

Académie Lille

**Formation**  
**Drupal 8:**  
**Cas pratiques**

**Version : J1**  
**01/10/2018**



The word "SMILE" is written in a bold, white, sans-serif font. It is surrounded by various orange decorative elements: a rocket, a heart, a speech bubble, and several small stars and dots.

I.T IS OPEN

## Table des matières

<b>I PRÉREQUIS</b>	<b>3</b>
I.1 SYSTÈME D'EXPLOITATION	3
I.2 OUTILS DE GESTION DU CODE SOURCE	3
I.3 INITIALISATION DU CODE SOURCE DRUPAL	4
I.4 CONFIGURATION DE LA BASE DE DONNÉES	4
I.5 OUTILS EN LIGNE DE COMMANDES	5
I.6 INSTALLATION DU SITE DRUPAL	5
<b>II ENVIRONNEMENT DE DÉVELOPPEMENT</b>	<b>6</b>
II.1 ÉDITEUR DE CODE	6
II.1.a PHPStorm	6
II.1.b Visual Code	6
II.2 CODINGS STANDARDS	6
II.3 VERSIONNING	6
II.4 SETTINGS DRUPAL	6
II.4.a Configurer un ou plusieurs sites	6
II.4.b Désactiver le cache twig	6
II.4.c Désactiver le cache Drupal	6
II.5 DÉPANNAGE DE SITE AVEC DRUSH	6
<b>III ENVIRONNEMENT DE PRODUCTION</b>	<b>7</b>
III.1 CONFIGURATION DE VARNISH	7
III.2 CONFIGURATION DE MEMCACHE	8
III.3 CONFIGURATION DE DRUPAL	8
<b>IV LES MODULES DRUPAL 8</b>	<b>9</b>
IV.1 MODULES COMMUNAUTAIRE	9
IV.1.a Installer un module	9
IV.1.b Activer un module	9
IV.2 ADMINISTRATION DE BASE (PRÉ-REQUIS)	9
IV.3 DEVEL	9
IV.3.a Dumper des variables	10
IV.3.b Exécuter du code	10
IV.4 DÉINSTALLATION	10
IV.5 PATCHER UN CODE COMMUNAUTAIRE	10

# I PRÉREQUIS

## I.1 SYSTÈME D'EXPLOITATION

Le support de PHP est optimisé pour les systèmes d'exploitation Linux, nous conseillons d'utiliser **Debian (Jessie, Stretch)**, ou un équivalent utilisant le gestionnaire de packages APT, comme **Ubuntu (16+)**.

### Dépendances LAMP (Debian / Ubuntu)

Les commandes suivantes sont utilisées dans une console Linux pour installer les dépendances d'un serveur LAMP (Linux Apache Mysql PHP) :

```
## Mise à jour des packages du système :  
apt update && apt upgrade
```

```
## Dépôt php 7 pour ubuntu 16 (natif à partir ubuntu 18 et debian stretch):  
apt-get install python-software-properties pkg-config software-properties-common  
  
add-apt-repository ppa:ondrej/php  
apt update
```

```
## Installation des packages du serveur LAMP  
apt install php7.2 apache2 libapache2-mod-php php-gd php-curl php-xml  
php-json php-mbstring php-mysql  
apt install percona-server-server
```

## I.2 OUTILS DE GESTION DU CODE SOURCE

**Composer** est utilisé pour gérer l'installation et la mise à jour du code source communautaire dont dépend l'écosystème Drupal 8.

```
## Installation globale de composer  
php -r "readfile('https://getcomposer.org/installer');" | php  
mv composer.phar /usr/local/bin/composer
```

**Git** est utilisé par Composer pour télécharger les dépendances, il est également utilisé lors de nos développements pour gérer l'historique des versions et partager le code source entre les développeurs.

```
## Prérequis à la gestion du code source  
apt install git
```

## I.3 INITIALISATION DU CODE SOURCE DRUPAL

Le code source du site Drupal est contenu dans un répertoire connu par le serveur web Apache, appelé Virtual Host (Vhost). Le dossier /var/www contient l'ensemble des Vhosts Apache.

```
## Téléchargement des sources Drupal
cd /var/www
composer create-project drupal-composer/drupal-project:8.x drupal8
--stability dev --no-interaction
## Réglage des permissions sur le dossier du site web :
chown -R www-data:www-data /var/www/drupal8/
```

### Configuration du Vhost Apache

Pour des raisons de sécurité, nous désactivons le support des fichiers .htaccess (exemples de failles : [https://medium.com/@insecurity\\_92477/utilizing-htaccess-for-exploitation-purposes-part-1-5733dd7fc8eb](https://medium.com/@insecurity_92477/utilizing-htaccess-for-exploitation-purposes-part-1-5733dd7fc8eb))

La configuration proposé du Vhost Drupal reprend donc le contenu du fichier .htaccess fournie par le core de Drupal 8.

```
## Copier le fichier /etc/apache2/sites-available/drupal8.conf depuis
l'archive conf.zip fournie vers le chemin correspondant sur le serveur.
```

Afin de gagner du temps de configuration des postes développement, nous utilisons le Fast CGI standard :

```
a2ensite drupal8 && a2dissite 000-default
a2dismod mpm_event
a2enmod rewrite mpm_prefork php7.2 # use php fpm on production server
for performance reason: https://www.howtoforge.com/tutorial/apache-with-php-fpm-on-ubuntu-16-04/
## Vérifie que le Vhost est bien configuré puis affiche la configuration
apache2ctl -t && apache2ctl -S
service apache2 restart
```

Depuis PHP 5.3.3, un nouveau interpréteur de PHP haute performance à fait son apparition : PHP FPM (PHP Fast Process Manager). La configuration de ce dernier nécessite des manipulations supplémentaires. Il est conseillé de l'utiliser sur les environnements de production, mais pas nécessaire pour développer. Exemple de config PHP FPM: <https://www.howtoforge.com/tutorial/apache-with-php-fpm-on-ubuntu-16-04/>

## I.4 CONFIGURATION DE LA BASE DE DONNÉES

Avec l'utilisateur {USER} et le mot de passe {PASSWORD} renseignés à l'étape d'installation de Percona (Système d'exploitation), nous ajoutons un utilisateur disposant des droits d'actions sur la base **drupal8**

```
mysql -u{USER} -p{PASSWORD}
CREATE DATABASE drupal8 DEFAULT CHARACTER SET utf8 DEFAULT COLLATE
utf8_general_ci;
GRANT SELECT, INSERT, UPDATE, DELETE, CREATE, DROP, INDEX,
ALTER, LOCK TABLES, CREATE TEMPORARY TABLES
ON drupal8.* TO drupal@localhost IDENTIFIED BY 'drupal';
```

## I.5 OUTILS EN LIGNE DE COMMANDES

Les outils **Drush** et **Drupal console** permettent de gagner du temps pour effectuer de nombreuses tâches tel que l'installation d'un site, le vidage de cache, la réinitialisation de mot de passe, la création d'entités et la manipulation des configurations du site.

Les deux sont comparables, Drush précède historiquement Drupal Console, mais ce dernier s'appuie sur des commandes Symfony, il permet donc d'afficher plus facilement des informations concernant les composants Symfony, notamment les services et les routes.

```
## Installation globale de Drush et Drupal Console
cd /var/www/drupal8/
composer require drush/drush:9.x
composer require drupal/console
echo export PATH=$PATH:/var/www/drupal8/vendor/bin >> ~/.profile

/!\ La mise à jour du path nécessite de se reconnecter au terminal /!\
```

## I.6 INSTALLATION DU SITE DRUPAL

Nous installons le nouveau site Drupal avec un utilisateur disposant des droits administrateur (admin;admin)

```
cd /var/www/drupal8
drupal site:install --account-name=admin --account-pass=admin \
--db-type=mysql --db-host=localhost --db-name=drupal8 --db-user=drupal \
--db-pass=drupal --langcode=en standard
chgrp www-data -R sites/default/files
chmod g+w -R sites/default/files
```

Pour vérifier que tout fonctionne, drush affiche le status de connection à la base de données du site

```
drush status
Database           : Connected
Drupal bootstrap   : Successful
```

## II ENVIRONNEMENT DE DÉVELOPPEMENT

### II.1 EDITEUR DE CODE

#### II.1.a PHPStorm

**Guide de configuration :**

<https://confluence.jetbrains.com/display/PhpStorm/Drupal+Development+using+PhpStorm>

#### II.1.b Visual Code

**Guide de configuration :**

<https://www.drupal.org/docs/develop/development-tools/configuring-visual-studio-code>

### II.2 CODINGS STANDARDS

- PHPCS et PHPMD dans phpstorm : vérifier que les notifications fonctionnent

### II.3 VERSIONNING

- Ajouter les sources du projet en vérifiant que les sous répertoires communautaires sont ignorés dans le .gitignore à la racine du projet (core et modules)

### II.4 SETTINGS DRUPAL

#### II.4.a Configurer un ou plusieurs sites

- Créer un répertoire `/var/www/drupal8/web/sites/d8-lille` en y copiant le contenu de `sites/default`
- Créer le fichier `sites/sites.php` à partir du fichier `example.sites.php`, activer le dossier `d8-lille`

#### II.4.b Désactiver le cache twig

- Identifier les paramètres du service twig dans `sites/default/default.services.yml` afin de désactiver le cache Twig dans `sites/development.services.yml` (vérifier en inspectant la source dans le browser).

#### II.4.c Désactiver le cache Drupal

- Copier le fichier `exemple.settings.local.php` vers `sites/d8-lille/settings.local.php` et désactiver le cache de rendu Drupal (`cache.backend.null`).

### II.5 DÉPANNAGE DE SITE AVEC DRUSH

```
# Récupérer l'accès administrateur (alias de user:login) :  
drush uli -l http://d8-lille.lxc
```

## III ENVIRONNEMENT DE PRODUCTION

### III.1 CONFIGURATION DE VARNISH

- Installer les packages du serveur:

```
apt install varnish  
service varnish status
```

- Installer le module Drupal :

```
cd /var/www/drupal8  
composer require drupal/adv_varnish
```

- Mettre à jour la configuration de Varnish :

```
sudo -u varnish cp modules/contrib/adv_varnish/varnish/default.vcl  
/etc/varnish/default.vcl
```

- Changer le port de Varnish pour utiliser le port 80 (Utilisation en Reverse Proxy) :

```
find /etc/ |grep varnish.service  
# sur ubuntu 16.04 :  
vim /etc/systemd/system/multi-user.target.wants/varnish.service  
# Modifier le port après l'argument -a (exemple -a :6081)  
ExecStart=/usr/sbin/varnishd -j unix,user=vcache -F -a :80 -T  
localhost:6082 -f /etc/varnish/default.vcl -S /etc/varnish/secret -s  
malloc,256m
```

- Changer le port de Apache pour utiliser le port 81 (Cible du reverse Proxy) :

```
# Editer le fichier /etc/apache2/sites-available/drupal8.conf  
<VirtualHost *:81>
```

- Vérifier en temps réel que le cache Varnish est utilisé :

```
varnishstat
```

## III.2 CONFIGURATION DE MEMCACHE

- Installer les packages du serveur (service et librairie php) :

```
apt install memcached php-memcache libmemcached-tools  
service memcached status  
service apache2 restart
```

- Installer le module Drupal :

```
cd /var/www/drupal18  
composer require drupal/memcache
```

- Configurer les settings du module à partir du README.txt :

**Instructions officielles pour le paramétrage avancé (exemples, répartition de charge en clusters) :**

- README.txt <http://cgit.drupalcode.org/memcache/tree/README.txt?h=8.x-2.x>

```
cd /var/www/drupal18/web  
drush en memcache memcache_admin -y
```

- Vérifier les statistiques d'utilisation (/admin/reports/memcache) :

```
# En temps réel :  
watch -td '(echo stats ; echo quit) | nc localhost 11211 | grep get_hits'
```

## III.3 CONFIGURATION DE DRUPAL

- Vérifier les recommandations de la page d'administration Status report (/admin/reports/status)
- Vérifier que le cache Drupal est actif dans les fichiers **services.yml** et **settings.php**
- Renseigner le nom de domaine dans le paramètre **trusted\_host\_patterns** du **settings.php**
- Désactiver le **rebuild\_access** setting (pour **bloquer l'accès à /update.php**)
- Vérifier les droits d'accès aux fichiers et aux sources (penser à **sudo -u www-data**)
- Désactiver le module **Devel**



## IV LES MODULES DRUPAL 8

### IV.1 MODULES COMMUNAUTAIRE

#### IV.1.a Installer un module

- Vérifier qu'un module existe : <https://www.drupal.org/project/{project name}>
- Identifier son ratio nombre de téléchargements / nombre d'installations en bas de page

Pour installer un module, la méthode recommandée est composer (exemple essentiel : admin toolbar)

```
composer require admin_toolbar:8.x
```

#### IV.1.b Activer un module

Pour activer un module, la commande recommandée est drush :

```
# Les commandes drush et drupal console fonctionnent depuis le répertoire  
# du fichier index.php  
cd /var/www/drupal8  
drush -l d8-lille.lxc en admin_toolbar -y  
# -l : le chemin du site (pour utiliser drush avec un multisite / env)  
# -y : accepter automatiquement l'action
```

- Utiliser Admin Toolbar pour vider le render cache
- Activer les modules field\_group, paragraph et field group
- Installer et activer le module examples

### IV.2 ADMINISTRATION DE BASE (PRÉ-REQUIS)

- Créer une nouvelle entité de type paragraphe
- Ajouter un field group rétractable autour d'un ensemble de champs de l'entité
- Ajouter un champ de article comme référence à l'entité paragraphe nouvellement créée

### IV.3 DEVEL

Devel est un outil d'assistance au développement

- Activer et installer la suite de modules Devel, Kint et Devel generate
- Générer du contenu automatiquement
- Accéder à l'onglet « Devel » sur les entités nouvellement générés

### IV.3.a Dumper des variables

Kint est comparable à `var_dump` mais plus puissant : il permet d'afficher des informations sur les méthodes de classes, de lire la phpdoc et de formater efficacement les imbrications de données (tableaux, objets).

```
Kint('kint is better than var dump') ;  
# Régler la profondeur des objets pour éviter les dépassements mémoire :  
kint_require();  
Kint::$maxLevels = 3;
```

### IV.3.b Exécuter du code

- Retrouver l'entité Paragraph précédente avec Kint, inspecter son contenu.

```
// la valeur de $id se cache dans les données de l'article  
Paragraph::load($id) ;
```

## IV.4 DÉINSTALLATION

- Supprimer toutes les instances d'entité node et paragraphe précédentes (/admin/modules/uninstall)

## IV.5 PATCHER UN CODE COMMUNAUTAIRE

- Ajouter [cweagans/composer-patches](#) avec composer
- Installer le module broken link en version 2.2
- Appliquer le patch [https://www.drupal.org/project/broken\\_link/issues/2938885](https://www.drupal.org/project/broken_link/issues/2938885)
- Vérifier que le module fonctionne

## IV.6 MODULES SPÉCIFIQUES

### IV.6.a Structure de base

- Faire apparaître un module et un thème custom dans le BO avec un dossier et un fichier info.yml

### IV.6.b Ajouter un plugin yml

- Sur la page qui liste les types de contenu, ajouter un item de menu de type « action » qui pointe vers l'édition des paragraphes. Créer un fichier **.links.action.yml** dans votre module à partir des informations suivantes :

**route\_name** : entity.paragraphs\_type.collection, **appears\_on** : entity.node\_type.edit\_form.

Documentation: <https://www.drupal.org/docs/8/api/menu-api/providing-module-defined-local-actions>