

Mise à jour de version en pod V3

- Avant une MAJ
 - Annoncez la maj aux utilisateurs
 - Le jour J
- Commandes générales de mise à jour
 - Via le makefile
 - Via les commandes
 - Mise à jour des paramètres
 - Base de données
 - Serveur d'encodage
- Optionnel - Mise à jour d'Opencast Studio

Avant une MAJ

Annoncez la maj aux utilisateurs

Dans l'administration de Pod, (https://VOTRE_SERVEUR/admin/main/configuration/), vous trouverez :

- Le champ "maintenance_text_sheduled" vous permet de définir un message de maintenance personnalisé.
- Le champ "maintenance_sheduled" vous permet d'afficher/masquer (=1 / 0) ce message sur Pod.

Le jour J

Basculez en mode maintenance (maintenance_mode = 1), cela va désactiver certaines fonctionnalités, et afficher un bandeau "*Maintenance en cours. Certaines fonctionnalités sont indisponibles*".

Commandes générales de mise à jour

Via le makefile

```
pod@pod:~$ cd django_projects/podv3/
pod@pod:~/django_projects/podv3$ workon django_pod3
(django_pod3) pod@pod:~/django_projects/podv3$ make upgrade

(django_pod3) pod@pod:~/django_projects/podv3$ sudo systemctl restart uwsgi-pod
```

Via les commandes

```
pod@pod:~$ cd django_projects/podv3/
pod@pod:~/django_projects/podv3$ workon django_pod3
(django_pod3) pod@pod:~/django_projects/podv3$ git status
(django_pod3) pod@pod:~/django_projects/podv3$ git pull origin master
(django_pod3) pod@pod:~/django_projects/podv3$ pip3 install -r requirements.txt
(django_pod3) pod@pod:~/django_projects/podv3$ python manage.py makemigrations
(django_pod3) pod@pod:~/django_projects/podv3$ python manage.py migrate
# mise à jour des composants js/css via yarn
(django_pod3) pod@pod:~/django_projects/podv3$ cd pod; yarn upgrade; cd ..
# Attention : avant de lancer collectstatic --clear, assurez-vous d'avoir sauvegardé le dossier static/custom si
vous y avez mis des fichiers personnalisés.
(django_pod3) pod@pod:~/django_projects/podv3$ python manage.py collectstatic --no-input --clear
(django_pod3) pod@pod:~/django_projects/podv3$ sudo systemctl restart uwsgi-pod
```

Mise à jour des paramètres

Après avoir fait une mise à jour d'Esup-Pod, la commande ci-dessous permet de connaître les nouveaux paramètres par rapport à une version précédente :

```
(django_pod3) pod@pod:~/django_projects/podv3$ python manage.py compareconfiguration *VERSION_PRECEDENTE*
```

par exemple, la commande

```
(django_pod3) pod@pod:~/django_projects/podv3$ python manage.py compareconfiguration 3.1.1
```

va lister tous les paramètres nouveaux (et ceux plus utilisés) depuis la 3.1.1 jusqu'à la version actuelle.

Base de données

Si vous mettez à jour depuis une version antérieure à Pod version 3.3.1 et que vous êtes sous MySQL ou MariaDB, il faut installer le timezone dans le moteur SQL (à faire en tant que root mysql !)

```
$ mysql_tzinfo_to_sql /usr/share/zoneinfo | mysql -u root -p --database=mysql
```

Serveur d'encodage

Attention à bien faire la mise à jour sur vos serveurs d'encodages !

```
pod@pod-encodage:~$ cd django_projects/podv3/  
pod@pod-encodage:~/django_projects/podv3$ workon django_pod3  
(django_pod3) pod@pod-encodage:~/django_projects/podv3$ git status  
(django_pod3) pod@pod-encodage:~/django_projects/podv3$ git pull origin master  
(django_pod3) pod@pod-encodage:~/django_projects/podv3$ pip3 install -r requirements.txt  
(django_pod3) pod@pod-encodage:~/django_projects/podv3$ sudo /etc/init.d/celeryd restart
```

Optionnel - Mise à jour d'Opencast Studio

Pour mettre à jour le studio d'Opencast dans votre instance de Esup-Pod, voici les étapes à suivre :

- Rendez-vous dans le dossier opencast-studio/
- Récupérer la dernière version d'Opencast Studio via la commande suivante :

```
# Choisir un tag récent car la branche master est sur la version 2.0 qui est un redesign complet  
# la liste des tags: https://github.com/elan-ev/opencast-studio/tags  
git checkout tags/2023-09-14  
  
git pull
```

- Pour les versions jusqu'à 2023-09-14 : régénérez l'Opencast Studio avec la bonne configuration pour Pod via les commandes suivantes :

```
export PUBLIC_URL=/studio  
npm install  
npm run build
```

Pour les versions plus récentes (tags > 2023-10-10), les commandes diffèrent légèrement :

```
export PUBLIC_PATH=/studio  
npm install  
npm run build:release
```

- Le répertoire **build** est alors mis à jour. Renommez-le en **studio**, puis copier le dans le répertoire **pod/custom/static/opencast/**

```
mkdir -p pod/custom/static/opencast/studio  
cp -r build/* pod/custom/static/opencast/studio
```

- Finalement, n'oubliez pas de collecter vos fichiers statiques pour la mise en production via la commande :

```
(django_pod) [userpod@video][/data/www/userpod/django_projects/podv2] python manage.py collectstatic
```