

Diffusion du conseil municipal

Solution de diffusion en direct du conseil municipal de la ville d'Échirolles.

La solution repose sur des caméras tourelles permettant l'intégration des flux vidéo en temps réel, le pilotage à distance depuis OBS et la diffusion multi-plateformes. OBS Studio est utilisé pour la gestion des caméras et la réalisation en direct du conseil.

- [Installation d'OBS Studio sur Zorin OS \(ou Ubuntu\)](#)
- [Matériel et setup](#)
- [Intégration de BigBlueButton \(visioconférence\)](#)
- [Intégration de présentations \(ZowieBox\)](#)

Installation d'OBS Studio sur Zorin OS (ou Ubuntu)

Avant propos

OBS Studio est le logiciel libre utilisé par la ville pour diffuser le conseil municipal, et d'autres manifestations, en direct. Cette procédure décrit l'installation du logiciel et des plugins nécessaires à son fonctionnement, dans l'environnement qui est le nôtre.

Prérequis

→ ZorinOS peut-être téléchargé [en cliquant sur ce lien](#). Choisir la version Core.

Installation d'OBS Studio

1. Cette procédure nécessite l'installation préalable de [Zorin OS Core](#), dans sa dernière version (version 17 au jour de la rédaction).
2. Ajouter le PPA permettant l'installation de la dernière version d'OBS Studio :

```
sudo add-apt-repository -y ppa:obsproject/obs-studio
sudo apt update
```

Installer la dernière version d'OBS Studio :

```
sudo apt install obs-studio
```

Attention : **n'installez pas le paquet obs-plugins**

Installation de NDI

NDI (ou NDI-HX) est un protocole temps réel qui permet de découvrir les caméras compatibles connectées **sur le même réseau**, de récupérer le flux vidéo et de l'intégrer simplement à OBS. L'avantage de NDI, par rapport aux protocoles vidéo standard (RTSP, par exemple) est qu'il permet d'obtenir un flux en temps réel, et donc de ne pas avoir de décalage entre l'audio et la vidéo.

L'installation se déroule en deux temps :

1. Installation du runtime NDI (propriétaire)

La procédure d'installation se trouve [sur cette page](#) (voir section : Required Components - NDI Runtime / Linux)

2. Installation du plugin DistroAV (anciennement OBS-NDI)

1. Télécharger le paquet depuis [cette page Github](#) (à la date de rédaction de cette documentation, il s'agit du paquet **distroav-6.1.1-x86_64-linux-gnu.deb**).
2. L'installer.

Lancer OBS Studio et vérifier que, quand on rajoute une source (clic droit dans le dock « Sources »), le choix « Source NDI™ » est bien disponible dans le menu.

Installation de PTZ Controls

PTZ Controls est un plugin qui permet de piloter les caméras depuis OBS Studio. La page Github du projet [se trouve ici](#).

1. Télécharger la version la plus récente du paquet [depuis cette page](#) (à la date de rédaction de cette documentation, il s'agit du paquet **obs-ptz-0.17.0-x86_64-linux-gnu.deb**).
2. Installer le paquet.

Attention, le point 3 ci-dessous est important : il existe une dépendance mal documentée pour faire fonctionner PTZ Controls. Si vous ne l'installez pas, PTZ Controls n'apparaîtra pas dans le menu « Docks » d'OBS Studio !

3. Conformément à l'alerte ci-dessus, installer ce paquet :

```
sudo apt install libqt6serialport6
```

Lancer OBS Studio. Dans le menu « **Docks** », un nouveau dock « **PTZ Controls** » devrait être présent. Sélectionner le dock et le glisser en bas, entre « Sources » et « Mélangeur audio » par exemple.

→ Vous devriez disposer d'un OBS Studio opérationnel.

Auteur original de cette documentation : Nicolas Vivant.

Matériel et setup

[Échirolles](#) a conçu le streaming de son conseil municipal avec le maximum d'outils libres (et environ 6000 € d'investissement), un thread testé et approuvé en 7 points.

Si ça peut aider...

1. Les caméras.

- Acquisition de 2 caméras-tourelles POE compatibles avec le protocole NDI (modèle : [Avonic CM70-NDI](#)).
- Le POE (Power Over Ethernet) permet de les alimenter et de récupérer les flux vidéos avec un câble unique.

2. Le son.

- Une carte son [Scarlett 2i2 3rd generation](#) : 2 entrées XLR/gros jacks (6.35 mm), une sortie USB-C → USB direction le PC.

3. Le réseau.

- Un switch 8 ports POE, sur lequel le PC et les caméras sont connectés.

4. La machine.

- Un PC sous [Zorin OS](#) Core, avec un processeur récent (Intel Core i5 ou AMD Ryzen équivalent), 16 Go de mémoire, et un disque dur SSD de 500 Go minimum.

5. Le logiciel.

- [OBS Studio](#) avec libndi, DistroAV (anciennement OBS-NDI) et [PTZ Controls](#).
- [PTZ Controls](#) permet de piloter les caméras et de gérer les prépositions directement depuis OBS.

6. Le graphisme.

- Le service communication fournit l'habillage et les polices, choisit le cadrage, etc.

7. La diffusion.

- Un serveur [Peertube](#) pour publier les vidéos du conseil municipal.
- <https://video.echirolles.fr> si vous voulez voir à quoi ressemble le résultat.

Nota :

- cette solution est couplée à BigBlueButton, notre solution libre de visioconférence, une intégration décrite dans [un autre article](#).
- un serveur rtmp interne (via un module nginx) permet de compenser la limitation d'OBS Studio (un seul flux de streaming possible) et de diffusion sur plusieurs plate-formes, une autre solution qu'il serait utile de documenter. Au moins deux plugins pour OBS permettent de faire la même chose dans passer par un serveur dédié : [Aitum Multistream](#) et [Multistream by Streamway](#)

Auteur original de cette documentation : [Nicolas Vivant](#).

Intégration de BigBlueButton (visioconférence)

Avant-propos

Pour les visioconférences, nous avons fait le choix de [BigBlueButton](#). Utilisable depuis un navigateur web, ses fonctionnalités de vote, de gestion de la parole et des présentations semblent adaptées pour une participation à distance en conseil municipal. Pour ce faire, un certain nombre de points sont incontournables :

1. les participantes et participants doivent pouvoir se voir, dans la visioconférence et dans la salle ;
2. ils doivent pouvoir s'entendre et interagir de façon fluide ;
3. une présentation, réalisée en salle ou à distance doit être également visible par toutes et tous ;
4. tout cela doit être accessible dans la diffusion en ligne, en direct et en rediffusion.

Trouver la solution nous a pris un peu de temps, et c'est pourquoi il me semble utile de la partager.

Se voir

Dans OBS, nous disposons déjà des images de la salle. L'utilisation de la fonctionnalité « caméra virtuelle » du logiciel va permettre de transformer ce flux en caméra, et de l'envoyer dans BigBlueButton. Il convient d'activer la fonction, puis d'utiliser son navigateur (Firefox, chez nous) sur la même machine pour rejoindre la visioconférence. Une caméra « OBS » est alors proposée, il suffit de l'activer et le flux vidéo est automatiquement transmis dans la réunion. Le son de la salle, en revanche, sera transmis par un autre moyen. Il ne faut donc pas l'activer.

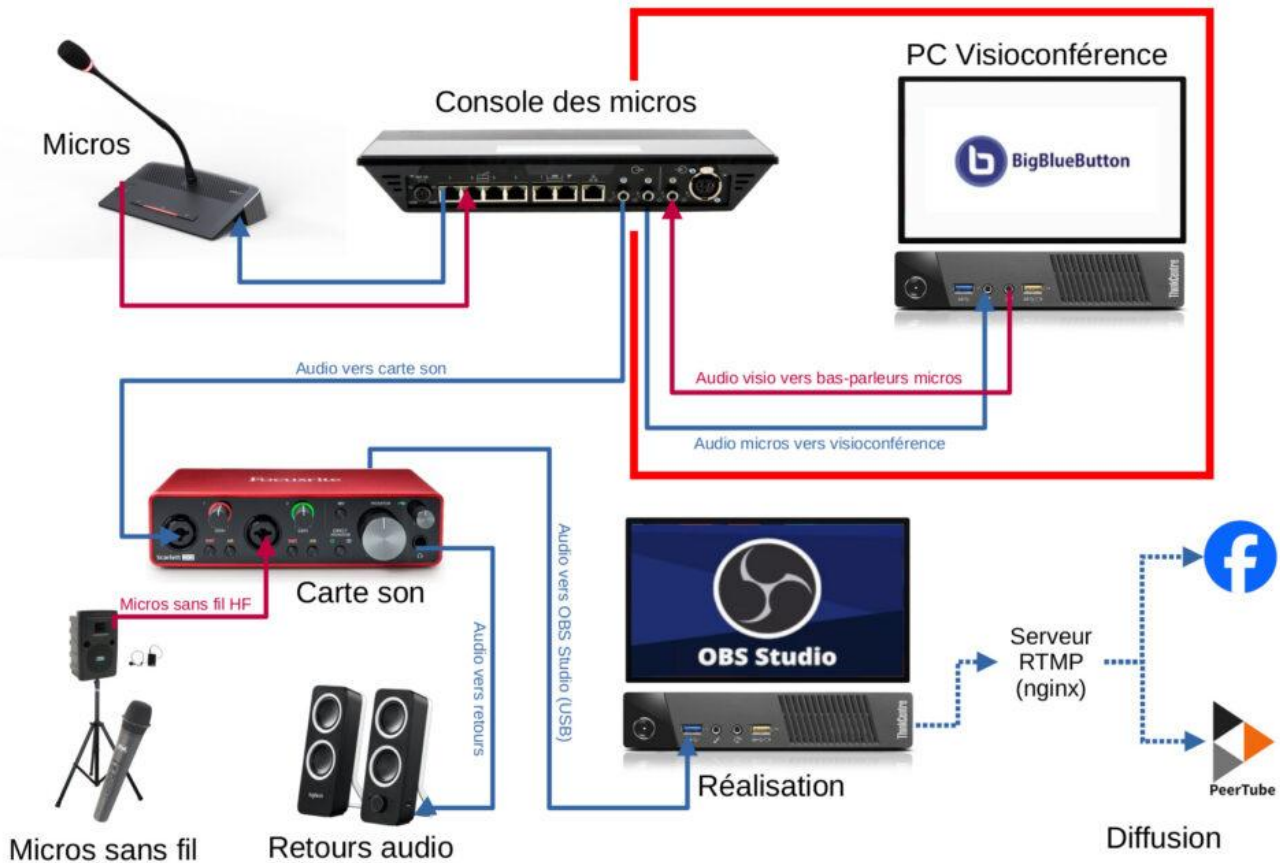
Une capture de la fenêtre de visioconférence dans une scène OBS dédiée permet de diffuser l'image de la visioconférence quand c'est utile. Dans la salle, un PC connecté dans la visioconférence permettra de retransmettre les images sur un écran ou un vidéoprojecteur, toujours sans le son et avec un minimum de délai (moins d'une seconde).

S'entendre

C'est là que les choses se compliquent. Dans notre conseil municipal, les élu·e·s parlent dans des micros et s'entendent dans les bas parleurs de ces mêmes micros. Un système intégré à la console de gestion permet d'éliminer l'écho, de telle sorte qu'un micro activé ne reprend pas le son diffusé

dans les bas parleurs. Dans le cas contraire, un larsen ne manquerait pas de se produire. Le problème est le même pour une intervention dans la salle en visioconférence : l'écho doit absolument être éliminé.

Nous avons donc utilisé la console de gestion des micros pour gérer le son de la visioconférence. Un PC dédié, sous Zorin OS, est connecté dans le salon via le navigateur Firefox. Il reçoit le son des micros depuis la console, et le renvoie dans la console pour que les interventions depuis BigBlueButton soient audibles dans la salle via les bas parleurs. Ce que décrit la partie encadrée en rouge ci-dessous.



Autre initiative intéressante : l'utilisation d'un serveur RTMP qui nous permet de contourner une limitation d'OBS, qui ne peut diffuser qu'un seul flux à la fois (deux, en fait, si on considère la fonction de caméra virtuelle comme un flux à part entière), et de récupérer le flux depuis n'importe quel PC pour de l'affichage en direct sur un ou plusieurs écrans distants. Installé sur notre serveur de visioconférence BigBlueButton, il permet de diffuser sur un nombre indéfini de plate-formes. À Échirolles, nous diffusons sur Facebook, la page de la ville étant plutôt bien suivie, et sur notre serveur Peertube pour une meilleure qualité et une meilleure intégration sur notre site web (Drupal).

Ainsi, les habitant·e·s qui ne disposent pas (ou ne souhaitent pas disposer) d'un compte Facebook peuvent assister au conseil en ligne, sans contrainte, algorithme ou publicité intempestive.

Si vous souhaitez diffuser sur plusieurs plateformes et que la mise en place d'un serveur RTMP vous semble trop lourde, au moins deux plugins pour OBS permettent de faire la même chose dans passer par un serveur RTMP dédié : [Aitum Multistream](#) et [Multistream by Streamway](#).

L'ensemble des vidéos d'Échirolles peuvent être consultées sur notre compte Peertube : <https://video.echirolles.fr>.

Auteur original de cette documentation : [Nicolas Vivant](#).

Intégration de présentations (ZowieBox)

Un-e présentateur-ice veut projeter une présentation, soit depuis le PC à disposition, soit depuis le sien. On veut récupérer les diapos :

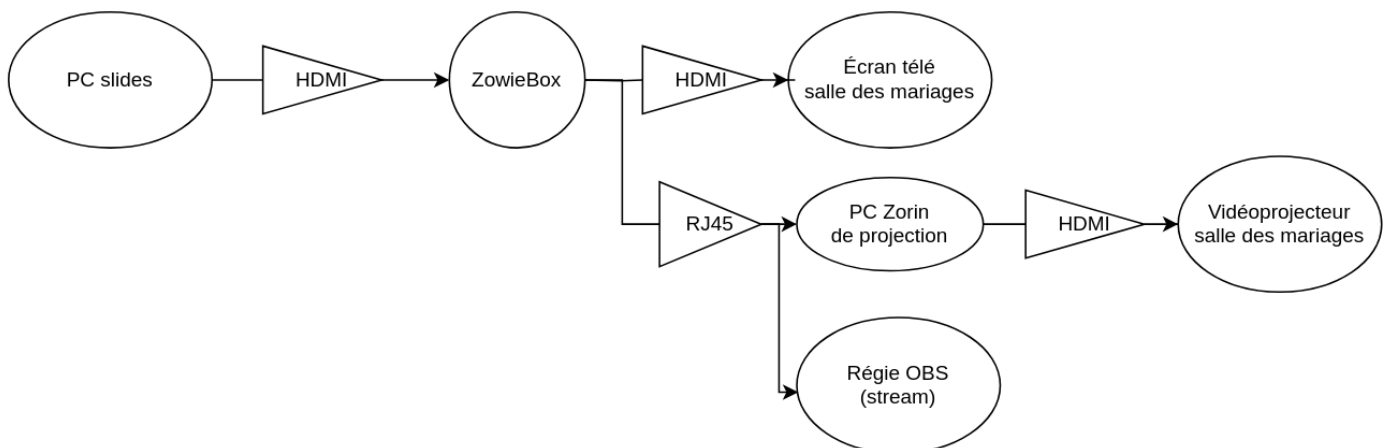
- sur l'écran de projection (écran télé) ;
- sur le vidéoprojecteur (salle des mariages, mais en plus grand) ;
- sur le stream Peertube

La solution

Nous avons choisi d'utiliser une ZowieBox ([voir ici](#)).

Connectée au réseau, la ZowieBox permet, à partir d'une entrée HDMI, d'obtenir plusieurs flux en sortie sur le réseau. Une sortie HDMI permet également de récupérer le signal vidéo en direct.

Schéma



ZowieBox

La ZowieBox prend une IP en DHCP. Elle réagit bien à un débranchement/rebranchement de l'HDMI d'entrée. Elle exporte :

- une sortie pass-through HDMI, qui réagit également bien au débranchement/rebranchement de l'HDMI d'entrée comme de sortie ;

- des flux rtsp, qui fonctionnent bien, mais ont quelques soucis :
 - lorsque le flux est interrompu (veille de l'ordinateur de présentation, HDMI débranché, ...), le flux est **coupé**. Le comportement des lecteurs de flux diffère : conservation de la dernière image, quitter, ... Pas extrêmement pratique. On ne peut pas non plus se connecter au flux tant qu'aucun écran ne projette dessus.
 - après branchement d'un flux, il faut quelques secondes avant que le flux ne revienne. Si on s'y connecte trop vite, il y a une latence assez pénible (quelques 100ms) qui persiste dans le temps, alors que tout se passe beaucoup mieux si on s'y connecte quelques secondes après.
- un flux NDI. Protocole propriétaire, mais qui marche globalement mieux.

Les paramètres du flux d'entrée (résolution + fréquence) sont repris pour l'encodeur à *chaque branchement d'HDMI d'entrée*. Une fréquence à 60Hz donne une mauvaise qualité d'image (plein d'artefacts) ; il faut baisser cette fréquence à 25 ou 30Hz.

PC Zorin de projection

On utilise [obs](#) pour afficher le flux NDI. L'installation n'est pas évidente avec NDI.

- installer obs-studio :

```
sudo apt install obs-studio # version 30.2.3 à cette date
```

- installer libndi **en version 5** : aujourd'hui (2025-02-17), la version 6 n'arrive pas à décoder un flux NDI correctement. Récupérer [libndi-get.sh](#), puis

```
$EDITOR libndi-get.sh # modifier la ligne 4, v5 -> v6
sudo bash ./libndi-get.sh # installer
```

- installer DistroAV aka obs-ndi. Récupérer le `.deb` dans les [releases GitHub](#) (en version 4.13.0 en février 2025), puis

```
sudo apt install ./obs-ndi-*.deb
```

- Lancer OBS. Ajouter une source NDI (appelée eg. « ZowieBox »), sélectionner la bonne source (le PC doit être dans le même VLAN que la zowiebox).
- Mettre la source en plein écran : clic droit sur la source, `Projecteur plein écran (source)` - `> Moniteur ...`

Pendant les présentations

- à chaque débranchement du câble HDMI : remettre en 30Hz l'encodeur de la ZowieBox (à la main...)
- gérer OBS comme pour les diffusions du conseil
- Si ça plante, essayer de :
 - Vérifier tous les branchements
 - Redémarrer OBS et la projection plein écran
 - Redémarrer la ZowieBox (reboot électrique) (1 min de boot environ ?)
 - Redémarrer les ordinateurs

Auteur original de cette documentation : [Philippe Le Brouster](#).