

ESSAIS DE TRANSFERT D'UN SITE MOODLE A PARTIR D'UN SERVEUR 1 VERS UN SERVEUR 2, PAR « CLONAGE »

La particularité de cet essai est qu'il est entrepris sur deux environnements différents. L'objectif poursuivi est de vérifier le comportement d'une installation ancienne de Moodle sur un environnement plus récent, après copie « telle qu'elle » de l'installation initiale :

Serveur 1 :

OS : ubuntu 12.04 sur un VPS 1and1
Moodle 2.6+ (Build: 20131205)
php : 5.4.38-1+deb.sury.org~precise+2
Apache Version Apache/2.2.22 (Ubuntu)
MySql : 5.5.41-0ubuntu0.12.04.1

Serveur 2 :

OS : ubuntu 16.04 sur un VPS OVH
Moodle , à installer par clonage des dossiers Moodle du serveur 1
php : 7.0.15-0ubuntu0.16.04.4
Apache Version Apache/2.4.18 (Ubuntu)
MySql : 5.7.18-0ubuntu0.16.04.1

Nota 1 : si par la suite, vous vous posez des questions sur la gestion des rôles, des droits et permissions sur les fichiers, et que vous ne comprenez pas comment il faudrait éventuellement adapter ce retour d'expérience à votre situation particulière, alors ne vous embarquez pas dans cet exercice de clonage de Moodle, à moins que vous n'ayez du temps à consacrer à apprendre par vous-même et par l'échec !

Nota 2 : voyez tout de suite la conclusion, qui présente quelques réserves... Si ces réserves sont rédhibitoires, ne perdez pas de temps à lire les 15 points de ce retour d'expérience.

Nota 3 : si vous voulez faire un clone de votre installation pour avoir une copie destinée à faire des tests (par exemple des tests de migration) , faites-le donc sur un unique système : vous vous assurez ainsi un environnement identique... A cette fin, la procédure qui suit n'est pas valable, car sur le même hôte, il faudrait impérativement changer les noms de dossiers moodle et moodledata ainsi que le nom de la base de données ou du moins le préfixe des tables. C'est donc une autre procédure qu'il faudrait mettre en œuvre. Je l'ai déjà fait, et ça fonctionne très bien. Je l'ai fait aussi en changeant de serveur de base de données (de MySql à Posgresql). Tout cela est bien expliqué dans la documentation Moodle.

SAUVEGARDE DE LA PLATEFORME MOODLE

1. PLACER LE SITE EN MODE MAINTENANCE

- se connecter au site en tant qu'administrateur sur la page :
`url_de_ma_plateforme/admin/settings.php?section=maintenancemode`

2. SE CONNECTER AU SERVEUR D'ORIGINE 1AND1

- avec putty si je suis sous windows
- via un terminal de commande sous linux
- prendre l'identité d'un administrateur (Je sais que je ne vais pas faire de bêtise, et donc je prends le rôle d'administrateur système "root"...):
`patrick@serveur_1&1:~$ su root`
`password :`
`root@serveur_1&1 :/home/patrick#`

- créer un dossier temporaire où je mettrai mes archives à transférer sur mon nouveau serveur ; et me placer dans ce dossier :
`mkdir /archive_tempo ; cd /archive_tempo`

3. FAIRE UNE ARCHIVE DU DOSSIER D'INSTALLATION DE MA PLATEFORME

- En général le dossier s'appelle "moodle", et il est probablement sous le chemin /var/www/
- #`tar jcvf moodle.tar.bz2 /var/www/moodle/`
(oubien : `tar jcvf nom_du_dossier.tar.bz2 /chemin_du_dossier_web/nom_du_dossier/`)

4. FAIRE UNE ARCHIVE DU DOSSIER DE DONNÉES DE MA PLATEFORME

- En général le dossier s'appelle "moodledata", et il est probablement sous le chemin /var/ . Il n'est normalement pas dans le même répertoire que le dossier d'installation de la plateforme
- #`tar jcvf moodledata.tar.bz2 /var/moodledata/`
(oubien : `tar jcvf nom_du_dossier-data.tar.bz2 /chemin_du_dossier-data/nom_du_dossier-data/`)

5. FAIRE UNE ARCHIVE DE LA BASE DE DONNÉES (LA MIENNE TOURNE SOUS MySQL)

- `mysqldump -e -c -u root -p nom_de_la_base > nom_de_la_base_dump_170522.sql`
(j'ai mis un groupe dateur 170522, car j'ai fait ce « dump » le 22 mai 2017)

6. FIN DÉTAPE :

- Dans mon dossier /archive_tempo , je dois retrouver trois fichiers :

moodle.tar.bz2	<i>nom_du_dossier.tar.bz2</i>
moodledata.tar.bz2	<i>nom_du_dossier-data.tar.bz2</i>
nom_de_la_base_dump_170522.sql	<i>nom_de_la_base_dump.sql</i>

- Pour chacun des fichier, je calcul l'emprunte md5, et les note quelque part pour vérification ultérieure du bon achèvement des transferts sur mon nouveau serveur.
`md5sum nom_du_fichier`

TRANSFERT DES FICHIERS SUR LE SECOND SYSTÈME

Il existe diverses méthodes, notamment par ftp ou sftp, en ligne de commande, ou à l'aide d'un « client ftp » comme filezilla.

Chacun fera son choix, notamment en fonction des possibilités de connexion laissées par la configuration

des serveurs.

Moi, j'ai utilisé la ligne de commande en connection ssh, en mode « push » depuis mon ancien serveur (1&1) vers le nouveau (ovh), à cause de certaines restrictions d'accès que j'ai sur mon serveur 1&1 de production. On peut aussi utiliser un mode « pull » à partir du nouveau serveur, ce que je propose ci-dessous. (Avec Filezilla, on ne se pose pas la question de « push » ou « pull ».)

Quoi qu'il en soit, selon le type de protocole utilisé pour la connexion (ftp, sftp, ssh) il faut bien évaluer la question des droits d'accès aux serveurs (en root, en simple utilisateur) et des permissions sur les fichiers (en lecture et écriture).

Pour simplifier les choses, je supprime toutes les restrictions sur les dossiers et fichiers concernés lors du transfert, pour les repréciser ensuite le moment venu.

J'ai choisi de transféré un par un mes trois fichiers, dans un dossier préalablement créé dans mon nouveau serveur.

7. SUR LE SERVEUR D'ORIGINE, SUPPRIMER LES ÉVENTUELLES RESTRICTIONS D'ACCÈS AU DOSSIER D'ARCHIVES.

- `# chmod -R 777 /archive_tempo` (avec un compte administrateur, ou sous le compte root)
- Sur le nouveau serveur : création d'un dossier d'accueil de mes archives (via putty pour un serveur distant) :
- `# mkdir /home/accueil_170522` /création du dossier
- `# chmod 777 /home/accueil_170522` /suppression des restrictions d'accès à ce dossier

8. TRANSFERT DU DOSSIER MOODLE ARCHIVÉ, PAR APPEL À PARTIR DU SERVEUR DE DESTINATION (NOUVEAU SERVEUR) :

- `# cd /home/accueil_170522`
`# scp -P9999 root@vps.1&1:/home/archive_tempo/moodle.tar.bz2 moddle.tar.bz2` /remplacer 9999 par le n° de port configuré pour la connexion ssh ; si c'est le port par défaut, mettre 22, ou supprimer -P9999)
`#md5sum moddle.tar.bz2` / calcul de l'emprunte du fichier téléchargé, à comparer avec le résultat calculé précédemment sur le système d'origine ; recommencer le téléchargement si les empruntes sont différentes.

9. TRANSFERT DU DOSSIER MOODLEDATA ET DE L'ARCHIVE DE LA BASE DE DONNÉES :

- Même procédure que pour le dossier moodle.

10. FIN D'ÉTAPE

Sur le serveur de destination, dans le dossier temporaire /home/accueil_170522, nous retrouvons donc nos trois fichiers issus de la plate-forme moodle initiale :

- moddle.tar.bz2
 - moodledata.tar.bz2
 - nom_de_la_base_dump_170522.sql
-

INSTALLATION DE MOODLE SUR LE NOUVEAU SYSTEME.

Préalables :

1. Attribuer les droits de lecture et d'écriture sur les fichiers moodle, à l'utilisateur principal de l'installation, ainsi qu'au groupe d'utilisateur du serveur web. Sur mes deux installations, il s'agit de l'utilisateur patrick et du groupe www-data (c'est à titre préventif, l'utilité n'étant pas vraiment prouvée...) :
`# chown -R patrick:www-data /home/accueil_170522`
2. Vérifier éventuellement les différents points de configuration dans le fichier config.php de moodle (éditer le fichier qui se trouve dans le répertoire racine de l'installation de moodle, sur le serveur 1)
3. A partir de maintenant, on ne travaille plus que sur le serveur de destination (serveur 2), avec l'utilisateur « patrick », qui peut obtenir les droits d'administration avec la commande sudo (sous OS_Debian).

11. DÉCOMPRESSER LES DOSSIERS MOODLE ET MOODLEDATA

Il faut décompresser ces deux dossiers dans le serveur 2, chacun sous une arborescence de répertoires identique à celle du serveur 1. Donc pour moi, le dossier d'installation de moodle, sous le chemin /var/www ; et le dossier moodledata sous le chemin /var :

- Décompresser l'archive de Moodle :
`cd /var/www & sudo tar xfv /home/accueil_170522/moodle.tar.bz2`
- Décompresser l'archive de BasSabledata :
`cd /var & sudo tar xfv /home/accueil_170522/moodledata.tar.bz2`

12. ADAPTATION DU FICHIER DE CONFIGURATION

- Editer le fichier de configuration config.php (avec votre éditeur préféré, et avec un droit en écriture), par exemple :
`sudo nano /var/www/moodle/config.php`
 - Modifier le chemin d'accès à votre serveur : **(ne pas conserver l'url du serveur 1 !)**
`$CFG->wwwroot = 'http://url_du_serveur/moodle';` #si c'est en local : `http://localhost/moodle`;
 - Vérifier la conformité du chemin d'accès au dossier de données :
`$CFG->dataroot = '/var/moodledata';`
 - Eventuellement, annuler ou conserver l'envoi de courriels à partir de la plate-forme (« c'est vous qui voyez » comme dirait Laspalès !)
`$CFG->noemailver = true;` # ou bien `$CFG->noemailver = false;` pour empêcher l'envoi
 - En profiter pour noter les moyens d'accès à la base de données :
 - `$CFG->dbhost = 'localhost';`
 - `$CFG->dbname = 'nom_de_la_base';`
 - `$CFG->dbuser = 'user_de_la_base';`
 - `$CFG->dbpass = 'Pass_du_user_de_la_base';`
-

13. IMPORTATION DE LA BASE DE DONNÉES

- créer la base de données : (on suppose que l'administrateur MySQL s'appelle « root », et qu'il se connecte avec un mot de passe que vous connaissez...)
`mysql -u root -p -e "create database nom_de_la_base"`
- importer la base de données initiale :
`mysql -u root -p nom_de_la_base < "nom_de_la_base_dump_170522.sql"`

14. CRÉER L'UTILISATEUR DE LA BASE DE DONNÉES

- `mysql -u root -p`
- `enter password`
- `mysql>`
 - `GRANT ALL PRIVILEGES ON `nom_de_la_base`.* TO 'user_de_la_base'@'localhost' IDENTIFIED BY 'Pass_du_user_de_la_base' WITH GRANT OPTION;`
 - `quit`

15. FIN D'ÉTAPE

- Vérifier que tout fonctionne, en se connectant au site nouvellement installé :
A partir du serveur 2 (local) , dans un navigateur internet : <http://localhost/moodle>
A partir d'un ordinateur distant, via Internet : http://ip_de_mon_serveur/moodle
- Si cela ne fonctionne pas, c'est probablement que votre configuration des « hosts » n'est pas conforme dans votre installation d'Apache (je n'en ai pas encore parlé ; on aurait pu vérifier avant, mais là ce n'est plus du Moodle...).
 - Faites alors une comparaison des fichiers présents sur vos deux serveurs (serveur 1 et serveur 2), dans les dossiers sites-enabled et sites-availabled
 - Faites les modifications nécessaires pour configurer le « host » de votre moodle
- Penser à supprimer les dossiers temporaires si vous ne voulez pas conserver les archives compressées des dossiers moodle et moodledata ; ceci sur les deux serveurs.
- Vérifier les « propriétaire » et « groupe » attachés à vos dossiers, ainsi que les droits en écriture lecture sur les dossiers de Moodle. Faites les modifications nécessaires.
- Ne pas oublier de configurer le programme Cron

CONCLUSION PROVISOIRE

- J'ai fait un clonage d'une installation de Moodle 2.6 sur un environnement php_5.4 / MySql_5.5 vers un environnement php7.0 / MySql_5.7 sur deux systèmes d'exploitation différents.
 - Au final j'ai un Moodle qui tourne bien, mais qui perd certaines fonctions.
-

- Sans doute faut-il désactiver le mode maintenance à la première connexion, en tant qu'administrateur. Je n'ai pas eu à le faire, car dans les premières étapes, j'avais « archivé » mon site sans l'avoir préalablement placé en mode maintenance.
- Mon Moodle 2.6 semble bien tourner, mais certains plugins présentent quelques défauts, ou ne fonctionnent pas du tout, ce qui se voit vite lors d'essais de thèmes différents :
 - Le thème « clean » ne présente pas de défauts évidents.
 - Le thème « essentiel » présente des défauts d'affichage de certaines images propres au thème
 - Plusieurs thèmes ne sont plus disponibles, par exemple, le thème décafé (l'ajout dans l'url de la fonction ?theme=décafé renvoie au thème par défaut).
 - Je suppose que d'autres plugins ne seront plus opérationnels.
- Pour l'anecdote, j'ai eu du mal à me connecter en tant qu'administrateur, car j'avais oublié mon mot de passe, et que mon fichier config.php ne permettait pas l'envoi d'email (donc pas d'envoi de nouveau mot de passe à partir de l'écran de connexion à Moodle!)
- Prochaine étape : faire un « upgrade » de cette installation vers 2.7 puis 3.1 et mettre à jour les plugins que je souhaite utiliser, s'ils existent toujours.
- Autre méthode à essayer : faire un clonage sur une machine virtuelle locale configurée spécialement pour ce transfert, avec les mêmes versions de php et de MySQL.

*
* * *

RÉSOLUTION DES DÉFAUTS DE CLONAGE

Comme certains plugins ne fonctionnent pas, je suppose que c'est en raison de la différence d'environnement, notamment de la version de php. Je vais donc installer la version 5.4 de php, telle la version utilisée sur le serveur initial :

Ajout d'un PPA proposant d'anciennes versions, et installation du package :

```
add-apt-repository ppa:ondrej/php
apt-get update
apt-get install php5.4 ==> paquet introuvable...
apt-get install php5.6 ==> ok
```

Activation php5.6

```
a2dismod php7.0
a2enmod php5.6
service apache2 restart
```

La navigation web vers le site donne :

```
Error: database driver problem detected
The site administrator should verify server configuration
```

PHP has not been properly configured with the MySQLi extension so that it can communicate with MySQL. Please check your php.ini file or recompile PHP. MySQLi extension is not available for PHP 4.

==> Il faut ajouter les extensions qui vont bien à php5.6 pour lier les bases SQL

```
apt-get install php5.6-mysqldb
```

```
service apache2 restart
```

==> a priori toujours les mêmes problèmes, **mais le thème decaf est maintenant fonctionnel** (!?)...

Pour la version 2.6 de Moodle, il est recommandé les modules suivants de php : gd, zlib, OPcache.

Un phpinfo.php , montre que gd n'est pas installé.

Par ailleurs, un phpinfo.php de l'ancien système, en comparaison avec la liste du nouveau système met en évidence l'absence des extensions php suivantes dans la nouvelle installation, modules qui ne sont pas nécessairement utiles pour Moodle, mais peut-être pour d'autres applications :

```
*
* curl
* dba
* dom
* gd
* imap
* intl
* ldap
* mbstring
* mcrypt
* memcache
* openssl
* pspell
* simple XML
* soap
* sqlite3
* wddx
* xml
* xmlreader
* xmlrpc
* xmlwriter
* zip
```

Ajout de l'extension php manquante parmi celles recommandées pour Moodle 2.6:

```
apt install php5.6-gd
```

```
service apache2 restart
```

A noter que l'activité « publier » ne fonctionne pas : elle requière l'activation d' xml-rpc

```
apt install php5.6-xmllrpc
```

```
service apache2 restart
```

==> activité publier fonctionne !

Les thèmes essential et eguru présentent toujours des problèmes d'affichage de certaines petites icônes. Je suppose qu'il y a un lien avec les « google fonts », pourtant le thème essential est configuré pour les « standard fonts ».

==> Je bascule en « google fonts » ==> CA MARCHE !!! Et ça fait aussi fonctionner le thème eguru !

==> Je rebasculer en « standard fonts » ==> CA MARCHE AUSSI !!! Mystère ==> Non c'est en fait un problème de cache de Moodle, qui se serait résolu simplement par la purge du cache via le script prévu à cet effet : [\[url_mon_moodle\]/admin/purgecaches.php](#)

En effet : un passage en « localhost » reproduit le problème, qui se résout alors par la purge du cache :

1. Modif du chemin d'accès à l'hôte dans le fichier config.php :
 - o \$CFG->wwwroot = 'http://localhost/moodle'; # 'http://url_du_serveur/moodle';
 - o ==> connexion au site via un client web local ==> problème réapparaît ==> purge du cache ==> problème disparaît.
2. Modif retour du fichier config.php :
 - o \$CFG->wwwroot = 'http://url_du_serveur/moodle'; # 'http://localhost/moodle';
 - o ==> connexion au site via un client web distant ==> problème réapparaît ==> purge du cache ==> problème disparaît.

==> Au final, il semble que mon site fonctionne correctement. Il y a peut-être encore des choses par-ci par-là qui demanderont l'ajout d'extensions, ou des configurations diverses, tel le réglage du cron. Mais j'en reste là.

ET SI CA FONCTIONNAIT AUSSI EN PHP 7.0 ???

Rebasculer le serveur php en version 7.0, et vérifier les extensions pour ce serveur, ajouter celles qui manqueraient, notamment gd, zlib, OpCache

```
# a2dismod php5.6 & a2enmod php7.0 & service apache2 restart
```

+ vérification faite de la présence des extensions gd, zlib, opcache et xmllrpc

==> **TOUT FONCTIONNE BIEN ! MIRACLE !**

REMARQUE : au cours des opérations décrites ci-dessus, je n'ai pas utilisé le script « replace » dont l'utilisation est proconisée dans [la documentation Moodle, au sujet des « migrations de plateforme »](#). C'est évidemment un oubli, et il me semble impératif de mettre au clair la base de données, en remplaçant toute indication d' url de l'ancien site par l'indication de l' url du nouveau site . A défaut, tant que l'ancien site sera disponible, le nouveau site tournera bien, mais quand l'ancien site ne sera plus opérationnel, alors certaines ressources ne seront peut-être plus « appelées » par le nouveau site...

A FAIRE PAR LA SUITE :

1. **Mise à jour de cette installation clone** vers 2.7 puis 3.1 puis 3.3, à des fins d'essais...
 - o Se référer aux procédures habituelles de la documentation Moodle, pour chacune des montées de version.
 - o Faire les modifications de plugins si nécessaire (nouveau plugins en remplacement des plugins utiles de la version 2.6, mais ne fonctionnant plus en 3.3).
2. **Mise à jour de l'installation initiale**, pour conservation de l'environnement complet du système de production.

NOTA : Mais à quoi ça sert ?

Benh, je sais plus trop... Mais je ne suis pas seul à avoir eu cette idée !

Pour [Olivier](#) , si j'ai bien compris, il s'agissait de « sécuriser » une mise à jour d'un site moodle, en la réalisant sur un serveur local, avant de la basculer sur un serveur en production :

1. Clone SiteMoodle 2.6 Serveur_Prod vers
2. Mise à jour Moodle 2.6 vers 3.x sur Serveur_Intégration
3. Clone SiteMoodle 3.x du Serveur_Intégration vers Serveur_Prod.

Pour moi, le fond du sujet est simplement la migration d'un serveur 1 (VPS chez 1and1) vers un serveur 2 (VPS chez un autre fournisseur).

- L'intérêt est alors de disposer rapidement de tous les modules supplémentaires installés sur mon site Moodle, le clonage ayant « embarqué » ces modules. Bien sûr, certains modules ne fonctionneront pas immédiatement, car il nécessiteront l'installation de logiciels supplémentaires dans mon nouvel environnement. Il suffit alors d'installer ces logiciels dans mon système d'exploitation hôte.
- Un autre intérêt est « d'embarquer » les modifications éventuellement faites dans « le dur du code Moodle », modifications qui ne seraient pas reprises dans une installation fraîche de Moodle (je précise ici que mon installation à migrer s'appelle « Bac à Sable »),

NOUVELLE CONCLUSION

Pour « cloner » un site d'un serveur 1 à un serveur 2 : se référer [la documentation Moodle, au sujet des migrations de plateforme](#) », et s'aider éventuellement de ce « retour d'expérience ».

En particulier :

- Ne pas oublier de configurer les extensions php requises pour sa version de Moodle, ainsi que les extensions nécessaires au bon fonctionnement des plugins ajoutés.
 - Ne pas négliger de remplacer dans toute la base de données les indications d' url , soit en utilisant le script « replace », soit en ligne de commande avec la fonction « sed », soit en utilisant une fonction de « rechercher / remplacer » dans un éditeur de texte. Les deux premières méthodes sont données dans la documentation de Moodle.
 - Faire les purges des caches de Moodle.
-