

Choix et installation de C3MS

Code: portal-tech

Originaux

url: <http://tecfa.unige.ch/guides/tie/html/portal-tech/portal-tech.html>

url: <http://tecfa.unige.ch/guides/tie/pdf/files/portal-tech.pdf>

Auteurs et version

- Daniel K. Schneider

- Version: 0.5 modifié le 7/11/04 par DKS

Prérequis

Module technique précédent: portal-def

Autres modules

Module concepts/théorie suppl.: tie-talk01

Objectifs

- Savoir choisir un portail en fonction de plusieurs critères
- Nécessite quelque visites de portails en même temps !!

url: <http://tecfaseed.unige.ch/>

url: <http://tecfa.unige.ch/guides/portals/pointers.html>

url: <http://opensourcecms.com/> (meilleur site de comparaison)

1. Table de matières détaillée

1. Table de matières détaillée	3
2. Les critères de base	4
3. Algorithmes de choix	5
4. Savoir “lire” un projet open source	7
5. Choix d’une technologie	8
6. Distribution et l’installation	10
6.1 Les archives zip, gzip, jar, rar, bz2	10
6.2 Gestion des fichiers sources	11
6.3 L’installation	12
7. Quelques commandes unix	13
7.1 Fichiers	13
7.2 Archives sous Unix	16
7.3 Permissions	17
7.4 Recherche, impression et visualisation	19

2. Les critères de base

- Il existe plusieurs critères pour choisir un portail C3MS gratuit et “open source”
- A notre connaissance, il n'existe pas de site qui fait une comparaison exhaustive des dizaines de systèmes (portalware) disponibles

Voici une suggestion pour une liste de critères:

1. Fonctionnalités (applications)
2. Solidité
3. Stabilité
4. Système d'utilisateur (permissions)
5. Système de “layout”
6. Interface utilisateur (ergonomie)
7. Interface administrateur (facilité)
8. Installation

3. Algorithmes de choix

- Il est souvent trop difficile de faire une évaluation "calculée"
- Pour savoir plus sur un type de portail il faut:
 - visiter des sites faits avec un système (portalware)
 - les tester
 - chercher l'opinion des autres Internautes

On suggère alors d'adopter une stratégie similaire à la suivante:

1. Définissez les critères de choix, par exemple:
 - fonctionnalités essentielles dont vous avez besoin
 - fonctionnalités supplémentaires (liste approximative, dans le portail ou "plugins")
 - réputation de "Solidité"
(cherchez par exemple "PostNuke vs PhPNuke" dans un engin de recherche)
 - mécanisme de plug-in "documenté"
(important si vous pensez éventuellement élargir ses fonctionnalités).....éliminez les portails qui ne les remplissent pas
2. Dressez une liste de portails qui semblent être vivants:
 - vous trouvez plusieurs installations qui fonctionnent très bien
 - vous constatez de l'activité sur le site des développeurs

- la dernière versions disponible ne date pas de plus que 6 mois
3. Si votre public est novice,
 - testez ensuite leur ergonomie (coté utilisateur !).... gardez ceux qui ont l'air corrects
 4. Insérez-vous dans une communauté locale
 - Si vous pouvez obtenir de l'aide localement pour un des portails sur votre "short list" choisissez celui-ci
 5. Sinon, installez ensuite tout ceux qui restent.
 - S'il y en a trop, ajoutez des critères, par exemple "extensibilité", qualité de la documentation technique, engin "thèmes", stabilité de l'API, taille de la communauté.
 - Eliminez ceux que vous n'arrivez pas à installer (parce qu'il manque l'infrastructure technique ou parce que la procédure est trop difficile).... choisissez avec votre "coeur"

4. Savoir "lire" un projet open source

Voici quelques conseils plus généraux

A. Lisez d'abord attentivement la "propagande" de chaque site

- Souvent les développeurs disent assez honnêtement où ils en sont ...

B. Le site "Source Forge" donne des statistiques de développement

url: <http://sourceforge.net/>

- La plupart des logiciels "open source" sont maintenant hébergés par ce site.
- Examinez l'activité de développement sur une plus longue période

C. Fouillez les forums, regardez la fréquence des "postings"

- Un site de support/développement sans activités est suspect.

D. Comptez les sites qui utilisent cette technologie

E. Visitez les sites de soutien (mods.xxx.org) etc.

F. Cherchez sur Google

- par exemple avec "xxx vs. yyy" ou encore "xxx yyy comparison"

5. Choix d'une technologie

- La plupart des portails "open source" sont écrits avec un langage de scripting et utilisent une base de données (surtout PHP/MySQL)
- La plupart des portails "cherchent encore leur architecture". Leur qualité est "Beta" et il faut vivre avec des "redesigns" et parfois des difficultés d'installation et/ou de bugs mineurs. On attend une stabilisation pour bientôt.
- La plupart sont "internationalisés". Toutefois, les versions françaises ont parfois du retard et la traduction manque pour certains modules.
- A Tecfa on utilise essentiellement PostNuke pour encadrer des pédagogies par activités ou encore des communautés. Il existe pleins d'autres alternatives de type C3MS.

url: <http://tecfa.unige.ch/guides/portals/pointers.html>

url: <http://opensourcecms.com/> (meilleur site de comparaison)

Quelques portails conseillés (différents types).

Portails testés à Tecfa	technologie	caractéristiques	Spécialités
phpWebSite	php/MySQL	C3MS simple	bon compromis entre complexité de PostNuke et la simplicité de certains weblogs

Portails testés à Tecfa	technologie	caractéristiques	Spécialités
PostNuke	php/MySQL	C3MS sophistiqué	très populaire, beaucoup de plugins et de thèmes. Lent & de qualité médiocre pour le moment (béta)
Drupal	php/MySQL	C3MS simple	Livre collaboratif, WebLog pour chaque utilisateur, "my"
GeekLog	php/MySQL	C3MS simple	Comme phpWebSite, mais avec un système d'utilisateurs, "My"
Jahia	Java / HSQL	Portail modulaire flexible (Genevois)	Déploiement de "portlets" par un mécanisme simple, gestion d'une population complexe, etc.
uPortal	Java / Oracle	Portail modulaire, très difficile à installer	Interface entre portail et services par "channels", gestion de multiples classes d'utilisateurs.
SPIP	php/mysql	CMS	Content Management System populaire
Site@Schools	php/MySQL	CMS / portail communautaire pour écoles primaires	Le bon compromis pour ceux qui veulent un peu de tout, mais pas trop de tout ...
Zope	python	CMS + C3MS	Portail gratuit sophistiqué
Wordpress	PhP/MySQL	Weblog personnel	Le blog en vogue (nov 2004)

6. Distribution et l'installation

6.1 Les archives zip, gzip, jar, rar, bz2

- Les archives sont des fichiers qui contiennent un ou plusieurs fichiers. Ils sont en règle générale comprimés. La plupart de logiciels "open source" sont distribués sous forme d'archives.
- Les archives *.zip sont très populaires dans le monde PC. Il s'agit d'archives comprimés.
- Les archives *.tar viennent du monde Unix. Ils ne sont pas comprimés.
- Les archives *.tar.gz ou *.tgz sont des archives *.tar comprimés avec en format *gz*. On obtient une meilleure compression qu'avec *.zip
- Les archives *.jar sont une variante des archives *.zip et sont utilisés surtout dans le monde Java.
- Les archives *.tar.bz sont des *.tar comprimés avec l'algorithme bz2 très puissant.

Outils Windows:

- WinZip (pour zip et tar/gz): utilitaire convivial avec un gui
- rar, jar, bzip, bunzip: utilitaires en ligne de commande.

6.2 Gestion des fichiers sources

A. Solution "locale"

- Téléchargez et décompressez l'archive sur votre machine en local
- Note: S'il existe des versions d'archives pour différents systèmes, choisissez celui du serveur ! (Si votre serveur tourne sous linux, il faut prendre l'archive pour Linux)
- Transférez les fichiers sur le serveur avec un outil FTP

B. Solution "serveur"

- Téléchargez l'archive et ensuite mettez-la directement sur le serveur
- Décompressez l'archive sur le serveur
 - Voir section 7.2 "Archives sous Unix" [16]

6.3 L'installation

- Emplacement de la documentation:
 - Lire la documentation sur le site de support si elle existe (cherchez un peu)
 - En règle générale, un fichier 'README.txt' (ou similaire) à la racine de l'archive contient les premières instructions, parfois il pointe vers un autre fichier (ex. INSTALL.text)
 - Si la documentation est plus importante, elle se trouve souvent dans un sous-répertoire "doc"
- Il est important de suivre ces instructions PAS par PAS !!!!
- Souvent l'installation proprement dite se fait par une interface Web. Toutefois, parfois il faut éditer un fichier de configuration. Dans les 2 cas, il faut fournir un travail précis !
- Si vous avez des problèmes:
 - vérifiez d'abord votre installation apache-mysql-php (ou autre)
 - allez lire les FAQ sur le site de support
 - cherchez sur le Web
 - postez (en dernier lieu!) une question dans un forum de support.

7. Quelques commandes unix

(pour pouvoir travailler sous telnet directement sur le serveur)

7.1 Fichiers

ls - lister le contenu d'un répertoire

Options:	-a	tous les fichiers
	-l	taille, date et permissions
	-R	affichage du contenu des sous-répertoires
	-t	trier selon la date
	-L	affiche le contenu d'un répertoire - lien symbolique
Exemples:	ls -la	affiche tous les fichiers du répertoire courant
	ls -lat *.text	affiche tous les fichiers "*.text" dans un répertoire, triés selon la date.

mkdir Créer un répertoire

Exemple :

```
(1) mkdir public_html
```

Crée un répertoire public_html à l'endroit où vous êtes

rmdir Détruire un répertoire

Exemple :

```
(1) rmdir test
```

détruit le sous-répertoire "public_html" à l'endroit où vous êtes
(rmdir marche uniquement si le répertoire est vide)

rm **Détruire un fichier**

Exemple:

(1) rm *.text

tue tous les fichiers *.text

(2) rm -R *

tue tous les fichiers y compris les sous-répertoires

cd **Changer de répertoire**

Exemple:

(1) cd

vous ramène dans votre "home". Equivalent à 'cd ~/'

(2) cd www/test

vous amène dans le répertoire www/test qui doit se trouver sous votre répertoire courant

(3) cd ~/www/test

vous ramène dans le répertoire www/test qui doit se trouver sous votre répertoire home

(4) cd /home/users/muller/www

vous met dans le répertoire /home/users/muller/www. Notez qu'il s'agit ici d'un chemin "absolu", c.a.d. attaché à la racine du système et qui marche partout

cp **Copier un fichier**

Exemple :

```
(1) cp bla.txt bla.save
```

mv **Bouger et renommer un fichier**

Exemple :

```
(1) mv bla.text bon-sens.text
```

renomme le fichier "bla.text" en fichier bon-sens.text"

```
(2) mv bla.text junk/
```

bouge le fichier "bla.text" vers le répertoire junk

find **Retrouver un fichier**

Exemple :

```
(1) find . -name bla.text -print
```

cherche le fichier bla.text a partir du répertoire courant

```
(2) find . -name '*.fm' -print
```

cherche tous les fichiers *.fm.

Attention: n'utilisez pas cette commande à la racine de notre système. Ca peut durer.

ln **Lien symbolique (alias)**

```
(1) ln -s /comm/tecfa/www/tecfa/tecfa-teaching/ tecfa-teaching
```

Crée un lien symbolique (alias) entre le répertoire '/comm/tecfa....' et 'tecfa-teaching' dans le répertoire ou vous etes.

INTERDICTION: Pour des raisons de sécurité, ne faites de liens de l'arbre WWW vers un autre endroit ! Pour optimiser le fonctionnement de nos engins de recherche etc. évitez des liens à l'intérieur de l'arbre WWW.

7.2 Archives sous Unix

unzip "défaire une archive *.zip"

Attention: il faut d'abord regarder ce qu'il y a dans une archive avant de "dézipper". En cas de doute, faites l'opération dans un répertoire vide

```
unzip -l pntlang_0.7.1.3_french.zip
```

Ensuite il est possible d'extraire les fichiers vers un autre répertoire:

```
unzip pntlang_0.7.1.3_french.zip -d french
```

Si vous avez fait une extraction dans un endroit temporaire, vous pouvez ensuite recopier les fichiers.

```
cp -r french/pntlang_0.7.1.3_french.zip/* ../rogue/html/
```

```
unzip --help
```

gtar "défaire une archive *.gz"

L'instruction suivante affiche le contenu:

```
gtar ztvf pn703_fix_3202.tar.gz
```

L'instruction suivante décompresse

```
gtar zxf pn703_fix_3202.tar.gz
```

L'instruction suivante décompresse dans un autre endroit

```
gtar zxf pn703_fix_3202.tar.gz -C /home/....
```

```
gtar --help
```


chmod Changer les permissions d'accès

Syntaxe: `chmod [ugoa] {+-} [rwx] <fichier(s)>`

<code><permissions> = u</code>	user's permissions
<code> g</code>	group's permissions
<code> o</code>	others' permissions
<code> a</code>	all permissions (user, group, and other)

Exemple:

(1) `chmod a+r *`

donne l'accès de lecture à tout le monde pour tous les fichiers dans le répertoire courant

(2) `chmod g+w hoho.txt`

donne l'accès d'écriture (détruire aussi!) au groupe pour le fichier hoho.txt

(3) `chmod o-w hihi.txt`

enlève l'accès d'écriture pour le "monde", (c.a.d. les gens qui ne font partie d'un même groupe)

chmod Syntaxe alternative:

PERMISSION COMMAND

`rw-rw-rw-rwxchmod 777 filename`

`rw-rw-r--r-xchmod 775 filename`

`rw-r--r--r-xchmod 755 filename`

`rw-rw-r--r--chmod 664 filename`

`rw-r--r--r--chmod 644 filename`

7.4 Recherche, impression et visualisation

lp **Imprimer un fichier**

Exemple :

(1) lp bla.txt

lpstat **Afficher les jobs d'impression**

cancel **Tuer une impression**

Exemple

(1) cancel ps-241

more **"Afficher" un fichier texte sur l'écran**

(1) more blabla.txt

less **"Afficher" un fichier texte sur l'écran**

grep **Chercher une chaîne de caractères dans un ensemble de fichiers**

(1) grep -ni "qui est connecté" *.php

Cherche "qui est connecté" dans tous les fichiers *.php du répertoire courant

(2) grep -nir "bla" *

Cherche "bla" dans les fichiers du répertoire et sous-répertoires.

(3) grep -nir "bla" * | more

Faire une sortie "paginée" (utiliser lorsque l'output est trop long)

man **Affiche une petite aide pour les commandes Unix**
(utilisez "Q" pour quitter et la barre d'espace pour afficher la page suivante)
exemple: man grep