



# Installation de VisuMatrice sur CentOs/RHEL

Version 1.1.4

## Document

<b>Auteur</b>	Franck MEIGNEN	<b>Date de diffusion</b>	01/03/17
<b>Chef de projet</b>	Florent VEYRÈS	<b>N° de version</b>	1.1.4

## Évolution du document

Version	Auteur	Nature des changements	Date
1.0	Franck MEIGNEN	Création du document	
1.1	Axel DARMON	Mise à jour pour la v1.1.2	28/09/17
1.2	Axel DARMON	Mise à jour pour la v1.1.3	19/10/17
1.3	Axel DARMON	Mise à jour pour la v1.1.4	03/11/17

## Licence

Ce document n'est pas libre de droits.

Ce manuel est publié sous la licence Creative Commons avec les particularités "Paternité – Partage à l'identique" (également connue sous l'acronyme CC BY-SA).

Détails de cette licence : <http://creativecommons.org/licenses/by-sa/2.0/fr/>



## Table des matières

---

<b>1 PRÉAMBULE</b>	<b>4</b>
<b>2 INSTALLATION</b>	<b>5</b>
2.1 Pré-Requis logiciels	5
2.1.1 Installation des dépendances	5
2.1.1.1 PHP 5.5	5
2.1.1.1.1 Sous CentOS/RHEL 6	5
2.1.1.1.2 Sous CentOS/RHEL 7	5
2.1.1.2 MySQL	6
2.1.2 Récupération du package	6
2.1.3 Récupération de SIARD Suite ®	6
2.1.4 Installation du JRE JAVA	6
2.2 Installation de VisuMatrice	7
2.2.1 Configuration du vhost HTTP	7
2.2.2 Droits et liens des utilisateurs sur les répertoires	8
2.2.3 Configuration de l'application	8
2.2.4 Exécution	8
2.2.5 Mise en place du SGBD	8
2.2.5.1 Pour MySQL	8
2.2.5.2 Pour Oracle	9
2.2.6 Export du premier fichier SIARD	9
2.2.6.1 Mode graphique	9
2.2.6.2 Ligne de commande	10
2.2.6.3 Pour MySQL	10
2.2.6.4 Pour Oracle	10
2.2.7 Configuration de la liaison	11
2.2.7.1 Pour MySQL	11
2.2.7.2 Pour Oracle	11
<b>3 FIN D'INSTALLATION</b>	<b>12</b>
<b>4 MISE À NIVEAU DEPUIS LA VERSION ANTÉRIEURE</b>	<b>13</b>
<b>5 PROBLÈMES DE CONFIGURATION</b>	<b>14</b>
5.1 Cookies de session	14
5.2 Réécriture d'url	14

# 1. PRÉAMBULE

Ce document décrit la procédure d'installation de la plate-forme Visumatrices v1.1.4 en production.

Ce guide d'installation-type est à destination de techniciens expérimentés.

Cette instance peut être physique ou virtuelle.

## 2. INSTALLATION

### 2.1. Pré-Requis logiciels

- PHP 5.5
- MySQL 5.6.25 (L'application est compatible avec **Oracle Database** mais l'utilisation de MySQL est fortement conseillée. L'installation d'Oracle et de ses dépendances ne seront pas développées ici).
- CakePHP 2 (la version 2.8.6 est requise pour une compatibilité avec PHP 7)

Pendant l'installation, le serveur doit être connecté sur un réseau relié à internet afin de récupérer et installer les dernières mises à jour de composants logiciels disponibles.

#### 2.1.1. Installation des dépendances

##### 2.1.1.1. PHP 5.5

La version 5.5 de PHP est requise, nous utilisons un dépôt externe :

###### 2.1.1.1.1. Sous CentOS/RHEL 6

La distribution Linux qualifiée et utilisée ici est Centos 6 x64. La configuration citée dans cette documentation doit être modifiée et adapté à vos choix.

```
rpm -Uvh http://dl.fedoraproject.org/pub/epel/6/x86_64/epel-release-6-8.noarch.rpm
rpm -Uvh http://rpms.famillecollet.com/enterprise/remi-release-6.rpm
```

Activer le repository php 5.5 (enabled=1)

```
vi /etc/yum.repos.d/remi.repo

[remi]
name=Remi's RPM repository for Enterprise Linux 6 - $basearch
#baseurl=http://rpms.remirepo.net/enterprise/6/remi/$basearch/
mirrorlist=http://rpms.remirepo.net/enterprise/6/remi/mirror
enabled=1
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-remi

[remi-php55]
name=Remi's PHP 5.5 RPM repository for Enterprise Linux 6 - $basearch
#baseurl=http://rpms.remirepo.net/enterprise/6/php55/$basearch/
mirrorlist=http://rpms.remirepo.net/enterprise/6/php55/mirror
# NOTICE: common dependencies are in "remi-safe"
enabled=1
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-remi
```

Mettre à jour les dépôts YUM :

```
yum update
```

###### 2.1.1.1.2. Sous CentOS/RHEL 7

La distribution Linux qualifiée et utilisée ici est Centos 7 x64. La configuration citée dans cette documentation doit être modifiée et adapté à vos choix.

```
rpm -Uvh http://dl.fedoraproject.org/pub/epel/7/x86_64/e/epel-release-7-9.noarch.rpm
rpm -Uvh http://rpms.famillecollet.com/enterprise/remi-release-7.rpm
```

Activer le repository php 5.5 (enabled=1)

```
vi /etc/yum.repos.d/remi.repo

[remi]
name=Remi's RPM repository for Enterprise Linux 7 - $basearch
#baseurl=http://rpms.remirepo.net/enterprise/7/remi/$basearch/
mirrorlist=http://rpms.remirepo.net/enterprise/7/remi/mirror
enabled=1
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-remi

[remi-php55]
name=Remi's PHP 5.5 RPM repository for Enterprise Linux 7 - $basearch
#baseurl=http://rpms.remirepo.net/enterprise/6/php55/$basearch/
mirrorlist=http://rpms.remirepo.net/enterprise/7/php55/mirror
# NOTICE: common dependencies are in "remi-safe"
enabled=1
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-remi
```

Mettre à jour les dépôts YUM :

```
yum update
```

### 2.1.1.2. MySQL

```
yum install wget httpd php php-mysql
```

Définir le démarrage automatique du service lors du démarrage du serveur

```
systemctl enable httpd
```

### 2.1.2. Récupération du package

```
cd /var/www && wget https://adullact.net/frs/download.php/file/xxxx/VisuMatrice_V1.1.4.tar.gz
```

Remplacez xxxx par le numéro de version de la forge de l'Adullact.

On copie l'archive vers le répertoire cible d'apache pour la décompresser :

```
// décompression
tar -zxvf VisuMatrice_V1.1.4.tar.gz

// suppression du package
rm VisuMatrice_V1.1.4.tar.gz
```

### 2.1.3. Récupération de SIARD Suite ®

L'application requiert la transformation de fichiers .siard en données SQL compatibles MySQL ou Oracle. Pour cela, il est indispensable de télécharger l'utilitaire SIARD Suite ®.

L'URL ci-après (<https://www.bar.admin.ch/bar/en/home/archiving/tools/siard-suite.html>) est donnée à titre indicatif : SIARD Suite ® est un outils propriétaire complètement indépendant de Visumatrice.

### 2.1.4. Installation du JRE JAVA

Une version de JAVA JRE est nécessaire pour l'exécution du programme SIAD Suite, nous installons une version 1.6. La version du JRE java n'étant plus présente dans les dépôts Centos/RHEL, nous allons nous procurer la version 6 depuis le site d'ORACLE.

Récupérer la dernière version du JRE java et déposer le sur le serveur dans le répertoire `/opt` :

(<http://www.oracle.com/technetwork/java/javase/downloads/jre6downloads-1902815.html>)

Donner les droits d'exécution au JRE

```
chmod +x jre-6u45-linux-x64.bin
```

Lancer son installation :

```
./jre-6u45-linux-x64.bin
```

Créer le lien symbolique :

```
ln -s /opt/jre1.6.0_45 /opt/jre
```

Définir en tant que JRE par défaut pour le système :

```
update-alternatives --install /usr/bin/java java /opt/jre/bin/java 1
```

La commande `java -version` nous retourne alors ces informations :

```
java version "1.6.0_45"  
Java(TM) SE Runtime Environment (build 1.6.0_45-b06)  
Java HotSpot(TM) 64-Bit Server VM (build 20.45-b01, mixed mode)
```

Si ce n' est pas le cas, il est nécessaire de choisir via cette commande le bon JRE

```
update-alternatives --config java
```

Les sources sont à copier dans le répertoire de CakePHP. En cas de mise à jour depuis la version précédente, se référer au chapitre [Mise à niveau depuis la version antérieure](#).

## 2.2. Installation de VisuMatrice

### 2.2.1. Configuration du vhost HTTP

Copie du virtualhost modèle dans le répertoire de configuration d'apache:

```
cp /var/www/Matrices/matrices /etc/httpd/conf.d/matrices.conf
```

Éditer le virtualhost et ajouter la valeur `ServerName` correspondant à votre URL d'accès.

```
vi /etc/httpd/conf.d/matrices.conf
```

Afin d'être compatible apache 2.4, remplacer les valeurs :

```
Options Indexes FollowSymLinks MultiViews  
Order allow,deny  
allow from all
```

par

```
Options -Indexes +FollowSymLinks +MultiViews
Require all granted
```

## 2.2.2. Droits et liens des utilisateurs sur les répertoires

On applique les droits d'utilisateurs pour le profil apache sur tous les répertoires de l'application :

```
chown -R apache: /var/www/Matrices*
chmod -R 775 /var/www/Matrices/
```

## 2.2.3. Configuration de l'application

On copie les fichiers de configuration par défaut et on les renomme pour qu'il soit utilisable par l'application :

```
cd /var/www/Matrices/app/Config/
cp database.php.default database.php
```

Une modification du fichier **php.ini** est nécessaire afin de définir le fuseau horaire :

```
vi /etc/php.ini
```

ajouter la valeurs

```
date.timezone = "Europe/Paris"
```

## 2.2.4. Exécution

On redémarre apache pour prendre en compte les modifications apportées sur la configuration :

```
systemctl restart httpd
```

## 2.2.5. Mise en place du SGBD

### 2.2.5.1. Pour MySQL

Nous installons depuis les dépôts officiels Mysql la version 5.6 :

```
rpm -Uvh https://dev.mysql.com/get/mysql57-community-release-el7-9.noarch.rpm
```

Activer le dépôt pour MySQL 5.6 et désactiver la version 5.7

```
vi /etc/yum.repos.d/mysql-community.repo

# Enable to use MySQL 5.6
[mysql56-community]
name=MySQL 5.6 Community Server
baseurl=http://repo.mysql.com/yum/mysql-5.6-community/el/7/$basearch/
enabled=1
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-mysql

[mysql57-community]
name=MySQL 5.7 Community Server
baseurl=http://repo.mysql.com/yum/mysql-5.7-community/el/7/$basearch/
enabled=0
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-mysql

yum install mysql-community-server
```



Définir le démarrage automatique du service lors du démarrage du serveur :

```
systemctl enable mysqld
```

Démarrer MySQL:

```
systemctl start mysqld
```

Configurer Mysql et définir le mot de passe **root**, valider toutes les options par défaut.

```
mysql_secure_installation
```

Création de l'utilisateur et de la base de données :

```
mysql -u root -p
```

```
CREATE USER matrices@localhost IDENTIFIED BY 'motdepasse' ;
CREATE DATABASE MATRICE;
GRANT ALL PRIVILEGES ON *.* TO matrices@localhost ;
GRANT CREATE ON *.* TO matrices@localhost ;
```

Remplacer **motdepasse** par le mot de passe de votre choix. Dans le cas où l'archive .siard est exportée depuis une autre machine que le serveur, remplacer **localhost** par %, modifier le fichier de configuration de MySQL afin de commenter la ligne commençant par bindaddress avec un #.

```
vi /etc/mysql/my.cnf
/etc/init.d/mysql restart
```

## 2.2.5.2. Pour Oracle

L'installation d'Oracle n'est pas abordée ici. Elle est indépendante de l'application.

Pour que les scripts s'exécutent correctement, il faut configurer les variables globales (il est conseillé de mettre ces commandes dans le fichier .bashrc afin de n'avoir aucun problème lors des futurs exports :

```
export PATH=$PATH:/usr/lib/oracle/xe/app/oracle/product/$version/server/bin
export ORACLE_HOME=/usr/lib/oracle/xe/app/oracle/product/$version/server
export ORACLE_SID=XE
```

Note : Remplacer **\$version** par le numéro de version de Oracle.

## 2.2.6. Export du premier fichier SIARD

Pour que l'application fonctionne, il faut exporter un premier fichier .siard à l'aide du logiciel SIARD Suite®. Ce fichier doit être intégralement exporté avant de poursuivre.

### 2.2.6.1. Mode graphique

Lancer dans en ligne de commande :

```
java -jar /$path_de_l_application/bin/SiardEdit.jar
```

Note : Remplacer **\$path\_de\_l\_application** par le chemin vers le répertoire où a été installé le logiciel.

Ouvrir l'archive .siard.

Choisir le menu File / Load into database (TRES LONG)

Attention ! La base de transition par défaut s'appelle MATRICE. Il est risqué de changer ce nom car les scripts d'export automatique l'intègrent. Ne fermez surtout pas la fenêtre avant la fin du traitement.

## 2.2.6.2. Ligne de commande

Lancer la commande d'export en spécifiant :

- le chemin du fichier SIARD que vous avez au préalable positionné sur le serveur, ex `:-s /root/monfichier.siard`
- le login et mot de passe SQL définis précédemment, ex `:-u matrices -p motdepasse`
- le SGBD, ex `:-d MySQL`
- l'adresse port et nom de la base, ex `:-n 127.0.0.1:3306/MATRICE`
- une valeur de timeout, ex `:-q 30000`
- le schéma, ex `:MATRICE`

Lancez ensuite la commande d'import :

```
java -cp bin/SiardEdit.jar ch.admin.bar.siard.SiardToDb -s /root/$monfichier.siard -d $SGBD -n $host:$port/$schema -u $user -p $password -q $timeout matrice $schema
```

Ensuite, il faut transférer les données de la base transitoire vers la base applicative.

```
cd /var/www/Matrices/app/webroot/files/sh/  
chmod +x *.sh
```

## 2.2.6.3. Pour MySQL

```
./mysqllexport.sh matrices $loginmatrice $passwordmatrice $hostmatrice $loginapplicative  
$passwordapplicative $hostapplicative
```

Note : Remplacer `$loginmatrice`, `$passwordmatrice`, `$hostmatrice` par les informations de la base de donnée créée précédemment, `$loginapplicative`, `$passwordapplicative`, `$hostapplicative` par les informations du serveur ou sera créée la base applicative (si elle a été créée sinon, entrez les mêmes informations que pour la base de donnée créée précédemment ; la base sera créée automatiquement pour le même utilisateur avec le même mot de passe sur le même serveur).

## 2.2.6.4. Pour Oracle

```
./oracleexportfirst.sh $loginmatrice $passwordmatrice $hostmatrice $portmatrice $SIDmatrice  
$loginapplicative $passwordapplicative $hostapplicative $portapplicative $SIDapplicative
```

Note : Remplacer `$loginmatrice`, `$passwordmatrice`, `$hostmatrice`, `$portmatrice`, `$SIDmatrice` par les informations de la base de donnée créée précédemment, `$loginapplicative`, `$hostapplicative`, `$passwordapplicative`, `$portapplicative`, `$SIDapplicative` par les

informations du serveur ou sera créée la base applicative (il est conseillé de l'installer sur le même serveur). Pour Oracle, il est nécessaire de créer les deux bases au préalable.

## 2.2.7. Configuration de la liaison

### 2.2.7.1. Pour MySQL

Il faut ensuite modifier le fichier de l'application afin de la connecter à la base de données :

```
cp /var/www/Matrices/app/Config/database.php.default /var/www/Matrices/app/Config/database.php
vi /var/www/Matrices/app/Config/database.php
```

et entrer les valeurs de la base de données à partir de la ligne :

Le fichier doit ressembler à ceci :

```
// Base applicative
public $default = array(
    'datasource' => 'Database/MySQL',
    'persistent' => false,
    'host'=> '', // remplir l'adresse du serveur de base de donnée
    'login' => '', // rentrer le nom de l'utilisateur créé
    'password' => '', // rentrer le mot de passe de l'utilisateur
    'database' => 'applicative', // ne pas changer si la base de production a gardé ce nom
    'prefix' => '',
    'encoding' => 'utf8'
);

// Base de transition
public $matrice = array(
    'datasource' => 'Database/MySQL',
    'persistent' => false,
    'host' => '', // remplir l'adresse du serveur de base de donnée
    'login' => '', // rentrer le nom de l'utilisateur créé
    'password' => '', // rentrer le mot de passe de l'utilisateur
    'database' => 'MATRICE', // ne pas changer si la base de transition a gardé ce nom
    'prefix' => '',
    'encoding' => 'utf8'
);
```

### 2.2.7.2. Pour Oracle

Dans le fichier `database.php`, supprimer le datasource MySQL et décommenter les lignes pour Oracle. Pour le chemin `/usr/lib/oracle/xe/app/oracle/product/$version/server`, remplacer `$version` par le bon numéro de version.

### 3. FIN D'INSTALLATION

A la fin de l'exécution du script d'export, un utilisateur **admin** avec mot de passe **admin** existe et permet la connexion à l'interface d'administration.

## 4. MISE À NIVEAU DEPUIS LA VERSION ANTÉRIEURE

Si vous n'effectuez pas une installation nouvelle mais que vous mettez à jour votre installation depuis la version précédente, la procédure est plus simple. - Vérifiez que les pré-requis sont respectés. Ensuite, récupérez le package de l'application comme indiqué au chapitre « Récupération du package », notamment pour ce qui concerne la version de CakePHP. - Remplacez le contenu du répertoire app dans le dossier de l'application sur le serveur par celui de l'archive. - Renommez le fichier `database.php.default` en `database.php` et éditez-le en complétant les tableaux avec les informations de votre installation (type de base de données, hôte, identifiant et mot de passe de connexion, etc.). La structure de la base de données n'a pas changé donc aucune installation n'est nécessaire de ce côté. De même, il n'est pas nécessaire de modifier votre **virtual host** pour accéder à l'application.

Pour éviter d'éventuels effets curieux, il est souhaitable de réinitialiser le cache de votre navigateur.

## 5. PROBLÈMES DE CONFIGURATION

### 5.1. Cookies de session

Dans certains cas, si vous aviez déjà une instance de VisuMatrice installée, vous pouvez avoir un conflit de cookies de session. Vous pouvez corriger le problème en corrigeant le `app/Config/core.php`.

```
Configure::write('Session', [  
    'defaults' => 'php',  
    'cookie'   => 'my_app1'  
]);
```

### 5.2. Réécriture d'url

Dans certains cas, les styles css et les images ne seront pas accessibles. Le problème est dû à la configuration de la réécriture d'url. Tentez de corriger le problème sur votre serveur (en principe apache2). Si cela ne suffit pas, vous pouvez parfois corriger le problème en éditant le fichier `app/Config/core.php` et en dé-commentant la ligne :

```
Configure::write('App.baseUrl', env('SCRIPT_NAME'));
```