section, chaque article et chaque fichier. Par défaut, les informations sont en accès public. Toutefois, les restrictions sont aussi transmises au contenu, ce qui signifie que yacs combine le réglage d'accès d'un élément avec celui du conteneur, et retient le plus restrictif. Avec yacs Martin, le formulaire de modification précise ces différents cas dans les messages d'aide à la saisie, pour faciliter la compréhension des utilisateurs.

Grâce à l'héritage des restrictions d'accès, il n'y a pas besoin de gérer explicitement les droits pour chaque information. Généralement, il suffit de préparer quelques sections publiques, plus quelques sections communautaires ou privées, pour arriver à ses fins.

Un autre problème commun est l'élargissement des accès à une page placée dans une section privée. Si vous pouvez définir explicitement la liste des ayants droits, alors le mieux est d'utiliser la fonction 'Inviter des participants' directement. Sinon, si vous devez ouvrir l'accès à tous les membres, voir à tous les internautes, il faudra déplacer l'information dans une section communautaire ou publique.

2.8. Propriétaire de section, d'article

Un nouveau champ a été ajouté aux sections et aux articles pour mémoriser le propriétaire de l'information. Par défaut, le propriétaire d'une information est son créateur. Ce droit peut être délégué à tout moment à une autre personne par le propriétaire lui-même, ou par un associé.

Pour changer le propriétaire d'une section ou d'une page il faut passer par le formulaire de modification, puis sélectionner l'onglet Options. Yacs affiche le propriétaire en cours, et propose un bouton pour le modifier. Le changement est géré par le script `articles/own.php` pour les articles.

Le propriétaire d'une section dispose aussi des droits de propriété sur l'ensemble du contenu, y compris les sous-sections et les articles. Ainsi, il est très courant que plusieurs personnes différentes puissent agir en tant que propriétaire d'une page donnée, et cette approche collective doit être prise en compte dans la gestion des droits mise en place sur un serveur.

Par exemple, le responsable d'un département pourra créer une section pour ses équipes, puis créer une sous-section pour chacune des équipes qu'il gère. A leur tour, les responsables d'équipe seront propriétaires de leur espace d'équipe, et pourront s'ils le souhaitent déléguer la responsabilité de certaines pages à un de leurs équipiers. Grâce à la gestion des droits de propriété, yacs martin permet une transposition facile des responsabilités organisationnelles dans l'espace web.

Par rapport aux versions précédentes, certains droits donnés aux éditeurs sont à présent dévolus aux seuls propriétaires. Par exemple, les éditeurs ne peuvent plus inviter de nouveaux participants, seuls les propriétaires peuvent le faire.

2.9. Centralisation de la gestion des droits

La gestion des droits de création et de modification a été centralisée dans quelques fonctions pour la plupart des objets gérés par yacs.

Types d'objets	Gestion des droits
----------------	--------------------

Actions (dans actions/actions.php)	Actions::allow_creation()
Articles (dans articles/articles.php)	Articles::allow_creation() Articles::allow_deletion() Articles::allow_display() Articles::allow_message() Articles::allow_modification() Articles::allow_publication()

2.10. Nouvelles fonctions liées à la gestion des droits

Les requêtes effectuées dans la base de données prennent systématiquement en compte les droits des membres sur les sections et les articles qui leur sont attribués.

De nouvelles fonctions sont introduites pour évaluer les droits sur un article : `is_assigned()`, `is_editable()`, `is_owned()` et `is_watched()`. Voir le fichier `articles/articles.php` pour plus de détails.

Un document d'une trentaine de pages a été publiée récemment pour servir de référence sur la gestion des droits de yacs Martin. De plus, la plupart des requêtes émises pas yacs vers la base de données ont été vérifiées et affinées pour s'assurer de l'implémentation correcte des principes édictés.

2.11. Formulaire simplifié de saisie de contenu

Les versions précédentes de yacs permettaient d'intégrer tout type de script externe pour la visualisation de pages, par le biais des options de configuration. Par exemple, sur le mot-cle `view_as_tabs`, la visualisation ordinaire est remplacée par le chargement du script `articles/view_as_tabs.php`. Dans le cadre d'un projet particulier, il est possible d'utiliser le mot-clé `view_as_truc`, à condition de fournir à yacs un script `articles/view_as_truc.php`.

Ce mécanisme d'extension simple et puissant a été étendu avec yacs Martin aux formulaire de saisie de l'information. L'option à mettre en oeuvre doit être préfixée par `edit_as_`. Par exemple, si vous souhaitez simplifier la saisie d'informations, vous pourrez utiliser le mot-clé `edit_as_simple`, qui chargera le script `articles/edit_as_simple.php` en lieu et place du script usuel. Comme son nom l'indique, ce script, fourni avec yacs Martin, propose un formulaire simplifié, sans les onglets de tabulation ni les options compliquées. Vous pourrez vous inspirer de ce script pour créer toutes sortes de formulaires de saisie différents.

En combinant les options `views_as_` et `edit_as_`, vous contrôlez les scripts chargés par yacs pour la saisie et pour la restitution d'informations. Il devient extrêmement facile d'adapter le serveur aux besoins spécifiques d'une application client, d'autant plus que vous conservez la puissance des overlays et des layouts si nécessaire.

2.12. Gestion des images

Les formulaires de saisie des informations ont été modifiés pour faciliter la gestion des différents médias, y compris les images. Par défaut, yacs intègre un code de rendu d'image dans le champ de description à chaque fois qu'une image est ajoutée à une page. En modifiant la page, on peut accéder à l'onglet Médias pour gérer la liste des images, et pour obtenir les codes d'intégration disponibles.

2.13. Ajout d'images

La première image ajoutée à une page devient la vignette de cette page. Bien sûr, ceci peut être repris par la suite en cas de besoin.

2.14. Notifications et mode brouillon

yacs n'envoie plus de notifications pour les ajouts effectués sur les pages en mode brouillon. En revanche, la publication est à présent notifiée à tous les observateurs.

Il est donc possible de créer une nouvelle page, en mode brouillon, et de la peaufiner progressivement, y compris en ajoutant des images ou des pièces jointes, sans provoquer une seule notification. L'étape de publication sera aussi celle de la notification aux observateurs.

2.15. Extension de publication

Dans les versions précédentes de yacs l'extension de publication était appelée seulement lors de la publication explicite d'une page. Avec yacs Martin cette extension est aussi invoquée lors de la création d'une page en mode d'auto-publication (option `auto_publish`).

Ce changement est important, parce qu'il aide votre serveur à diffuser les nouvelles pages sur Internet. Précédemment, seules les publications explicites donnaient lieu à des requêtes 'ping'. A présent toute page publiée sera annoncée sur les serveurs que vous aurez paramétrés.

De plus, lors de la création d'une page publiée, yacs indique la liste des serveurs avertis de votre contribution. Vous pouvez donc vérifier, en temps réel, que la diffusion des informations publiées s'effectue correctement.

2.16. Messages aux participants

La version yacs Martin offre deux mécanismes distincts pour l'envoi de messages liés à une section ou à une page.

Le premier mécanisme sert à inviter de nouveaux participants à une section ou à une page. Un message est préparé par un propriétaire ou par un associé, à destination de certains membres de la communauté, ou à destination de personnes extérieures. Les membres sont aussi ajoutés à la liste des observateurs de la section ou de la page cible, ce qui signifie qu'ils seront automatiquement notifiés des ajouts effectués par la suite. Les invités peuvent utiliser la commande 'Arrêter les notifications' pour réguler les messages qu'ils reçoivent. Les invitations pour les articles sont gérées dans le script `articles/invite.php`.

Le deuxième mécanisme sert à envoyer un message électronique aux participants d'une section ou d'une page, en complément des notifications automatiques. Par exemple, le propriétaire d'une section pourra utiliser cette fonction pour rappeler l'échéance d'un travail collectif, ou un membre pour demander de l'aide aux autres membres. De plus, certains destinataires peuvent être désélectionnés en cas de besoin, ce qui permet un routage assez précis des messages, en fonction des besoins. Tout ceci est géré dans le script `articles/mail.php` pour les articles.

2.17. Extension de l'interface overlay

L'usage de la fonction membre `get_text()` a été étendue aux champs d'introduction et de description. Ceci permet par exemple de compléter un titre de page en fonction des données contenues dans l'overlay, comme implémenté dans l'overlay `issue` (dans le fichier `overlays/issue.php`). L'appel des fonctions de l'interface overlay a aussi été systématisé dans les différents scripts de yacs Martin.

L'overlay d'un article peut déclarer la fonction Javascript `validateOnSubmit()` qui, si elle existe, sera

appelée automatiquement par yacs pour valider les données saisies avant qu'elles ne soient transmises au serveur. Pratique pour améliorer l'interactivité des contrôles de saisie, non ?

2.18. Ouverture à des scripts fournis par des tiers.

Les fonctions `get_url()` s'accommodent de variantes extérieures à yacs. Auparavant, ces fonctions renvoyaient systématiquement à l'un des scripts de yacs, et l'intégration de scripts spécifiques était plus compliquée.

2.19. Sauvegarde de paramètres dans les conteneurs

Une nouvelle fonction membre `set_values()` a été ajoutée aux objets de type `Anchor`. Voir par exemple le fichier `articles/article.php` pour l'implémentation de cette fonction pour les articles.

2.20. Héritage des options

Une nouvelle fonction `Articles::has_option()` a été introduite dans `articles/articles.php` pour déterminer si une option est positionnée soit au niveau d'un article, soit de l'un de ses conteneurs. Par exemple :

```
if(Articles::has_option('links_by_title', $anchor, $item))  
    ...
```

2.21. Extension des scripts inclus

Les scripts suivants sont maintenant inclus systématiquement dans `shared/global.php` :

- `categories/categories.php`
- `files/files.php`

2.22. Cache dans des fichiers

Le cache dans la base de données est toujours disponible, quoique moins utilisé que dans les versions précédentes de yacs à cause du niveau de personnalisation atteint pour la génération des pages dynamiques.

De plus, yacs martin s'est doté d'un cache statique dans des fichiers, placés dans le répertoire `temporary`. Le but spécifique de ce cache est de gérer les informations quasi-stables, en particulier les fils de nouvelles RSS et ATOM, y compris ceux générés par yacs que ceux provenant de l'extérieur.

2.23. Taille maximum des fichiers

La taille maximum définie dans `PHP.INI` est à présent évaluée une seule fois dans `shared/global.php`, et placée dans la variable `$context['file_maximum_size']`. Le code redondant qui faisait ces évaluations en a été simplifié d'autant, bien sûr...

2.24. Support de PHP 5

La version yacs Martin est développée en environnement PHP 5, et tous les messages d'avertissement spécifiques à cette version du langage ont été pris en compte. Le logiciel fonctionne sans erreur

d'incompatibilité sur les systèmes PHP 5 courants.

Note : Le positionnement de l'option 'STRICT' dans PHP 5 continue de créer des message d'avertissement, qui sont liés à de nouveaux mots-clés à ajouter dans les modèles de programmation objet. Ces raffinements seront pris en compte lorsque la communauté aura formellement décidé d'abandonner complètement les environnements d'exécution PHP 4, ce qui devrait être le cas pour la prochaine version majeure.

3. Fichiers supprimés

Fichiers supprimés	Commentaires
agents/browsers.php agents/browsers_hook.php	Les calculs statistiques ne sont plus effectués par yacs. L'alternative recommandée est piwik, ou Google Analytics.
articles/fetch_for_palm.php	Qui utilise encore un Palm ? On pourra le remettre si de nombreuses voix le demandent, bien sûr.
articles/layout_articles_as_boxes_and_arrows.php articles/layout_articles_as_ids.php articles/layout_articles_as_rpc.php articles/layout_articles_as_slides.php articles/layout_articles_as_thread.php articles/layout-articles_as_thumbnails.php	

4. Fonctions supprimées ou changées

4.1. *are_allowed()*, dans *actions/actions.php*

Cette fonction est remplacée par `Actions::allow_creation()`, dans le même fichier. Les paramètres n'ont pas besoin d'être changés.

4.2. *are_allowed()*, dans *articles/articles.php*

Cette fonction est remplacée par `Aarticles::allow_creation()`, dans le même fichier. Attention, l'ordre des paramètres est inversé.

4.3. *check_default_editor()*, dans *shared/surfer.php*

L'appel de la fonction `Surfer::check_default_editor()` a changé. Dans les versions précédentes, on pouvait écrire :

```
$fields = Surfer::check_default_editor(array());
```

Maintenant, il faut écrire à la place :

```
$fields = array();  
Surfer::check_default_editor($fields);
```

4.4. *get_action_label()*, dans *shared/anchors.php*

Avant, il suffisait d'appeler `get_action_label()`. Avec yacs martin, le code de cette fonction a été déplacé dans le fichier `shared/anchors.php`, et il convient d'employer `Anchors::get_action_label()` à la place. Les paramètres d'appel restent inchangés.

4.5. *is_editable()*, dans *articles/article.php*

Cette fonction est remplacée par `is_assigned()` dans le même fichier. Attention, cette nouvelle fonction est plus complète que `is_editable()`. Voir aussi `Articles::is_editable()` dans `articles/articles.php`.

4.6. *is_interactive()*, dans *articles/article.php*

Cette fonction est déplacée dans `shared/anchor.php`.

4.7. *stat_for_author()*, dans *articles/articles.php*

Cette fonction est remplacée par `count_for_user()` dans le même fichier. Attention, la structure de l'information fournie est différente.