

Mise en place du système de diffusion de webinaires avec couplage BBB/POD (bbb-pod-live)



Cette documentation ne concerne que la version v2.X de Pod et non la version 3.X de Pod.

Le module utilisé sera supprimé de Pod dans une future version de Pod.

Une solution à cette problématique est en cours de travaux.

- Contexte et solution apportée
 - Contexte
 - Solution
 - Remarques importantes
- Architecture de la solution
 - Choix du plugin permettant la diffusion de la session BigBlueButton en direct
 - Installation et configuration
 - Prérequis
 - Installation et configuration à réaliser sur les serveurs bbb-pod-live
 - Installation des prérequis de BigBlueButton-liveStreaming
 - Installation Docker et Docker-compose sous CentOS 7
 - Installation Docker
 - Installation Docker-compose
 - Gestion des droits
 - Installation Docker et Docker-compose sous Ubuntu 20.04
 - Installation de bbb-pod-live
 - Configuration de bbb-pod-live
 - Fichier docker-compose.yml
 - Explications concernant l'utilisation du tchat
 - Explications concernant l'utilisation de l'enregistrement du direct
 - Mise en place du job CRON
 - Installation et configuration à réaliser sur Pod
 - Migration depuis une version 2.6 ou 2.7 de Pod
 - Modification de la périodicité du job CRON existant
 - Configuration dans Pod
 - Récapitulatif des étapes à réaliser pour l'installation
 - Synthèse du processus global
 - Démarrage d'un direct
 - Arrêt d'un direct
 - Exploitation
 - L'interface d'administration
 - Les directs BigBlueButton
 - Exploitation du script lancé par le job CRON du serveur Pod
 - Exploitation du script lancé par le job CRON du serveur bbb-pod-live
 - Exploitation courante/exceptionnelle
 - Un enseignant ne voit aucune session BBB en cours
 - Arrêter un direct BigBlueButton manuellement



Mise à jour du script bbb-pod-live.php

Le script `pod/scripts/bbb-pod-live.php` sera mis à jour dans la prochaine version **v2.8.2** de Pod. Les modifications réalisées sont les suivantes :

- Modification du nom du diffuseur créé : les diffuseurs créés par ce script commenceront par [BBB] et non (BBB) (*cela permet d'éviter des problèmes en lien avec l'API Rest*).
- Il est maintenant possible de streamer une session BigBlueButton dont le nom contient une quote.
- Un email est envoyé aux administrateurs en cas de démarrage d'un direct.

Contexte et solution apportée

Contexte

Suite à la pandémie de COVID-19, il a été mis en place, à l'université de Montpellier, un système de classe virtuelle Open Source reposant sur Big Blue Button (BBB).

Pour informations, BigBlueButton (<https://bigbluebutton.org/>) est un outil de classe virtuelle ayant les fonctionnalités suivantes :

- Vidéo/webcam
- Audio
- Chat
- Partage de document + annotation
- Partage d'écran
- Sondage
- Enregistrement
- Création de groupes
- Prises de notes partagées
- Intégration de vidéos externes
- Intégration Moodle et WordPress

Cependant, l'une des plus grosses contraintes de BigBlueButton concerne la limite de 100 étudiants par session (cf. <https://docs.bigbluebutton.org/support/faq.html#how-many-simultaneous-users-can-bigbluebutton-support>).

Dans la plupart des cas, cette limite de 100 étudiants par session n'est pas un blocage, mais dans certains cas bien précis - par exemple, des cours magistraux pour des L1 - cette limite devient bloquante à l'utilisation de BigBlueButton.

Solution

Pour résoudre cette problématique, je suis parti d'une solution mise en place à l'Université Polytechnique Hauts-de-France et l'ai modifié pour l'intégrer totalement dans Pod v2.

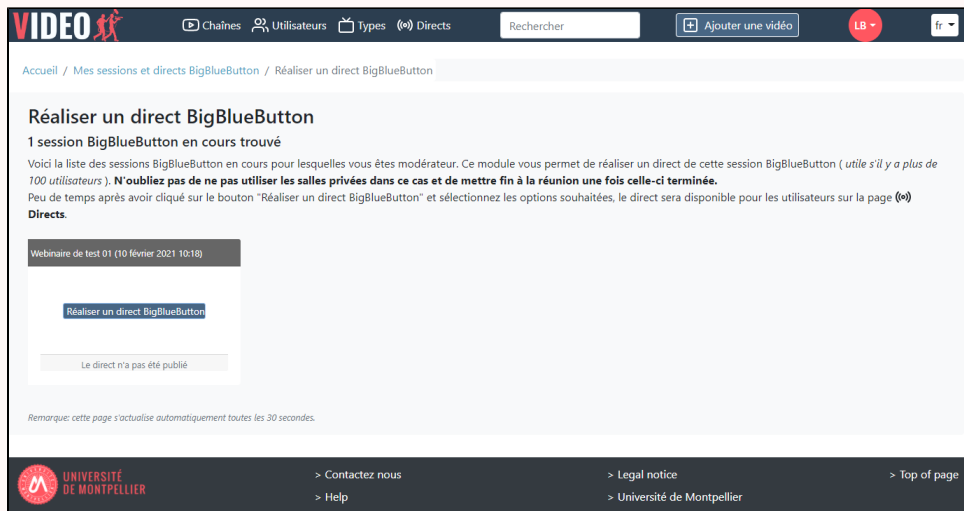
Cette solution fonctionne pour BigBlueButton mais également pour Scalelite (cf. <https://github.com/blindsidenetworks/scalelite>), un système de répartition de charge pour BigBlueButton.

L'idée de cette solution est de :

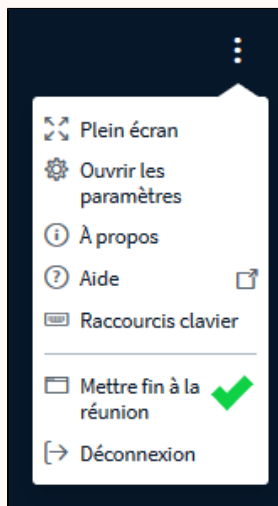
- Récupérer régulièrement les informations concernant les sessions en cours, ainsi que les usagers, dans BigBlueButton pour les insérer dans la base de données de Pod.
Cf. le système de publication des présentations Web de BigBlueButton vers Podv2 : <https://www.esup-portail.org/wiki/x/AgCBNg>
- Traiter ces données pour permettre aux utilisateurs, connectés dans Pod, de pouvoir réaliser un direct de leur session BigBlueButton en cours,



Il ne faut pas que l'utilisateur utilise de salles privées dans BigBlueButton. En effet, le système de salles privées de BigBlueButton ouvre des popups, et il pourrait arriver que le direct n'affiche que l'appel de la popup.



Il faut également que l'utilisateur mette fin à la réunion pour que le direct s'arrête (une déconnexion n'est pas suffisante et le direct continue alors).



- Pour réaliser ce direct, l'utilisateur devra valider les options possibles en validant un formulaire :

VIDEO

Chânes
Utilisateurs
Types
Directs
Rechercher
Ajouter une vidéo
LB
fr

Accueil / Mes sessions et directs BigBlueButton / Réaliser un direct BigBlueButton / Confirmation de la réalisation d'un direct BigBlueButton

Êtes-vous sûr de vouloir réaliser un direct BigBlueButton ?

Sélectionnez les options souhaitées puis validez ce formulaire en cliquant sur "Réaliser un direct BigBlueButton".

ⓘ Ce direct sera arrêté automatiquement à la fin de la session BigBlueButton.

☐ **ACCÈS RESTREINT**
Le direct est-il uniquement accessible aux utilisateurs authentifiés ?

☒ **AFFICHAGE DU CHAT PUBLIC**
Souhaitez-vous montrer le chat public en direct ?

☐ **ENREGISTRER LA SESSION DANS MES VIDÉOS**
Souhaitez-vous enregistrer la vidéo de cette session, à la fin du direct, directement dans Mes vidéos ?

☐ **ACTIVER LE CHAT**
Voulez-vous un chat sur la page de direct pour les étudiants ? Les messages envoyés dans le chat de cette page de direct se retrouveront dans le chat public de BigBlueButton.

Remarque: peu de temps après avoir cliqué sur le bouton "Réaliser un direct BigBlueButton", le direct sera disponible pour les utilisateurs sur la page **Directs**.

Réaliser un direct BigBlueButton

UNIVERSITÉ DE MONTPELLIER

> Contactez nous
> Help

> Legal notice
> Université de Montpellier

> Top of page

Ces options sont les suivantes :

- **Accès restreint** : Mettre l'accès restreint permet que le direct ne soit accessible qu'aux utilisateurs authentifiés, typiquement via le système d'authentification CAS de l'université. Sans accès restreint, le direct peut être accessible à tous.
- **Affichage du tchat public** : En cas d'activation, le tchat public sera affiché dans la partie gauche du direct.
- **Enregistrer la session dans Mes vidéos** : Cette option permet l'enregistrement de la vidéo du direct en même temps. Cela signifie qu'une fois le direct réalisé, le fichier vidéo sera automatiquement publié en mode Brouillon pour cet utilisateur et sera alors mis automatiquement dans la file d'attente pour encodage.



Cette option peut être désactivée dans le paramétrage de Pod et ne sera alors pas affiché aux utilisateurs (*cf. explications techniques ci-dessous*).

- **Activer le tchat** : Via cette option, un tchat sera affiché dans la page de ce direct de Pod, sous la vidéo en direct. Les messages envoyés dans le tchat de cette page de direct se retrouveront dans le tchat public de BigBlueButton. Ainsi, les étudiants qui consultent la page de direct pourront communiquer avec l'enseignant et autres usagers de BigBlueButton.



Pour informations, seuls les utilisateurs authentifiés sur la page de direct pourront envoyer des messages, même si le direct est accessible à tous.

Non authentifié, l'utilisateur obtiendra ce message :

/// Envoyer un message

Vous devez être authentifié pour envoyer un message.

Une fois authentifié, l'utilisateur pourra envoyer un message :

/// Envoyer un message

Vous pouvez envoyer un message (100 caractères maximum) à la session BigBlueButton. Il sera affiché dans les 15 à 30 secondes sur la vidéo en direct.

Envoyer





Dans BigBlueButton, le nom de l'utilisateur qui a envoyé le message sera bien entendu affiché :



Chat 11:10

(Loic Bonavent) Message envoyé depuis Pod vers BigBlueButton

Une fois le formulaire validé, le direct sera lancé en moins d'une minute.


VIDEO  Chaînes  Utilisateurs  Types  Directs LB fr

Accueil / Mes sessions et directs BigBlueButton / Réaliser un direct BigBlueButton

Réaliser un direct BigBlueButton


1 session BigBlueButton en cours trouvé

Voici la liste des sessions BigBlueButton en cours pour lesquelles vous êtes modérateur. Ce module vous permet de réaliser un direct de cette session BigBlueButton (*utile s'il y a plus de 100 utilisateurs*). **N'oubliez pas de ne pas utiliser les salles privées dans ce cas et de mettre fin à la réunion une fois celle-ci terminée.**

Peu de temps après avoir cliqué sur le bouton "Réaliser un direct BigBlueButton" et sélectionnez les options souhaitées, le direct sera disponible pour les utilisateurs sur la page 

Directs.

Webinaire de test 01 (27 janvier 2021 08:27)




Le direct n'a pas encore démarré

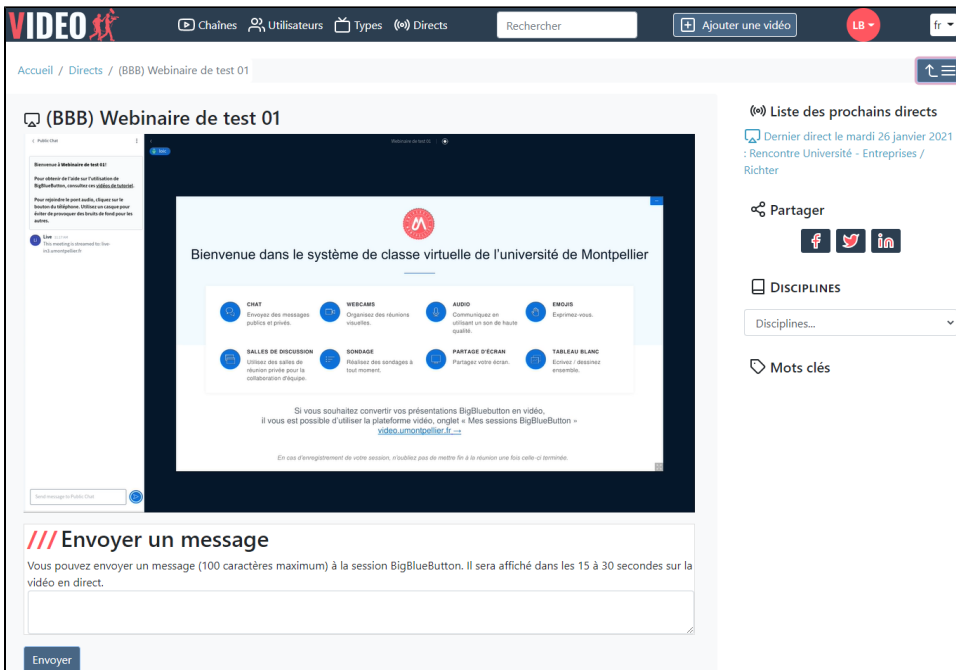
Le direct BigBlueButton a été publié et va bientôt démarrer.

×

Remarque: cette page s'actualise automatiquement toutes les 30 secondes.

 UNIVERSITÉ DE MONTPELLIER > Contactez nous > Legal notice > Top of page
> Help > Université de Montpellier

- Les étudiants pourront accéder au direct créé automatiquement, via l'onglet des directs de Pod.



Le système va créer automatiquement un diffuseur (terminologie Pod) dont le nom correspond à une concaténation de *(BBB)* et du *nom de la session dans BigBlueButton*.

Bien entendu, ce diffuseur sera automatiquement supprimé lors de l'arrêt de la session BigBlueButton.

Remarques importantes



Aucun impact sur BigBlueButton

Cette solution repose totalement sur Pod et n'impacte en rien BigBlueButton. Aucune modification n'est à réaliser côté BigBlueButton.



Les utilisateurs dans BigBlueButton

Les informations concernant les modérateurs dans BigBlueButton dépendent du client BBB utilisé : **Greenlight** ou le plugin **mod_bigbluebuttonn** pour Moodle.

Le système réalisé n'a été testé qu'avec le plugin **mod_bigbluebuttonn** pour Moodle (CASifié, donc les données proviennent de notre annuaire LDAP); cela signifie que les modérateurs sont définis - dans mon cas - sous la forme "Prenom Nom".

Il est possible de paramétrer ce format, via le paramètre **BBB_USERNAME_FORMAT**. A l'heure actuelle, les formats "Prenom Nom" ou "Nom Prenom" peuvent être gérés via ce paramétrage.

⚠ Ce point est crucial pour que le système fonctionne correctement : une correspondance doit exister sur le "Prenom Nom" ou "Nom Prenom" (selon la configuration BBB_USERNAME_FORMAT) des modérateurs de BigBlueButton et le "Prenom Nom" des utilisateurs dans la base de données de Pod.

? A priori, cela devrait pouvoir fonctionner avec **Greenlight** si celui-ci est configuré pour utiliser l'**annuaire LDAP** de l'établissement et les champs **givenName** et **sn**.

Architecture de la solution

Choix du plugin permettant la diffusion de la session BigBlueButton en direct

Pour diffuser une session BigBlueButton en direct, certains plugins existent déjà sur Github; il ne paraît pas raisonnable de redévelopper un tel système.

Pour arriver à faire mon choix, j'ai testé les 2 solutions suivantes :

- bbb-recorder (<https://github.com/jibon57/bbb-recorder>) : un plugin, indépendant de BigBlueButton, qui permet de convertir - via un script - une présentation Web BigBlueButton en fichier vidéo et qui permet également une diffusion en direct (flux RTMP) d'un cours BigBlueButton.
- BigBlueButton-liveStreaming (<https://github.com/aau-zid/BigBlueButton-liveStreaming>) : un plugin, indépendant de BigBlueButton, permettant de publier en live (via RTMP) une session BigBlueButton et de l'enregistrer.

Au final, voici ce qui est ressorti de mes tests.

	bbb-recorder	BigBlueButton-liveStreaming
Exporte un cours BBB en direct live	✓	✓
Enregistre un cours BBB en vidéo	✓ format webm, mp4	⚠ peut enregistrer le cours qu'il publie en live au format mkv
Indépendance vis-à-vis de BigBlueButton	✓ peut-être installé sur n'importe quel serveur	✓ peut-être installé sur n'importe quel serveur
Contenu de la vidéo finale	<ul style="list-style-type: none"> ✓ présentation ✓ audio ✓ vidéo ✓ partage d'écran ✓ chat ✓ whiteboard 	<ul style="list-style-type: none"> ✓ présentation ✓ audio ✓ vidéo ✓ partage d'écran ✓ chat ✓ whiteboard
Possibilité de chat depuis la page de direct avec BBB	—	✓
Possibilité de faire plusieurs streams sur une même VM	?	✓
Technologies	NodeJS, xvfb, Chrome, ffmpeg, shell	Docker, python, xvfb, ffmpeg, shell
Mise à jour régulière	✓	✓
Notion de charge	? 💡 Le fichier vidéo fait entre 2 et 4Mo par minute, en webm ou mp4	ⓘ 1 stream nécessite 4-6 vCPU + 4 Go RAM 💡 le fichier vidéo fait ~19Mo par minute, en mkv

Aux vues des besoins, et en particulier suite à la possibilité d'ajouter un tchat public à la page de direct, j'ai alors choisi **BigBlueButton-liveStreaming** comme solution pour la diffusion de sessions BigBlueButton en direct.

De plus, l'université Polytechnique Hauts-de-France avait déjà développé un système (reposant sur un script Bash) utilisant également BigBlueButton-liveStreaming, ce qui montre que ce plugin correspond bien aux besoins.

Installation et configuration

Prérequis

Techniquement, la solution repose sur :

- Le projet Github **BigBlueButton-liveStreaming**: installé sur les serveurs **bbb-pod-live** (sous la forme de containers Docker).
<https://github.com/aau-zid/BigBlueButton-liveStreaming>
- Un serveur de streaming pour la publication RTMP reposant sur **RTMP Nginx** (cf. documentation <https://www.esup-portail.org/wiki/x/BgC8KQ>)
- Le système existant de publication des présentations Web de BigBlueButton vers Pod (**CRON job**, **plugin BBB**).
- De nouvelles **API REST** pour Pod.
- Un script **PHP (CRON job)** pour le démarrage/arrêt des directs, création/suppression des diffuseurs...
 - Totalement **indépendant** des serveurs de Pod et de BigBlueButton.
 - Peut être installé sur des **serveurs spécifiques bbb-pod-live** (*load balancing simpliste*).
 - Ce script PHP peut gérer **plusieurs instances de BigBlueButton-liveStreaming**.

Installation et configuration à réaliser sur les serveurs bbb-pod-live

Installation des prérequis de BigBlueButton-liveStreaming

Pour installer les prérequis de BigBlueButton-liveStreaming, il suffit de consulter la documentation officielle du projet : <https://github.com/aau-zid/BigBlueButton-liveStreaming>

Au final, il suffit d'avoir **Docker** et **Docker-compose** qui tournent sur cette machine bbb-pod-live.



Il n'est pas utile d'installer BigBlueButton-liveStreaming à ce stade. C'est le script PHP qui installera automatiquement BigBlueButton-liveStreaming et qui initialisera sa configuration.

Installation Docker et Docker-compose sous CentOS 7

Personnellement, sous Cent OS 7, j'ai utilisé les documentations suivantes :

- Installation de Docker sous CentOS 7 : <https://www.digitalocean.com/community/tutorials/how-to-install-and-use-docker-on-centos-7>
- (Au cas où) Installation de Docker sous Scientific Linux 7 : <https://medium.com/@Dylan.Wang/how-to-install-docker-ce-on-scientific-linux-7-8-180e161f5947>
- Installation de Docker-compose sous CentOS 7 : <https://www.digitalocean.com/community/tutorials/how-to-install-and-use-docker-compose-on-centos-7>
- Installer Docker et Docker-compose depuis les dépôts officiels : <https://computerz.solutions/centos-docker/>

Au final, voici les commandes utilisées.

Installation Docker



Installation de Docker sous CentOS 7

```
#Anciennement
# sudo yum check-update
#curl -fsSL https://get.docker.com/ | sh

sudo yum install -y yum-utils device-mapper-persistent-data lvm2

sudo yum-config-manager --add-repo https://download.docker.com/linux/centos/docker-ce.repo

# Installation de la version 20.10.5
#La version 20.10.6 pose problème au serveur n'ayant pas d'adresse IP v6 (Error starting userland proxy: listen tcp6 [::]:9982: socket: address family not supported by protocol) cf. https://forums.docker.com/t/ipv6-disabled-on-my-computer-but-docker-network-seems-looking-for-it/107299
sudo yum install docker-ce-20.10.5 docker-ce-cli-20.10.5 containerd.io

sudo systemctl enable docker && sudo systemctl start docker
```

Installation Docker-compose



Installation de Docker-compose sous CentOS 7

```
sudo curl -L "https://github.com/docker/compose/releases/download/1.29.1/docker-compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose

sudo chmod +x /usr/local/bin/docker-compose

docker-compose --version
```

Gestion des droits

Ces droits permettent de pouvoir lancer docker sans sudo et d'éviter l'erreur de type *Permission denied*.



Gestion des droits sous CentOS 7

```
chown %USER% /usr/local/bin/docker-compose -R

#Si le groupe docker n'existe pas, on peut le créer
sudo groupadd docker

#Ajout de l'utilisateur à ce groupe
sudo usermod -aG docker %USER%

#Astuce permettant d'être sûr de ne plus avoir de problèmes de droit
sudo chmod 666 /var/run/docker.sock
```


⚠ Après avoir ajouté le user au groupe docker, ne pas oublier de fermer sa session et de se reconnecter pour que cela soit pris en compte.

Installation Docker et Docker-compose sous Ubuntu 20.04



Installation de Docker et Docker-compose sous Ubuntu 20.04

Sous Ubuntu 20.04, j'ai utilisé la documentation suivante : <https://docs.docker.com/engine/install/ubuntu/>

Ce qui donne :

```
sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/ubuntu \
$(lsb_release -cs) \
stable"
sudo apt-get update
sudo apt-get install docker-ce docker-ce-cli containerd.io
sudo docker run hello-world
docker -v
sudo apt install docker-compose
```



Informations importantes concernant bigbluebutton-liveStreaming

La solution reposant sur le projet Github **bigbluebutton-liveStreaming**, il est indispensable de s'assurer que ce dernier fonctionne correctement avant d'installer bbb-pod-live.

Pour ce faire, le mieux est de vérifier, au plus tôt, qu'il n'y ait pas de problème en lien avec ce projet et de réaliser un test simple, dès que possible, en suivant la documentation fournie : <https://github.com/aau-zid/BigBlueButton-liveStreaming>

Pour informations, lors de l'exécution de la commande `docker-compose up -d`, si vous obtenez une erreur de ce style : **ERROR: Couldn't connect to Docker daemon at http+docker://localhost - is it running?**

Cela signifie que docker nécessite d'être exécuté avec des privilèges. Dans ce cas, 2 possibilités :

- faire en sorte de pouvoir exécuter docker avec le **user connecté**.
- exécuter **sudo docker-compose up -d**. Par contre, dans ce cas, **il sera nécessaire de faire lancer le script bbb-pod-live.php via un CRON installé sous le compte root**.

Installation de bbb-pod-live

bbb-pod-live se présente sous la forme d'un simple script PHP (et d'un fichier de référence `docker-compose.default.yml`). Il faut donc avoir à minima :

- **PHP 7**,
- **php-xml**,
- **php-redis**

qui tourne sur cette machine.

Une fois PHP installé, il suffit de :

- Récupérer le répertoire **bbb-pod-live** de Pod, se trouvant dans le répertoire **scripts** (à la racine).



Répertoire bbb-pod-live

Ce répertoire bbb-pod-live va contenir les différentes instances de BigBlueButton-liveStreaming, qui peuvent contenir les vidéos enregistrées (lorsque l'enseignant a sélectionné l'enregistrement du direct).

Il faut donc prévoir un stockage suffisant (*typiquement un minimum de 5Go par instance de BigBlueButton-liveStreaming souhaitée*).

- Vérifier qu'il y ait bien le script **bbb-pod-live.php** et le fichier **docker-compose.default.yml** dans ce répertoire, avec les bons droits utilisateurs.

Configuration de bbb-pod-live

Pour l'instant, la configuration se réalise directement dans les premières lignes du script PHP. A terme, il faudra utiliser un fichier de configuration.

Voici la liste des éléments paramétrables :

Paramètre	Valeurs possibles / Exemples	Commentaires
Paramétrage nécessaire pour bbb-pod-live		
DEBUG	true/false	Application en mode débogage (true - on logue toutes les lignes) ou en production (false - on logue seulement les erreurs et infos).
PHYSICAL_BASE_ROOT	"/home/user/bbb-pod-live/"	Répertoire de base de l'application et avoir suffisamment d'espace disque pour l'enregistrement de quelques vidéos (stockage temporaire) ❗ Ce répertoire doit être sur un disque dur de la machine serveur.
PHYSICAL_LOG_ROOT	"/home/user/bbb-pod-live/logs/"	Constante permettant de définir le chemin physique du répertoire contenant les logs applicatifs. ❗ L'arborescence doit être sur un disque dur local de la machine serveur.
ADMIN_EMAIL	"administrateur@univ.fr"	Mail de l'administrateur de BBB-POD-LIVE, qui recevra les mails en cas d'erreur
SERVER_HOSTNAME	"server.infra.univ.fr"	Hostname de ce serveur BBB-POD-LIVE (utile pour Redis et le chat)
NUMBER_SERVERS	2	Nombre de serveurs BBB-POD-LIVE
SERVER_NUMBER	1	Numéro unique de ce serveur dans la liste des serveurs BBB-POD-LIVE Par exemple: s'il y a 2 serveurs BBB-POD-LIVE (NUMBER_SERVERS = 2), alors un serveur devra avoir SERVER_NUMBER=1 et l'autre SERVER_NUMBER=2
NUMBER_LIVES	2	Nombre de directs gérés par ce serveur (à adapter selon les ressources du serveur)
Paramétrage nécessaire pour BigBlueButton-liveStreaming (cf. https://github.com/aau-zid/BigBlueButton-liveStreaming)		
BBB_URL	"https://bbb.univ.fr/bigbluebutton/api"	URL du serveur BigBlueButton/Scalelite, avec la notion d'API
BBB_SECRET	"xxxxxxxxxxxxxxxxxxxxx xxxxxxxxx"	Clé secrète du serveur BigBlueButton/Scalelite
BBB_RESOLUTION	"1280x720"	Résolution pour diffuser / télécharger au format WxH (Défaut: 1920x1080). cf. BBB_RESOLUTION
FFMPEG_STREAM_VIDEO_BITRATE	"3000"	Bitrate de la vidéo (Défaut: 4000). cf. FFMPEG_STREAM_VIDEO_BITRATE
FFMPEG_STREAM_THREADS	"0"	Threads utilisés pour le flux (Défaut: 0). 0 signifie auto. cf. FFMPEG_STREAM_THREADS
BBB_STREAM_URL	"rtmp://liveX.univ.fr/live/"	Serveur RTMP qui va gérer les directs pour ce serveur bbb-pod-live cf. BBB_STREAM_URL Format, sans authentification : rtmp://serveurRTMP.domaine.fr:port/application/ Format, avec authentification : rtmp://user@password:serveur.domaine.fr:port/application/ Exemple : rtmp://live.univ.fr:1935/live/
BBB_ATTENDEE_PASSWORD	"xxxxxx"	Mot de passe des participants cf. BBB_ATTENDEE_PASSWORD ❗ Doit être défini comme le mot de passe du participant de Moodle / Greenlight ou de tout autre frontend pour permettre la participation via leurs liens
BBB_MODERATOR_PASSWORD	"xxxxxx"	Mot de passe des modérateurs cf. BBB_ATTENDEE_PASSWORD ❗ Doit être défini comme le mot de passe du modérateur de Moodle / Greenlight ou de tout autre frontend pour permettre la participation via leurs liens
Paramétrage nécessaire pour Pod		
POD_HLS_STREAM	"https://liveX.univ.fr/hls/"	Flux HLS, dépend de la configuration du serveur RTMP Nginx utilisé
POD_URL	"https://pod.univ.fr"	URL du serveur Pod
POD_TOKEN	"xxxxxxxxxxxxxxxxxxxxx xxxxxxxxx"	Token de sécurité de Pod, utile pour attaquer Pod via les API Rest (cf. administration de Pod / Jeton)
POD_ID_BUILDING	1	Identifiant du bâtiment POD (au sens live/building de POD) de rattachement des diffuseurs créés par bbb-pod-live

POD_DEFAULT_BBB_PATH	"/data/www/user/bbb-recorder/"	<p>Répertoire dans lequel copier les fichiers vidéo générés par BigBlueButton-liveStreaming</p> <p>⚠ Ce répertoire - typiquement un partage NFS - doit être accessible aussi par POD et correspondre à DEFAULT_BBB_PATH du fichier settings_local.py.</p> <p>ℹ Si ce n'est pas possible, laisser ce champ vide "" et positionner USE_BBB_LIVE_DOWNLOADING = False dans le settings_local de POD.</p>
----------------------	--------------------------------	--

Après avoir réalisé ce paramétrage, le mieux est de lancer manuellement, en mode DEBUG, le script **bbb-pod-live.php** via :

```
cd /home/user/bbb-pod-live/; php -f ./bbb-pod-live.php
```

Cette première exécution est importante car :

- cela va créer le **répertoire des logs** et le **fichier de log** correspondant à la journée,
- cela va créer un répertoire **bbb-live-streaming** par le nombre de directs gérés par ce serveur (*NUMBER_LIVES*).
- ✓ Par exemple, si 2 directs sont gérés par ce serveur, 2 répertoires seront créés dans le répertoire bbb-pod-live :
 - » bbb-live-streaming1
 - » bbb-live-streaming2
- pour chacun de ces répertoires bbb-live-streaming, un fichier de configuration de bbb-live-streaming, **docker-compose.yml**, sera positionné avec les valeurs par défaut dans ce répertoire (cf. paragraphe suivant).

Fichier docker-compose.yml

⚠ *Il est primordial de ne pas modifier le fichier docker-compose.default.yml : il servira de modèle pour les containers bbb-liveStreaming.*

Lors de la première exécution, le fichier de référence **docker-compose.default.yml** va être copié en tant que fichier **docker-compose.yml** et sera positionné dans chaque répertoire *bbb-live-streaming* créé :

Fichier docker-compose.yml

```
version: '3.3'
services:
  redis:
    image: redis
    ports:
      - "modifie_automatiquement:6379"
    networks:
      - app-tier
  bbb-streamer:
    image: aauzid/bigbluebutton-livestreaming
    container_name: liveStreaming1
    shm_size: '2gb'
    environment:
      # BigBlueButton Server url:
      - BBB_URL=modifie_automatiquement
      # BigBlueButton secret:
      - BBB_SECRET=modifie_automatiquement
      # BigBlueButton meetingID:
      - BBB_MEETING_ID=modifie_automatiquement
      # start meeting (optional):
      - BBB_START_MEETING=false
      # attendee password (optional - has to be set to the attendee password of moodle/greenlight or any other
      frontend to allow joining via their links):
      - BBB_ATTENDEE_PASSWORD=modifie_automatiquement
      # moderator password (optional - has to be set to the moderator password of moodle/greenlight or any
      other frontend to allow joining via their links):
      - BBB_MODERATOR_PASSWORD=modifie_automatiquement
      # meeting title (optional):
      - BBB_MEETING_TITLE=modifie_automatiquement
      # download / save BigBlueButton meeting
      - BBB_DOWNLOAD_MEETING=modifie_automatiquement
      # Media server url:
      - BBB_STREAM_URL=modifie_automatiquement
      # Resolution to be streamed/downloaded in format WxH (default 1920x1080)
      - BBB_RESOLUTION=modifie_automatiquement
      # stream video bitrate
      - FFMPEG_STREAM_VIDEO_BITRATE=modifie_automatiquement
      # threads used for stream (0=auto)
      - FFMPEG_STREAM_THREADS=modifie_automatiquement
      # Enable chat functionality
      - BBB_ENABLE_CHAT=modifie_automatiquement
      # shows the chat on the left side of the window (Default: false)
      - BBB_SHOW_CHAT=modifie_automatiquement
      # Timezone (default: Europe/Vienna):
      - TZ=modifie_automatiquement
      # Set REDIS host (default: 'redis')
      - BBB_REDIS_HOST=redis
      # Set REDIS channel to subscribe (default: 'chat')
      - BBB_REDIS_CHANNEL=modifie_automatiquement
      # Username for the chat (default: 'Chat')
      - BBB_CHAT_NAME=Chat
      # DEBUG
      - DEBUG=modifie_automatiquement
    networks:
      - app-tier
  volumes:
    - ./videodata:/video
networks:
  app-tier:
    driver: bridge
```

💡 Ce fichier de configuration est extrêmement important pour le bon fonctionnement de la solution.

Ce fichier correspond au paramétrage d'un container de bbb-liveStreaming; la documentation est accessible à l'adresse suivante : <https://github.com/aau-zid/BigBlueButton-liveStreaming>

L'idée de cette solution est que le paramétrage des différents containers **bbb-liveStreaming** soit généré automatiquement par **bbb-pod-live**.

Ainsi, le fait de lancer le script `bbb-pod-live.php` va modifier cette configuration `bbb-liveStreaming` et va remplacer quasiment toutes les valeurs des paramètres (ceux avec *modifie_automatiquement*).

Certains paramètres proviennent :

- de la *configuration bbb-pod-live* (`DEBUG`, `BBB_URL`, `BBB_SECRET`, `BBB_STREAM_URL...`),
- des *options sélectionnées par l'enseignant* dans l'interface de Pod (`BBB_ENABLE_CHAT`, `BBB_SHOW_CHAT`, ...),
- du script *bbb-pod-live.php* selon le contexte (`ports`, `BBB_MEETING_ID`, `BBB_MEETING_TITLE...`).

Explications concernant l'utilisation du tchat

Le fichier `docker-compose` ci-dessus permet d'utiliser un tchat sur la page de direct, si l'enseignant le souhaite.

Pour réaliser un chat avec BigBlueButton, on utilise `bbb-liveStreaming`. Voici comment cela fonctionne :

- `bbb-liveStreaming` utilise une base REDIS locale pour enregistrer les messages des étudiants,
- quand des messages sont publiés dans cette base REDIS, `bbb-liveStreaming` transfère ces messages à la session BigBlueButton au travers du client HTML5.



Problématique

★ La problématique initiale vient du fait qu'il est possible d'avoir plusieurs instances de `bbb-liveStreaming` par serveur `bbb-pod-live`, et que le serveur Web de Pod - qui contient le système de publication du chat - est un autre serveur que ceux de `bbb-pod-live`.

Pour résoudre, l'astuce a été d'utiliser 2 paramètres du fichier *docker-compose.yml* :

- le paramètre des **ports du container REDIS**, de la forme *"host:container"*.
- le paramètre du **channel REDIS** (`BBB_REDIS_CHANNEL`).

Ainsi, pour le 1° `bbb-liveStreaming`, il y aura les lignes suivantes :

Ports REDIS

```
ports:
- "6379:6379"
...
- BBB_REDIS_CHANNEL = chat1
```

Pour le 2° `bbb-liveStreaming`, il y aura les lignes suivantes :

Ports REDIS

```
ports:
- "6380:6379"
...
- BBB_REDIS_CHANNEL = chat2
```

Pour le 3° `bbb-liveStreaming`, il y aura les lignes suivantes :

Ports REDIS

```
ports:
- "6381:6379"
...
- BBB_REDIS_CHANNEL = chat3
```

Et ainsi de suite ...



Host:container

Il faut bien comprendre la signification de ces lignes; par exemple, la ligne "**6381:6379**" signifie que, pour publier un message depuis le chat se trouvant sur la page des directs de Pod, on devra attaquer la base REDIS se trouvant sur le serveur bbb-pod-live, sur le port **6381**.

Ensuite, avec cette configuration, les requêtes seront transmises au container REDIS sur le port 6379.

Ainsi, le container REDIS n'utilise que le port 6379, même s'il y a plusieurs instances de bbb-liveStreaming.

Tout se joue sur le nom du channel (chat1, chat2, chat3...) pour faire la différence.

⚠ En cas de Firewall, il est nécessaire de mettre en place les règles adéquates pour que le(s) serveur(s) bbb-pod-live puissent être accéder sur ces ports 6379, 6380... par les serveurs Web de Pod.

Explications concernant l'utilisation de l'enregistrement du direct

Si vous souhaitez que l'utilisateur puisse enregistrer son direct, il faut et configurer le **script bbb-pod-live.php** (paramètre **POD_DEFAULT_BBB_PATH**) et configurer Pod (paramètres **USE_BBB_LIVE_DOWNLOADING** et **DEFAULT_BBB_PATH** du **settings_local.py**).

Voici les étapes du système d'enregistrement d'un direct, si cela a été configuré pour et que l'utilisateur a sélectionné cette option (**⚠ cela consomme plus de ressources sur le serveur bbb-pod-live concerné**) :

- durant le direct, le plugin **bbb-liveStreaming** enregistre le fichier vidéo au format **mkv** dans le répertoire **videodata** correspondant,
- une fois la session BBB arrêté, le script **bbb-pod-live.php** va arrêter le container docker, supprimer le diffuseur, mettre à jour la base de données Pod et **copier le fichier mkv** - généré précédemment - dans le répertoire configuré via **POD_DEFAULT_BBB_PATH**,
- par la suite, ce fichier mkv sera pris en compte par le **job CRON de Pod**, qui va alors assigner la vidéo directement à l'utilisateur et va lancer l'encodage depuis les serveurs d'encodage.

Pour informations, il faut compter environ 20Mo par minute enregistrée.



Gestion des droits

Selon la configuration en lien avec Docker et Docker-compose, le fichier vidéo généré peut appartenir à root:root, avec les droits 644. Normalement, ces droits sont suffisants pour être copié puis encodé par les serveurs d'encodage.

A bien tester avant une mise en production effective.

Mise en place du job CRON

Il est nécessaire de faire tourner ce script régulièrement, typiquement toutes les minutes (*pour éviter que les enseignants/étudiants attendent le démarrage des directs*). Pour informations, s'il n'y a pas de direct à démarrer ou à arrêter, ce script ne consomme rien comme ressources.

Job CRON

```
crontab -e
*/1 * * * * /usr/bin/bash -c 'cd /home/%USER%/bbb-pod-live/; php -f ./bbb-pod-live.php'
```

⚠ Ce CRON tourne dans mon environnement. Il est possible que vous deviez le modifier à votre convenance, selon votre environnement (typiquement /usr/bin/bash ne correspond pas à votre environnement).

Installation et configuration à réaliser sur Pod

Migration depuis une version 2.6 ou 2.7 de Pod

Lors de la migration de Pod vers une version 2.8, lors de l'exécution de la commande *python manage.py makemigrations*, il sera demandé s'il y a eu un changement de modèle en lien **Attendee** en lieu et place de **User**.

C'est le cas, la classe Attendee vient remplacer User (cf. *bbb/models.py*): il faut alors répondre **Yes** à la question posée à ce moment de la migration.

Ainsi, dans la base de données de Pod, la table **bbb_user** est renommée en **bbb_attendee**.

Modification de la périodicité du job CRON existant

Comme expliqué préalablement, le système repose également sur le job CRON utilisé pour le système de publication des présentations Web (cf. <https://www.esup-portail.org/wiki/x/AgCBNg>).

Ce job CRON est à installer sur un serveur Pod de votre choix et devra tourner régulièrement (toutes les 2 minutes me paraît un délai correct).

Le script à lancer est positionné dans **django_projects/podv2/pod/video/management/commands/bbb.py** et permet de gérer les enregistrements effectués par BigBlueButton, mais surtout le fait qu'un enseignant ait lancé une session BBB.

i Concrètement, cela signifie qu'un enseignant, après avoir lancé une session dans BigBlueButton, verra dans Pod que sa session BBB est en cours et pourra réaliser un direct dessus, en attendant moins de 2 minutes.

Personnellement, mon CRON est configuré de la sorte :

Job CRON

```
crontab -e
*/2 * * * * /usr/bin/bash -c 'export WORKON_HOME=/data/www/%userpod%/.virtualenvs; export
VIRTUALENVWRAPPER_PYTHON=/usr/bin/python3.6; cd /data/www/%userpod%/django_projects/podv2; source /usr/bin
/virtualenvwrapper.sh; workon django_pod; python manage.py bbb main'
```

! Ce CRON tourne dans mon environnement. Il est possible que vous deviez le modifier à votre convenance, selon votre environnement (typiquement /usr/bin/bash ne correspond pas à votre environnement).

Configuration dans Pod

i Ne pas oublier d'avoir réalisé au préalable la configuration pour la publication des présentations Web (cf. [Mise en place de la publication des présentations Web de BigBlueButton vers Pod v2#Configuration dans Pod](#)).

Dans le fichier *settings_local.py*, ce système n'utilise que peu de nouveaux paramètres, à savoir :

Settings_local.py

```
# Allow the users to perform a BigBlueButton live
USE_BBB_LIVE = True
# Allow users to record their BigBlueButton live
USE_BBB_LIVE_DOWNLOADING = True
# Number maximum of BBB lives simultaneous managed by the system
BBB_NUMBER_MAX_LIVES = 5
```

! Le fait de mettre *USE_BBB_LIVE_DOWNLOADING = True* implique de mettre la configuration adéquat pour **DEFAULT_BBB_PATH** et également dans le script **bbb-pod-live.php** sur le serveur *bbb-pod-live*.

Les éléments de paramétrage sont les suivants :

- **USE_BBB_LIVE** : utilisation (True/False) de la gestion des directs BBB pour Pod
- **USE_BBB_LIVE_DOWNLOADING** : utilisation (True/False) de l'enregistrement des directs BBB pour Pod
- **BBB_NUMBER_MAX_LIVES** : nombre maximum de directs BBB simultanées gérés par le système (calculé par rapport au nombre de serveurs et d'instances de bbb-liveStreaming que vous avez configuré).



Par exemple, si vous avez configuré `BBB_NUMBER_MAX_LIVES = 5`, et qu'il y a déjà 5 directs BBB en cours, le prochain utilisateur qui souhaitera démarrer un direct BigBlueButton ne pourra pas le faire; il obtiendra un message correspondant à l'image ci-dessous et devra alors attendre qu'un autre direct soit arrêté. Il est alors important de bien dimensionner l'architecture selon vos besoins.

[Accueil](#) / [Mes sessions et directs BigBlueButton](#) / Réaliser un direct BigBlueButton

Réaliser un direct BigBlueButton

1 session BigBlueButton en cours trouvé

Voici la liste des sessions BigBlueButton en cours pour lesquelles vous êtes modérateur. Ce module vous permet de réaliser un direct de cette session BigBlueButton (*utile s'il y a plus de 100 utilisateurs*). **N'oubliez pas de ne pas utiliser les salles privées dans ce cas et de mettre fin à la réunion une fois celle-ci terminée.**

Peu de temps après avoir cliqué sur le bouton "Réaliser un direct BigBlueButton" et sélectionnez les options souhaitées, le direct sera disponible pour les utilisateurs sur la page **(🔊) Directs**.

Webinaire de test 01 (15 février 2021 10:25)



Impossible de réaliser un direct
BigBlueButton pour le moment
(toutes les ressources sont occupées)

Le direct n'a pas été publié

Remarque: cette page s'actualise automatiquement toutes les 30 secondes.



Utilisation du système de publication des présentations Web BBB vers Pod

A l'heure actuelle, le système de diffusion de webinaire BBB est totalement couplé avec le système de publication des présentations Web vers Pod.

S'il s'avère que cela ne devait pas être le cas, des modifications seront réalisées pour enlever ce couplage.

Au final, le fichier `settings_local.py`, concernant BBB, va être de ce type :

Configuration settings_local.py / BBB

```
##
# BigBlueButton settings
#
# Use of BigBlueButton
USE_BBB = True
# Directory of bbb-recorder plugin (see documentation https://github.com/jibon57/bbb-recorder)
# bbb-recorder must be installed in this directory, on all encoding servers
# bbb-recorder create a directory 'homedir'/Downloads that needs disk space
DEFAULT_BBB_PLUGIN = '/home/%userpod%/bbb-recorder/'
# Directory that will contain the video files generated by bbb-recorder
DEFAULT_BBB_PATH = '/data/www/%userpod%/bbb-recorder/'
# BigBlueButton or Scalelite server URL, where BBB Web presentation and API are
BBB_SERVER_URL = 'https://bbb.univ.fr/'
# BigBlueButton key or Scalelite LOADBALANCER_SECRET
BBB_SECRET_KEY = 'xxxxxxxxxxxxxxxxxxxxxxxxxxxx'
# Username format in BBB
BBB_USERNAME_FORMAT = 'first_name last_name'
# Type of the generated video by default
DEFAULT_BBB_TYPE_ID = 1
# Number of days before removal the meetings (and associated users) not already published
# To not remove old meetings, set 0 value
BBB_NUMBER_DAYS_BEFORE_DELETE = 0
# Allow the users to perform a BigBlueButton live
USE_BBB_LIVE = True
# Allow users to record their BigBlueButton live
USE_BBB_LIVE_DOWNLOADING = True
# Number maximum of BBB lives simultaneous managed by the system
BBB_NUMBER_MAX_LIVES = 5
```

Récapitulatif des étapes à réaliser pour l'installation

Au final, voici la liste des étapes à réaliser sur les serveurs *bbb-pod-live* :

- Installation de Docker
- Installation de Docker-compose
- Installation de PHP 7 avec php-xml et php-redis
- Création du répertoire bbb-pod-live
- Copie du script bbb-pod-live.php dans ce répertoire
- Paramétrage du script bbb-pod-live.php
- Mise en place du CRON *php f bbb pod-live.php* toutes les minutes
- Vérifier que les ports REDIS (6379, 6380...) sont bien accessibles aux serveurs Web de Pod
- Vérifier que le partage NFS pour les enregistrements est bien accessible depuis les serveurs bbb-pod-live (*attention aux problèmes de droits*)

Au final, voici la liste des étapes à réaliser sur les serveurs *Pod* :

- Suivre la documentation concernant la publication des présentations Web de BigBlueButton <https://www.esup-portail.org/wiki/x/AgCBNg>
- Configuration adéquate dans le settings_local.py
- Mise en place (ou modification) du CRON *python manage.py bbb main* toutes les 2 minutes
- Vérifier que le partage NFS pour les enregistrements est bien accessible depuis les serveurs d'encodage de Pod (*attention aux problèmes de droits*)

Installation annexe :

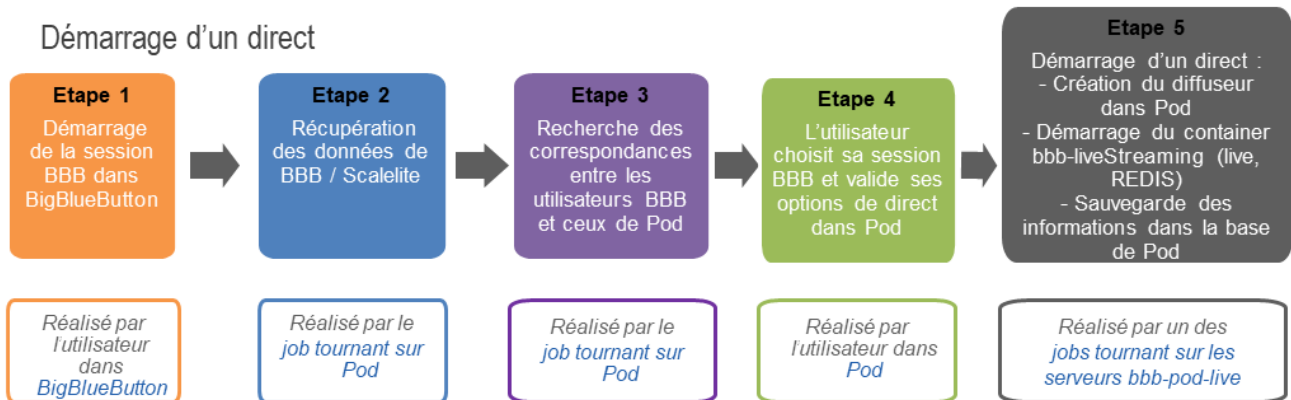
- Installation d'un ou plusieurs serveurs RTMP Nginx (cf. <https://www.esup-portail.org/wiki/x/BgC8KQ>)

Synthèse du processus global

Démarrage d'un direct

DIFFUSION DE WEBINAIRES AVEC COUPLAGE BBB/POD (BBB-POD-LIVE)

Démarrage d'un direct

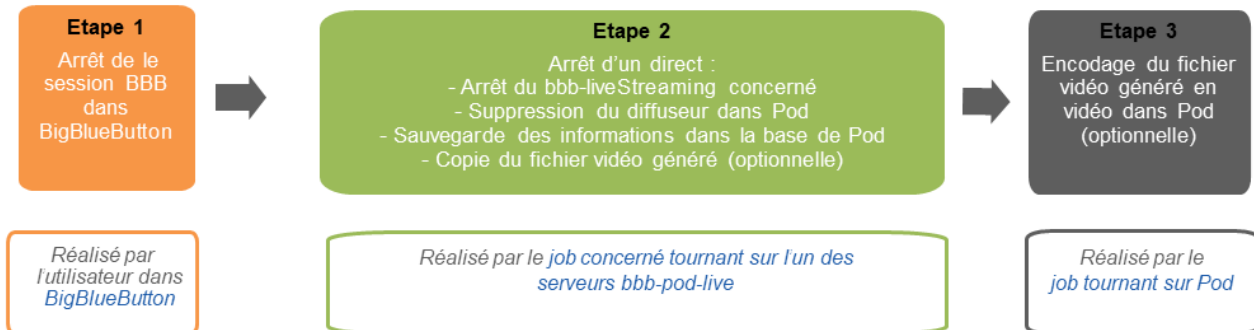


<https://www.esup-portail.org/wiki/x/BgApOQ>

Arrêt d'un direct

DIFFUSION DE WEBINAIRES AVEC COUPLAGE BBB/POD (BBB-POD-LIVE)

Arrêt d'un direct



<https://www.esup-portail.org/wiki/x/BgApOQ>

Exploitation

Dans le cas normal, où tout se passe bien, ce système ne devrait pas engendrer d'exploitation particulière et vous ne recevrez d'emails de la part du système qu'en cas d'erreurs.

Cependant, une interface d'administration a été prévue afin de suivi des directs BigBlueButton mais également pour réaliser l'exploitation en cas d'incidents.

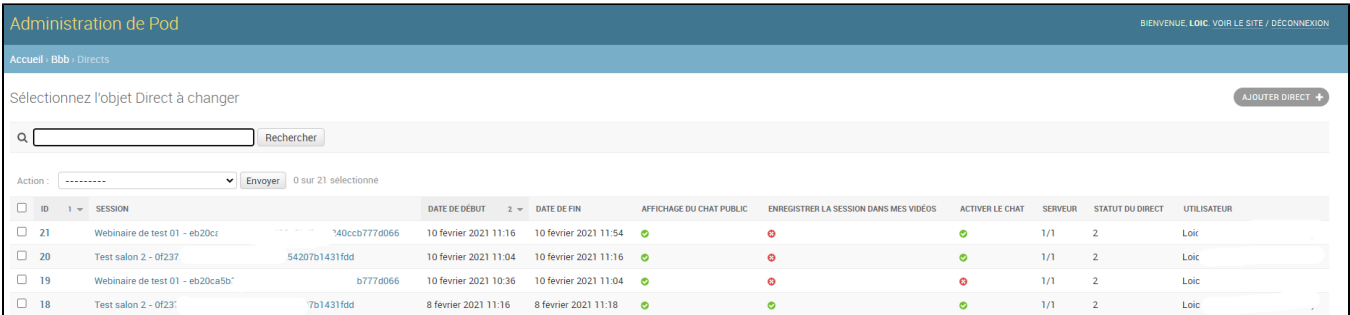
L'interface d'administration

Via l'administration de Pod, vous aurez accès au menu, **BBB**, qui contient - en plus des options **Sessions** et **Participants** (cf. le système de publication) - l'option **Directs**.



Les directs BigBlueButton

Voici l'interface de listing des directs réalisés dans BigBlueButton :



Il est possible de modifier les données si nécessaire.

Modification de Direct

HISTORIQUE

Session :

Webinaire de test 01 - eb20ca5b...

10ccb777d066

Date de debut :

Date :

10/02/2021

Aujourd'hui

Heure :

11:16:49

Maintenant

Start date of the live.

Date de fin :

Date :

10/02/2021

Aujourd'hui

Heure :

11:54:23

Maintenant

End date of the live.

Statut du direct :

2

Statut du direct: 0. non démarré, 1. en cours, 2. arrêté

Serveur :

1/1

Serveur / processus effectuant le direct.

Utilisateur :

Loïc

Nom d'utilisateur / ID d'utilisateur, qui souhaite effectuer le direct.

☐ Accès restreint

Le direct est-il uniquement accessible aux utilisateurs authentifiés ?

Broadcaster :

80

Broadcaster in charge to perform live.

☒ Affichage du chat public

Souhaitez-vous montrer le chat public en direct ?

☐ Enregistrer la session dans Mes vidéos

Souhaitez-vous enregistrer la vidéo de cette session, à la fin du direct, directement dans Mes vidéos ?

☒ Activer le chat

Voulez-vous un chat sur la page de direct pour les étudiants ? Les messages envoyés dans le chat de cette page de direct se retrouveront dans le chat public de BigBlueButton.

Redis hostname :

ts-sun-ontpellier.fr

Redis hostname, useful for chat

Redis port :

6379

Redis port, useful for chat

Redis channel :

chat1

Redis channel, useful for chat

Supprimer

Enregistrer et ajouter un nouveau

Sauvegarder et continuer d'éditer

SAUVEGARDER

Exploitation du script lancé par le job CRON du serveur Pod

Pour la gestion des directs BigBlueButton, ce job CRON permet d'associer un utilisateur de Pod à un utilisateur de BigBlueButton.

Pour voir ce que ce script réalise, il est possible de :

- soit il faut configurer Pod pour être en **mode debug** (cf. `settings_local.py`).
 ⚠ Attention : le mode debug ne doit pas être activé dans un environnement de production.
- soit modifier directement le script `/django_projects/podv2/pod/video/management/commands/bbb.py` pour qu'il affiche directement les traitements réalisés (cf. fonction `print_if_debug`).

Il ne reste plus qu'à exécuter ce script `/django_projects/podv2/pod/video/management/commands/bbb.py` manuellement via la commande `python manage.py bbb main` :

Exécution du script manuellement, compte %userpod%

```
cd /data/www/%userpod%/django_projects/podv2
workon django_pod
python manage.py bbb main
```

En mode debug, le script affiche l'ensemble des traitements réalisés :

```
*** Check BBB/Scalelite actual meetings ***
Request on URL : https://...fr/bigbluebutton/api/getMeetings?checksum... , status : 200

- Meeting : ebd2081d1407b94b
+ Meeting already exists in Pod database.
+ User already exists in Pod database : Tiberiu

- Meeting : cd4a3f06414b30e
+ Meeting already exists in Pod database.
+ User already exists in Pod database : Sylvie

- Meeting : e8abaf671c3678
+ Meeting already exists in Pod database.
+ User already exists in Pod database : Valerie
+ User already exists in Pod database : Valerie

- Meeting : 65e7af18958a41
+ Create the meeting in Pod database. internal_meeting_id : 65e7af18958a41

- Meeting : 96bb4de76512
+ Meeting already exists in Pod database.
+ User already exists in Pod database : Aurelie

- Meeting : 5120478e984e
+ Meeting already exists in Pod database.
+ User already exists in Pod database : Marie-george
+ User already exists in Pod database : Marie-george

- Meeting : 78ba4369be2
+ Meeting already exists in Pod database.
+ User already exists in Pod database : Marina

- Meeting : bcbad06d69c
+ Meeting already exists in Pod database.
+ User already exists in Pod database : Souhila

- Meeting : aee4d55453d
+ Meeting already exists in Pod database.
+ User already exists in Pod database : Bruno
```

Exploitation du script lancé par le job CRON du serveur bbb-pod-live

Le script PHP bbb-pod-live.php écrit dans un fichier de log (1 fichier par jour) dans le répertoire de logs (cf. **PHYSICAL_LOG_ROOT**).

En mode debug :

- le fichier de log (**AAAA-MM-JJ_bbb-pod-live.log**) contient l'ensemble des requêtes réalisées (création de diffuseur, connexion à l'API Pod, démarrage du direct...)
- un autre fichier de log (**id.log**) est créé et contient les logs du container Docker bbb-liveStreaming. Son id correspond au numéro de direct dans la base de Pod.

Exploitation courante/exceptionnelle

Un enseignant ne voit aucune session BBB en cours

Typiquement, si un enseignant lance une session dans BigBlueButton, se connecte à Pod et ne voit aucune session en cours dans la partie "**Réaliser un direct BigBlueButton**" dans les 2 minutes, c'est que le job CRON de Pod n'a pas réussi à trouver une correspondance entre l'utilisateur BBB et l'utilisateur dans Pod.

VIDEO

[Chaînes](#)
[Utilisateurs](#)
[Types](#)
[Directs](#)

Rechercher

Ajouter une vidéo

LB

fr

[Accueil](#) /
 [Mes sessions et directs BigBlueButton](#) /
 Réaliser un direct BigBlueButton

Réaliser un direct BigBlueButton


Aucune session BigBlueButton en cours trouvée

Voici la liste des sessions BigBlueButton en cours pour lesquelles vous êtes modérateur. Ce module vous permet de réaliser un direct de cette session BigBlueButton (utile s'il y a plus de 100 utilisateurs). **N'oubliez pas de ne pas utiliser les salles privées dans ce cas et de mettre fin à la réunion une fois celle-ci terminée.**

Peu de temps après avoir cliqué sur le bouton "Réaliser un direct BigBlueButton" et sélectionnez les options souhaitées, le direct sera disponible pour les utilisateurs sur la page **(🔊) Directs**.

Aucune session BigBlueButton n'est en cours.

Remarque: cette page s'actualise automatiquement toutes les 30 secondes.



UNIVERSITÉ DE MONTPELLIER

[> Contactez nous](#)
[> Legal notice](#)
[> Top of page](#)

[> Help](#)
[> Université de Montpellier](#)

Il est alors possible de modifier directement cette assignation dans l'administration, module **BBB / Participants**.

Administration de Pod

BIENVENUE. LOIC. VOIR LE SITE / DÉCONNEXION

Accueil > Bbb > Participants > loic - MODERATOR

Modification de Participant

HISTORIQUE

Session :

Webinaire de test 01 - eb20ca5' 5777d066

Nom complet :

loic

Nom complet de l'utilisateur dans BBB.

Rôle de l'utilisateur :

MODERATOR

Rôle de l'utilisateur dans BBB.

Username / Identifiant utilisateur :

pC -

Nom d'utilisateur / ID utilisateur, si l'utilisateur BBB correspond à un utilisateur existant de Pod.

Utilisateur :

Loic

Utilisateur dans la base de données de Pod, si l'utilisateur est trouvé.

Supprimer

Enregistrer et ajouter un nouveau

Sauvegarder et continuer d'éditer







SAUVEGARDER

Arrêter un direct BigBlueButton manuellement

Si nécessaire, il est possible d'arrêter un direct BigBlueButton manuellement via la procédure suivante :

- Accéder, dans l'administration, au module **BBB / Directs** puis cliquer sur le direct concerné (*profitez en pour identifier le serveur et le process ayant réalisé le direct*) :

Modification de Direct

Session :	Webinaire de test 01 - et 6	
Date de début :	Date : 15/02/2021 	Aujourd'hui
	Heure : 10:49:48 	Maintenant
	Start date of the live.	
Date de fin :	Date : <input type="text"/>	Aujourd'hui 
	Heure : <input type="text"/>	Maintenant 
	End date of the live.	
Statut du direct :	1	
	Statut du direct: 0. non démarré, 1. en cours, 2. arrêté	
Serveur :	1/1	
	Serveur / processus effectuant le direct.	
Utilisateur :	Loic  	
	Nom d'utilisateur / ID d'utilisateur, qui souhaite effectuer le direct.	
<input type="checkbox"/> Accès restreint	Le direct est-il uniquement accessible aux utilisateurs authentifiés ?	
Diffuseur :	82	
	Broadcaster in charge to perform live.	
<input checked="" type="checkbox"/> Affichage du chat public	Souhaitez-vous montrer le chat public en direct ?	
<input type="checkbox"/> Enregistrer la session dans Mes vidéos	Souhaitez-vous enregistrer la vidéo de cette session, à la fin du direct, directement dans Mes vidéos ?	
<input type="checkbox"/> Activer le chat	Voulez-vous un chat sur la page de direct pour les étudiants ? Les messages envoyés dans le chat de cette page de direct se retrouveront dans le chat public de BigBlueButton.	
Redis hostname :	ts-sun-jmontpellier.fr	
	Redis hostname, useful for chat	

- Positionner une **date de fin** et **mettez le statut du direct à 2 : direct arrêté**.
- Sur le serveur concerné, dans le répertoire bbb-liveStreaming concerné, exécuté la commande : *docker-compose down*.

Exemple de commande d'arrêt

```
# Exemple de commande à exécuter sur le serveur bbb-pod-live n°1, pour le processus 1 (Serveur : 1/1 en base de données Pod).
cd /home/%USER/bbb-pod-live/bbb-live-streaming1
docker-compose down
```