



scribite! 2.3

Documentation pour l'administrateur

Scribite! est un module qui permet d'intégrer facilement des scripts WYSIWYG comme Xinha, TinyMCE, FCKeditor, openWYSIWYG ou nicEdit dans des champs de type 'textarea' dans le but de rendre l'édition de texte plus simple et plus confortable.

scribite! en version 2.3 est disponible uniquement pour Zikula 1.1 ou supérieur.

Cette documentation s'adresse aux webmasters, les développeurs de modules peuvent lire la documentation qui leur est dédiée afin d'intégrer scribite! Dans leur propre module.

Auteur : Sven Schomacker (hilope)

Traducteur : Christophe Beaujean (Yokav)

URL: <http://trac.zikula.org/projects/scribite>

Publié sous licence GPL (pndocs/license.txt)

1. Features

scribite! ajoute la fonctionnalité wysiwyg aux champs textarea des modules suivants :

1. About
2. Admin_Messages
3. Book
4. Content
5. ContentExpress
6. cotype
7. crpCalendar
8. element
9. Eventliner
- 10.eventia
- 11.FAQ
- 12.htmlpages
- 13.Mailer
- 14.mediashare
- 15.News
- 16.PagEd
- 17.Pages
- 18.pagesetter
- 19.PhotoGallery
- 20.pncommerce
- 21.pnForum
- 22.pnhelp
- 23.pnMessages
- 24.pnWebLog
- 25.Profile
- 26.PostCalendar
- 27.Reviews
- 28.ShoppingCart
- 29.tFAQ
- 30.Timelt

2. scribite! Update

Si une version précédente de scribite! est installée, elle peut être mise à jour à partir de l'administration dans la partie 'Modules' après que tous les fichiers du dossier modules/scribite aient été mis à jour.

Depuis la version 1.2 les éditeurs files/javascripts ont été déplacés dans /javascript/scribite_editors et /modules/scribite/pnincludes.

Dans l'administration un message d'alerte apparait car les popus ne fonctionnent pas correctement depuis pn.8 qui utilise une restriction via .htaccess pour les dossiers des modules.

3. scribite! installation

1. Uploader tous les fichiers dans **modules/scribite**
2. Uploader les éditeurs que vous désirez utiliser dans **javascript/scribite_editors**.
3. Dans la partie 'Modules' de l'administration, régénérer la liste de modules et initialiser puis activer le nouveau module scribite!.
4. Dans l'administration de scribite! appliquer les paramètres désirés.
5. Certains modules ne fonctionnent pas directement avec scribite! Et ils doivent être légèrement modifiés :
 1. **PagEd, Pagesetter (v6.x), Postcalendar (v4.02 CVS), Book (v1.12), pnhelp (v1.0.0.0)**
Les fichiers requis sont dans **pndocs/ext_modules** et peuvent être copiés dans le dossier spécifique à chaque module (faites un backup des anciens fichiers!)
 2. **pncommerce**
Dans le fichier **pntemplates/pncommerce_admin_itemedit.html** rechercher après :

```
<textarea name='ItemDescription' rows='10' cols='40' wrap><!--  
[ $Item.Desc ]-->
```


et modifier comme ceci :

```
<textarea id='ItemDescription' name='ItemDescription' rows='10'  
cols='40' wrap><!--[ $Item.Desc ]-->
```
 3. **CoType 1.0**
Les fichiers requis sont dans **pndocs/ext_modules** et peuvent être copiés dans le dossier du module (faites un backup des anciens fichiers!)
6. Vider le dossier de cache et des fichiers compilés.
 - pnTemp/Xanthia_compiled
 - pnTemp/Xanthia_cache
 - pnTemp/pnRender_compiled
 - pnTemp/pnRender_cache
7. Terminé! ;o)

4. Ajouter de nouveaux modules

Si le module que vous désirez utiliser avec scribite! n'est pas dans la liste des modules compatibles, vous pouvez l'ajouter à scribite!.

Voici ce dont vous aurez besoin :

- nom du module (pas son alias mais uniquement le nom du dossier correspondant au module)
- les fonctions du module pour lesquels scribite! devrait être activé.
Si c'est un ancien module sans fonctions paramétrées ou que toutes les fonctions devront être utilisées, vous pouvez mettre „all“.
- Les ID des textarea pour lesquels l'éditeur devra être utilisé.
Si tous les ID devront être converti dans un format wysiwyg vous pouvez mettre „all“.

5. Editer des modules

Si des fonctions d'un module changent, vous pouvez utiliser le bouton éditer pour mettre à jour le module ainsi que toutes les fonctions et les ID des textarea.

Les mêmes choses sont nécessaires que pour „Ajouter de nouveaux modules“ („all“ paramètres). Pour PagEd vous pouvez utiliser „PagEd“ dans le champ ID dans le but de ne pas avoir tous les champs convertis par l'éditeur.

6. Supprimer des modules

Ici vous pouvez supprimer complètement un module de scribite!.

7. Templates

Par défaut le template **scribite_EDITOR_editorheader.htm** qui est utilisé convient à la plupart des utilisateurs. Si vous désirez ajouter ou modifier quelque chose, vous pouvez copier ce fichier dans le dossier /themes/YOURTHEME/templates/modules/scribite/ et y faire vos modifications.

Si un fichier **scribite_EDITOR_MODULENAME.htm** existe, il sera utilisé à la place du template par défaut, ainsi vous avez la possibilité d'utiliser différents éditeurs pour différents modules.

Les auteurs de modules peuvent créer une api appelée par scribite! directement (voir la documentation complémentaire) ou un template spécial qui sera appelé pour utiliser ses propres paramètres dans scribite!.

8. Xinha - configuration avancée



La configuration peut être modifiée dans **pntemplates/scribite_xinha_editorheader.htm**. (voir aussi 7. Templates)

Le fichier **pnconfig/xinha/editor.css** peut être modifié dans le but d'avoir la même apparence que le thème dans l'éditeur (uniquement IE pour le moment). Les modifications sur la barre de raccourcis peuvent être fait dans **scribite_xinha_toolbar_reduced.htm** ou **scribite_xinha_toolbar_mini.htm**.

Pour plus d'information visitez <http://xinha.webfactional.com/>.

Les plugins sont décrit ici <http://xinha.webfactional.com/wiki/Plugins>.

GetHtml est recommandé en tant que plugin.

(eng)

The plugin „ExtendedFileManager“ is at the moment not configurable in the administration of scribite!.

If needed the file

/javascript/scribite_editors/xinha/plugins/ExtendedFileManager/config.inc.php has to be edited for your needs.

Ex.:

```
...
$IMConfig['images_dir'] = $_SERVER['DOCUMENT_ROOT'].'/images/xinha';
...
$IMConfig['images_url'] = $_SERVER['SITE_ROOT'].'/images/xinha';
...
```

9. TinyMCE - configuration avancée



La configuration peut être modifiée dans **pntemplates/scribite_tinymce_editorheader.htm**. (voir aussi 7. Templates)

Pour plus d'informations visitez <http://tinymce.moxiecode.com/> .

Les plugins sont décrit ici

<http://wiki.moxiecode.com/index.php/TinyMCE:Plugins> .

Si vous désirez utiliser TinyMCPUK vous devez suivre cette procédure (merci à Erik Spaan):

- Créer un dossier pour les envois d'images (défaut: /images/tinymcpuk)
- Créer des sous-dossiers pour chaque type de fichier (Image, Media, File et Flash).
- CHMOD 777 sur tous les répertoires ci-dessus.

Si vous désirez utiliser un dossier différent de images/tinymcpuk pour les envois de fichiers éditez ce qui suit :

- /javascript/scribite_editors/tiny_mce/imagemanager/config.inc.php:
\$IMConfig['base_dir'] = \$_SERVER['DOCUMENT_ROOT'] . '/images/tinymcpuk/Image';
\$IMConfig['base_url'] = 'http://' . \$_SERVER['SERVER_NAME'] .
'/images/tinymcpuk/Image';
- /javascript/scribite_editors/tiny_mce/filemanager/connectors/php/config.php
\$fckphp_config['UserFilesPath'] = "/images/tinymcpuk" ;

10.FCKeditor - configuration avancée



La configuration peut être modifiée dans **pntemplates/scribite_fckeditor_editorheader.htm** .(voir aussi 7. Templates)

Pour plus d'information visitez <http://www.fckeditor.net/> .

11.OpenWYSIWYG - configuration avancée



La configuration peut être modifiée dans
pntemplates/scrubite_openwysiwyg_editorheader.htm .
(voir aussi 7. Templates)

Pour plus d'information visitez
<http://www.openwebware.com/products/openwysiwyg/docs/> .

L'Imagemanager(gestion d'images) qui est intégré (uniquement en mode avancé/complet) utilise /images dans le but d'avoir un accès général aux images de Zikula/Postnuke.

Si vous désirez pouvoir envoyer des fichiers :

- Créer un dossier images/openwysiwyg et faire un CHMOD 777
- Modifier (utile uniquement si vous désirez utiliser un autre dossier que images/ openwysiwyg)
/javascript/scrubite_editors/openwysiwyg/addons/imagelibrary/config.inc.php
\$imagebasedir = '../..../images/openwysiwyg';
\$imagebaseurl = 'images/openwysiwyg';

12.NicEdit - configuration avancée



La configuration peut être modifiée dans
pntemplates/scribite_nicedit_editorheader.htm .
(voir aussi 7. Templates)

Pour plus d'information visitez <http://www.nicedit.com/> .

13. Remerciements

- InteractiveTools.Com - they created htmlArea (<http://interactivetools.com/>), xinha is based on htmlarea
- all programmers who made xinha possible (<http://xinha.webfictional.com/>) especially gogo
- Moxiecode Systems AB - the creators of tiny_mce (<http://tinymce.moxiecode.com>)
- the programmers at <http://p4a.sourceforge.net/tinymcepuk> for their TinyMCPUK
- FCKeditor (<http://www.fckeditor.net>)
- openWYSIWYG (<http://www.openwebware.com>)
- nicEdit by Brian Kirchoff (<http://www.nicedit.com>)
- Sébastien Gruhier for Prototype Window Class (<http://prototype-window.xilinus.com>)
- the programmers of Snoopy Class
- Andreas Krapohl, who made the base module (<http://krapohl.info/>)
- Frank Schummertz, Steffen Voss, Carsten Volmer und Axel Guckelsberger for their help and inspiration (<http://pn-cms.net>)
- many thanks to all other pn users for their feature request, bug reports and hints making scribite! better
- translations by Daniel Neugebauer (por), Wilfried Santer (ita), tuerkyilmaz (tur), Arjen Tebbenhof (nld), Blaz Carli (slv), Dinmana (bos/hrv/srp), AnyKiller (spa), Yokav (fra), Jussuf (pol)
- everyone spending time in supporting and developing Zikula/Postnuke, xinha, TinyMCE, FCKeditor, openWYSIWYG and all the other fine scripts
- and special thanks to my wonderful girls Steffi and Charlotte ;)

14. Changelog

Voir changelog.txt