



■ Configuration d'un serveur pour l'hébergement d'un site eZpublish V4

Ce document vous présente les différents points à respecter pour monter un serveur permettant l'hébergement de votre site réalisé par Inovagora, et installer le site sur ce serveur.

Si besoin, vous pourrez contacter nos administrateurs systèmes à l'adresse suivante : tech@inovagora.net

➔ Sommaire

➔ INTRODUCTION	3
➔ INSTALLATION DES LOGICIELS EXTERNES.....	4
■ Linux.....	4
■ Apache.....	4
■ PHP.....	4
■ MySQL.....	5
■ Accélérateur PHP.....	5
■ Envoi de mail.....	5
■ Manipulation des images.....	5
■ Extraction du contenu des fichiers en téléchargement.....	6
➔ CONFIGURATION GLOBALE.....	7
■ Apache.....	7
■ PHP.....	7
■ eAcclerator.....	7
■ Installation des modules PEAR.....	8
■ Compléments d'information.....	8
➔ INSTALLATION ET CONFIGURATION DU SITE.....	9
■ Livrables.....	9
■ Base de données.....	9
■ Désarchivage et paramétrage du site.....	9
■ Paramétrage Apache.....	11
■ Installation solr.....	11
■ Scripts d'initialisation.....	12
■ Crontab.....	12
➔ SAUVEGARDES.....	14
■ MySQL.....	14
■ Fichiers.....	14

Introduction

Il est possible que vous choisissiez d'héberger vous même votre site eZpublish finalisé. Ce document a pour vocation de vous informer sur les aspects techniques à respecter pour assurer un hébergement stable et de qualité.

Le temps nécessaire à la mise en service d'un serveur d'hébergement, dans le cas où aucune manipulation spécifique n'est nécessaire (recompilation du noyau Linux pour le support de certains matériels, configuration d'éléments externes au dit serveur), est approximativement d'une journée.

Il est recommandé de prévoir une journée supplémentaire pour l'installation du site et les tests de fonctionnement.

Le serveur doit respecter les préconisations faites dans le document ad hoc. Celui-ci préconise la distribution Debian, mais eZpublish peut fonctionner sur toutes sortes de serveurs Linux. Inovagora ne l'a toutefois testé que sur Debian et CentOS.

➔ Installation des logiciels externes

Inovagora ne garantit eZpublish que sous Linux. Bien qu'eZpublish fonctionne sur windows, cela nécessite des paramétrages particuliers qu'Inovagora ne maîtrise pas. D'autre part, les développements complémentaires faits par Inovagora ne sont pas garantis sur Windows.

Inovagora préconise l'utilisation de la distribution Debian, sur laquelle la société a acquis une certaine expertise. Nous pourrions vous aider plus efficacement et plus complètement lors d'une installation sur cette distribution.

Mais la solution proposée fonctionne parfaitement sur d'autres distributions, et Inovagora l'a déjà installée sur CentOS.

Toutefois, en terme de système d'exploitation, toutes les références seront données par rapport à une Debian. Cela s'applique particulièrement à la liste des packages à installer, qui sont donnés en terme de package Debian, et pas en terme de package rpm.

■ Linux

eZpublish fonctionne en architecture LAMP: Linux, Apache, MySQL, PHP.

La version du noyau Linux utilisée en standard est la version standard de la distribution Debian Lenny: 2.6.26. La solution fonctionne parfaitement sur une Debian Etch (2.6.18)

■ Apache

Apache doit tourner en version 2. La version utilisée couramment est la version 2.2.9 (toujours la version courante de la version Debian Lenny). Les modules Apache installés en standard sont:

```
apt-get install apache2-mpm-prefork
apt-get install apache2-utils
apt-get install apache2.2-common
```

■ PHP

PHP doit tourner en version 5.2.8 minimum. La version standard sur Debian Lenny est 5.2.6. Pour installer la dernière version de PHP sur une Debian, il faut modifier le fichier `/etc/apt/sources.list` et y ajouter les 2 lignes suivantes à la fin:

```
deb http://packages.dotdeb.org stable all
deb-src http://packages.dotdeb.org stable all
```

Il convient alors de réinstaller tous les modules PHP, y compris les module apache et mysql:

```
apt-get install libapache2-mod-php5
apt-get install php5
apt-get install php5-cli
apt-get install php5-common
apt-get install php5-curl
apt-get install php5-dev
apt-get install php5-gd
apt-get install php5-imagick
apt-get install php5-mcrypt
apt-get install php5-mysql
apt-get install php5-xmllrpc
```

```
apt-get install php5-xsl
apt-get install php-pear
```

ATTENTION: sur une Debian, l'extension PHP mbstring est installée en standard. Ce n'est pas le cas pour d'autres distributions. Cette extension doit être installée et activée.

■ MySQL

Le serveur MySQL doit être en version 4.1 minimum, mais nous recommandons l'utilisation d'une version 5. Il est nécessaire d'avoir un client mysql pour l'administration et les sauvegardes:

```
apt-get install libmysqlclient15off
apt-get install mysql-common
```

Une interface web à la base de donnée peut également être installée:

```
apt-get install phpmyadmin
```

■ Accélérateur PHP

eZpublish et Inovagora recommandent fortement l'installation d'un accélérateur PHP, ce qui améliore dans des proportions importantes les performances du site. Inovagora préconise eAccelerator. Ce logiciel n'existe pas sous forme de package Debian, il faut donc le compiler. Cela nécessite l'installation des packages php5-dev et automake (et possiblement d'autres).

```
apt-get install php5-dev
apt-get install automake
```

Il faut ensuite se procurer les sources du logiciel (la version est la dernière version au moment où ce document est écrit):

```
wget
```

<http://bart.eaccelerator.net/source/0.9.5.3/eaccelerator-0.9.5.3.tar.bz2>

et la compiler

```
bunzip2 eaccelerator-0.9.5.3.tar.bz2
tar xf eaccelerator-0.9.5.3.tar
cd eaccelerator-0.9.5.3
phpize
./configure
make
make install
```

La liste des dernières versions d'eAccelerator peut être obtenue à l'URL <http://eaccelerator.net/>

■ Envoi de mail

Le site web envoie des courriels à divers moments: remplissage d'un formulaire par un internaute, circuit de validation, etc...

Diverses solutions existent pour ce faire. La solution livrée utilise `sendmail`. Nos serveurs utilisent pour cela postfix.

■ Manipulation des images

Pour redimensionner automatiquement les images, eZpublish utilise ImageMagick. Il appelle la version shell d'ImageMagick (`convert`) et les fonctions PHP de l'interface `imagemagick`. Il convient donc que le PHP aie l'autorisation d'appeler des programmes sur la machine comme des commandes shell, et qu'ImageMagick soit

installé:

```
apt-get install php5-imagick  
apt-get install imagemagick
```

■ Extraction du contenu des fichiers en téléchargement

Les fichiers en téléchargement peuvent être indexés. Pour cela, il faut en extraire le contenu. La solution proposée par Inovagora a besoin de 2 logiciels à cette fin: `vwWare` et `xpdf`:

```
apt-get install wv  
apt-get install xpdf
```

Le contenu extrait sera indexé par le moteur d'indexation `eZFind`, qui est en fait une connexion à `solr` porté par Apache. Pour fonctionner, `solr` nécessite une machine virtuelle java.

Il faut donc télécharger une version de java pour votre machine (à télécharger depuis <http://developers.sun.com/downloads/>), et à l'installer. La version actuellement installée sur nos serveurs est la 1.6.14 (l'installateur est `jre-6u14-linux-i586.bin`). Une version un peu plus ancienne ou plus récente devrait convenir.

L'indexation demande aussi que PHP puisse appeler des commandes systèmes.

➔ Configuration globale

■ Apache

Pour que le site fonctionne, il est impératif que les rewrite-rules soit activées.

```
cd /etc/apache2/mods-enabled/  
ln -s ../mods-available/*rewrite* .
```

Apache doit être configuré pour accepter les virtual hosts, écouter sur le port 80, et éventuellement 443.

■ PHP

La configuration du site se fait dans `/etc/php5/apache2/php.ini`. Les valeurs suivantes doivent être modifiées.

Le fuseau horaire doit être positionné à CET:

```
date.timezone = CET
```

La place mémoire maximum utilisée par un script doit être augmentée, ainsi que le temps d'exécution:

```
memory_limit = 128M  
max_execution_time = 60
```

Des fichiers parfois lourds sont mis en ligne, il faut autoriser leur prise en compte par le site:

```
post_max_size = 50M  
upload_max_filesize = 50M
```

Certaines fonctionnalités du site ne sont opérationnelles que par le passage de scripts activés par la commande cron. Il faut donc aussi modifier le paramétrage du site pour le client PHP, dans le fichier `/etc/php5/cli/php.ini`. Le paramétrage précédent peut s'appliquer, à l'exception du `memory_limit`:

```
memory_limit = 256M
```

■ eAccelerator

eAccelerator utilise la mémoire partagée. Il faut déterminer quelle espace va lui être alloué dans cette mémoire. Cela dépend évidemment de la machine utilisée. Nous recommandons de lui allouer au moins 64M, si cela est possible.

On peut paramétrer la taille de la mémoire partagée en éditant le fichier `/etc/sysctl.conf` et en ajoutant la ligne:

```
kernel.shmmax = 100663296
```

La taille spécifiée ci-dessus correspond à 96M, dont nous allouerons 64 à eAccelerator.

Pour que la valeur soit prise en compte sans redémarrer la machine, on modifiera dynamiquement `/proc/sys/kernel/shmmax`:

```
echo 100663296 > /proc/sys/kernel/shmmax
```

Il faut également créé une directory dans laquelle eAccelerator stockera les fichiers PHP compilés. Nous suggérons `/tmp/eaccelerator`, mais ce peut être n'importe où.

On peut alors créer le fichier `/etc/php5/conf.d/eaccelerator.ini`, qui contiendra les valeurs suivantes:

```
extension="eaccelerator.so"
eaccelerator.shm_size="64"
eaccelerator.cache_dir="/tmp/eaccelerator"
eaccelerator.enable="1"
eaccelerator.optimizer="1"
eaccelerator.check_mtime="1"
eaccelerator.debug="0"
eaccelerator.filter=""
eaccelerator.shm_max="0"
eaccelerator.shm_ttl="0"
eaccelerator.shm_prune_period="0"
eaccelerator.shm_only="0"
eaccelerator.compress="1"
eaccelerator.compress_level="9"
```

ATTENTION: `shm_size` ne doit pas être supérieur à la taille de la mémoire partagée du système.

Après redémarrage d'apache, on doit avoir des directories apparaître dans `/tmp/eaccelerator`.

■ Installation des modules PEAR

Les modules PEAR sont utilisés par certaines fonctionnalités du site. Une partie d'eZpublish est elle-même installée sous forme de modules PEAR.

Les dernières versions de PHP (> 5.2.9) sont livrées avec un bug dans la configuration PEAR. Pour le corriger, on fera (cf <http://pear.php.net/>, paragraphe 'Fixing "unsupported protocol"'):

```
cd `pear config-get php_dir`
mv .channels .channels-broken
pear update-channels
pear update
```

La plupart des modules vont être installés à l'aide de la commande `pear`

```
pear install Cache_Lite
pear install Image_Color
pear install Image_Text-0.6.0beta
pear install Image_Tools-1.0.0RC1
pear install Numbers_Words-0.16.1
pear install OLE-1.0.0RC1
pear install Spreadsheet_Excel_Writer-0.9.1
pear install Text_Figlet
pear install Text_Password
pear install Text_CAPTCHA-0.4.0
pear channel-discover components.ez.no
pear install -a ezc/eZComponents
```

Un module supplémentaire ne fait pas partie du système de distribution PEAR, mais doit être installé comme un module PEAR: `Audio/CAPTCHA.php`. Ce fichier est livré avec le site, et doit être décompressé dans la directory contenant les modules PEAR.

■ Compléments d'information

La configuration type de nos serveurs peut être consultées à l'URL <http://info.bizlats.inovawork.net>

Les versions peuvent toutefois être plus anciennes que celles citées ici.

➔ Installation et configuration du site

■ Livrables

La livraison du site contient les éléments suivants:

- Un fichier sql permettant de remplir la base données
- Une archive contenant l'ensemble des fichiers du site
- Un fichier contenant le VirtualHost apache
- Une archive contenant le module PEAR `Audio/CAPTCHA.php` à décompresser dans l'architecture PEAR (cf ci-dessus)

Selon les cas, la livraison peut être faite en une grosse archive compressée, ou en 4 fichiers séparés eux-même compressés.

■ Base de données

Pour fonctionner, le site a besoin d'une base de données, d'un utilisateur et d'un mot de passe. Le serveur MySQL peut indifféremment être sur la même machine que le site, ou sur une autre.

Pour créer la base de données, vous pouvez utiliser les commandes suivantes:

```
echo 'CREATE DATABASE ` ${DBNAME} ` DEFAULT CHARACTER SET utf8 COLLATE utf8_bin;'\
> create_db.sql
echo "GRANT USAGE ON *.* TO ` ${DBUSER} `@`%` IDENTIFIED BY ` ${DBUSERPW} `
WITH MAX_QUERIES_PER_HOUR 0 MAX_CONNECTIONS_PER_HOUR 0
MAX_UPDATES_PER_HOUR 0 ;"\
>> create_db.sql
echo "GRANT ALL PRIVILEGES ON ` ${DBNAME} ` . * TO ` ${DBUSER} `@`%`; " >> create_db.sql
mysql -u root -p${ROOTPW} -h ${DBHOST} < create_db.sql
rm create_db.sql
```

où `${DBNAME}` est le nom de la base, `${DBUSER}` le nom de l'utilisateur qui sera utilisé par le site, `${DBUSERPW}` sont mot de passe. Il faut bien entendu avoir un utilisateur avec les privilèges permettant de créer une base de données, un utilisateur, et d'accorder des droits aux utilisateurs.

Une fois la base de données créée, vous pouvez jouer le fichier mysql (ici appelé `db.sql`) livré pour créer la base de données:

```
mysql -u ${DBUSER} -p${DBUSERPW} -h ${DBHOST} ${DBNAME} < db.sql
```

■ Désarchivage et paramétrage du site

Il faut aller dans la directory d'installation du site, et y désarchiver le site:

```
tar xf site.tar
```

L'architecture extraite contient a minima les fichiers et sous-directories suivants:

- `access.php`
- `autoload (directory)`
- `autoload.php`
- `bin (directory)`
- `cronjobs (directory)`
- `design (directory)`
- `doc (directory)`
- `extension (directory)`
- `ezcron.sh`
- `ezpdfototext.sh`
- `ezpm.php`

- ezpublish.cron
- index_image_mysql.php
- index_image.php
- index.php
- index_treemenu.php
- kernel (directory)
- kickstart.ini-dist
- lib (directory)
- LICENSE
- packages (directory)
- pre_check.php
- README.txt
- runcronjobs.php
- schemas (directory)
- settings (directory)
- share (directory)
- soap.php
- support (directory)
- update (directory)
- var (directory)
- webdav.php

L'utilisateur apache doit avoir les droits en lecture sur l'ensemble des contenus, et les droits en lecture/écriture sur les directories settings/override, settings/siteaccess/*, var et leurs contenus respectifs uniquement.

On peut laisser les droits en exécution sur l'ensemble des fichiers sans risque.

Le fichier settings/override/site.ini.append.php contient une série de correspondances permettant de relier les URLs de vos différents sites avec le concept eZpublish de siteaccess. Il y a au moins 2 sites: un pour l'administration (généralement à l'URL admin.<domaine>) et un pour les internautes (généralement www.<domaine>).

Note importante: pour améliorer l'indexation, il convient que lorsqu'un site peut être atteint par plusieurs URL, toutes ces URLs soient redirigées au niveau d'apache vers une seule, appelée URL principale. Dans le fichier ci-dessus, il ne doit y avoir qu'une seule URL correspondant à un siteaccess.

Ces correspondances sont de la forme:

```
HostMatchMapItems[]=www.<domaine>;XXX_user
HostMatchMapItems[]=admin.<domaine>;XXX_admin
```

où **XXX_user** et **XXX_admin** sont les siteaccess

Pour chacun des siteaccess qu'on va trouver dans dans cette liste, on éditera le fichier de paramétrage correspondant. Par exemple, pour **XXX_user**, on éditera le fichier settings/siteaccess/**XXX_user**/site.ini.append.php et on modifiera les lignes définissant les paramètres de la base de données:

```
[DatabaseSettings]
DatabaseImplementation=ezmysql
Server=XXXX
Port=
User=XXXX
Password=XXXX
Database=XXXX
Charset=
Socket=disabled
```

```
SQLOutput=disabled
```

Les valeurs **xxxx** sont à modifier en fonction de votre configuration. Si le serveur de base de données est sur la même machine que le site (Server=localhost), vous pouvez remplacer la valeur du paramètre Socket par le fichier à utiliser.

Nous vous conseillons également de vérifier dans ces fichiers que les valeurs de [MailSettings]
AdminEmail=mail@domain
[EmailSender=mail@domain](mailto:mail@domain)
ont été correctement positionnées.

■ Paramétrage Apache

Inovagora fournit un virtual host apache pour le site. Ce fichier doit être mis dans la directory sites-available de la directory de configuration d'apache (a priori, /etc/apache2).

Si on est dans la directory ou l'extraction de l'archive livrée a été faite:

```
mv vhost /etc/apache2/sites-available/<domain>
```

puis on va faire un lien symbolique dans la directory sites-enabled de la directory de configuration d'apache:

```
cd /etc/apache2/sites-enabled  
ln -s ../sites-available/<domain> .
```

On éditera ensuite le fichier /etc/apache2/sites-available/<domain> pour modifier les lignes:

```
<Directory xxx>  
DocumentRoot xxx  
ErrorLog /var/log/apache2/yyy  
CustomLog /var/log/apache2/zzz combined
```

pour les faire correspondre à votre architecture.

Vous pouvez alors tester si la configuration apache:

```
apache2ctl configtest
```

puis relancer apache:

```
apache2ctl graceful
```

Vous pouvez alors l'accès à votre site en allant à l'URL d'administration (en principe, admin.<domain>). Si la page apparaît sans erreur (une erreur courante quand il y a un problème de configuration est une erreur « kernel 50 » sur la page de login).

■ Installation solr

L'indexation des contenus utilise le projet apache solr. Ce projet est écrit en java et doit tourner en permanence sur la machine. Il est donc important que son démarrage soit intégré aux scripts de boot de la machine.

ezpublish fournit ces scripts pour les distribution Debian, CentOS et Red Hat Enterprise Linux. Il suffit de les adapter en fonction de votre architecture. Si vous utilisez une autre distribution, vous pouvez les adapter à vos spécificités.

Ces fichiers se trouvent (à partir de la directory d'installation) dans extension/ezfind/bin/scripts/debian pour une Distribution Debian, et extension/ezfind/bin/scripts/rhel pour une distribution Red Hat ou CentOS

Il faut copier le fichier solr correspondant à votre distribution dans `/etc/init.d`, et l'éditer pour positionner la valeur de `SOLR_HOME` à **`directory_d_installation`**/`extension/ezfind/java`, et éventuellement de `JAVA_HOME` à la directory où java est installé.

Il faut alors mettre le script dans les directories exécutées au redémarrage de la machine (`/etc/rcX.d`) en suivant les instructions définies au début du fichier.

On peut alors démarrer solr (attention aux droits d'exécutions sur `/etc/init.d/solr`):

```
/etc/init.d/solr start
```

■ Scripts d'initialisation

Pour le bon fonctionnement du site, il faut exécuter 4 scripts, après s'être mis dans la directory d'installation.

```
php bin/php/ezpgenerateautoloads.php -extension
php bin/php/flatten.php -s xxx_admin all
php extension/ezfind/bin/php/updatesearchindexsolr.php -s xxx_admin --php-exec=/usr/bin/php
```

où **`xxx`**_admin est le siteaccess de votre site d'administration

Pour chaque siteaccess correspondant à un site internaute, il faut exécuter:

```
php runcronjobs.php -s xxx_user sitemap
```

où **`xxx`**_user est le siteaccess du site.

■ Crontab

Les sites utilisent quelques scripts dont le lancement doit être planifié à intervalles réguliers.

Les commandes suivantes doivent donc être insérées dans la crontab d'un utilisateur ayant les mêmes droits que l'utilisateur apache.

```
bash directory_d_installation/ezcron.sh
```

Cette commande est à exécuter une fois par jour, de préférence la nuit après minuit. Elle fait du nettoyage, dépublie les contenus obsolètes, publie les contenus qui ont été spécifiés comme devant l'être, vérifie la validité des liens dans les contenus, etc... Son exécution peut être lourde.

```
bash directory_d_installation/ezcron.sh hourly
```

Cette commande est à exécuter toutes les heures. Notamment, elle intègre les flux RSS externe sur le site.

```
bash directory_d_installation/ezcron.sh sitemap
```

Cette commande est à exécuter une fois par jour, de préférence la nuit. Elle régénère le fichier sitemap.xml utilisé par les robots pour y intégrer les contenus.

```
bash directory_d_installation/ezcron.sh wf
```

Cette commande permet de faire avancer les circuits de validation et de publication. Sa fréquence dépend de vos besoins. Sur une machine mutualisée, nous conseillons une fréquence d'une fois toutes les demi-heures. Sur une machine dédiée, on peut avoir une fréquence plus grande.

```
bash directory_d_installation/ezcron.sh import_directory_csv
```

Cette commande permet d'importer les annuaires automatiquement à partir de fichiers CSV. Sa fréquence dépend de vos besoins. Son exécution peut être lourde si vous importez de gros annuaires

```
bash directory_d_installation/ezcron.sh send_newsletter
```

Cette commande n'est à utiliser que si votre site propose une lettre d'information. C'est elle qui envoie effectivement lettre d'information. Sa fréquence dépend de vos besoins. Son exécution peut être lourde si vous avez de grosses liste de diffusion.

D'autres commandes peuvent être nécessaire selon les cas particuliers.

Un fichier crontab type a la forme suivante:

```
15 1 * * * bash directory_d_installation/ezcron.sh
0 * * * * bash directory_d_installation/ezcron.sh hourly
15,45 8-18 * * * bash directory_d_installation/ezcron.sh wf
50 3 * * * bash directory_d_installation/ezcron.sh send_newsletter
30 * * * * bash directory_d_installation/ezcron.sh import_directory_csv
```

Les crons génèrent des logs dans la directory d'installation, dans var/log/cron.**aaaammjj**.log. Ces logs sont conservés 7 jours.

➔ Sauvegardes

Dans ce chapitre, nous contenterons de souligner quelques points importants. Les sauvegardes sont à mettre en place selon vos procédures propres.

■ MySQL

Les sauvegardes de la base de données ne doivent pas être faites en copiant les fichiers de la base de données, mais en sauvegardant un dump de celle-ci. On peut faire un dump en utilisant la commande `mysqldump`:

```
mysqldump -u user -ppassword -h host db_name > fichier_dump.sql
```

On peut alors sauvegarder *fichier_dump.sql*

■ Fichiers

La plupart des fichiers de l'installation sont invariants, sauf en cas d'ajout d'un nouveau site, d'un nouveau module, ou de fonctionnalité particulière (flipbook, par exemple).

Dans ce cadre, les seuls changements se font dans les directories settings et var. En cas de problème de place pour les sauvegardes, on peut se contenter de sauvegarder uniquement ces 2 directories journalièrement.

Nous recommandons toutefois de sauvegarder l'ensemble de la directory d'installation. `rsync` est un outil puissant et utile dans cette optique.