Table of Contents

A guide to live stream musical performances from home

This guide is intended for people wanting to live stream a musical performance from home. It provides informations for most kind of setups, from purely acoustic performances to more complex cases implying a DAW, an external sound card with connected instruments and mics.

It focuses mostly on audio configuration as it seems to be where most people struggle, although it also includes walkthroughs for all the steps from installing a streaming software to actually start streaming.

Prerequisites

- Knowing what you want to play and the involved equipment
- Basic knowledge about your DAW (how to access its settings, add a plugin, manage tracks inputs and outputs)
- The streaming platform you are going to use (Twitch, Facebook, Youtube...) and a link to its documentation
- A computer running **Windows** or **macOS**[1]
- A bit of free time to read this guide carefully

We will use OBS Studio as our streaming software but you are not required to know how to use it.

How to use this guide?

Simply follow the parts in the correct order! All parts can always be reached through the navigation side bar on the left.

Start by part a super fast introduction to OBS (part 1) that will help you to install OBS and understand its basis.

You will then make OBS work with your camera in part 2. Video configuration

Then come the main point of this guide: audio configuration. You will first follow part 3. Which audio setup do I use? which will lead you to the proper instructions in part 4. Audio configuration corresponding to your case.

You will then have to configure OBS to make it stream to your favorite platform (part 5)

There is also a Troubleshooting part that should help you to solve most encountered problems.

Why I wrote this guide?

It began during the COVID-19 lockdown. During this period we saw many initiatives (online festivals, groups, personal projects) aiming to live stream artists and musicians performing from their home. I got somehow involved in some of these projects and saw the lack of information regarding the audio configuration, which can differ A LOT depending on the musical project itself. So I tried to find a solution for most cases and sorting everything properly allowing people to find which case fit their needs and how to do it.

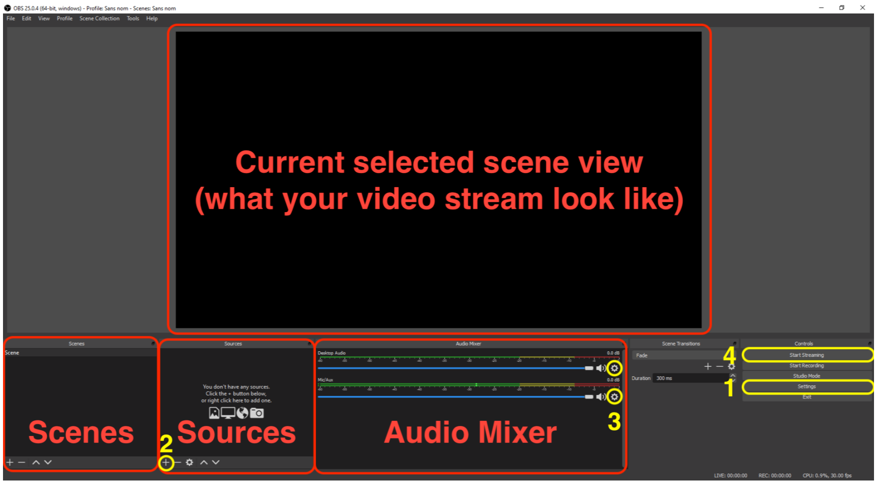
---

[1] Sorry dear Linux user, I have no specific instructions for you, but you might still find this guide useful! Solution logics are the same but you will have to find compatible alternative softwares by yourself. Have a look at Jack Audio or Pulse Audio.

A super fast introduction to OBS

Note: If you are already familiar with OBS, feel free to skip the next two parts and go straight to part 3. Which audio setup du I use?.

1. Start by downloading OBS Studio from here, then install it.
2. Launch the application. The first time, it will prompt an automatic configuration wizard. **Skip it**. A first scene with the name "Scene" should already be created.
3. You should then see the main OBS interface:


Current selected scene view
(what your video stream look like)
Scenes
Sources
Audio Mixer
1
2
3
4

As meant by its own name, OBS is an open broadcasting software. OBS works with **Scenes**, each scene being a patchwork of a variety of **Sources** arranged as you want. Sources can be pieces of hardware (a camera, a microphone...), files from your computer (images, video files...), virtual devices (like internal audio input/outputs) or even external live streams!

You can stream one scene at a time, and switch between scenes simply by clicking on it in the Scenes window. Several scenes can share the same audio/video sources. So you can for example have two scenes each showing a different camera, but both using the same audio source. This way you can switch from a camera to another one simply by selecting another scene.

In this guide we will only cover the simplest possible setup, including only one scene with only one video source and one audio source.

Look at the OBS interface screenshot above. In this guide we will mostly use the four yellow circled buttons:

1. The `Settings` button allows you to open the Settings panel. You can also access it through the menu bar, at `OBS > Preferences` in macOS, or `File > Settings` in Windows
2. The + button at the bottom of the Sources window allow you to create a new source.

3. The gear icon ⚙ nearby each source in the Audio Mixer allow you to access the source's **Properties**, the **Filters** applied to it, and the global **Advanced Audio Properties** panel allowing you some advanced tasks such as monitoring your sources.
4. The `Start Streamingk` button, quite self-explanatory.

Notice that here we have two audio sources in the Audio Mixer ("Desktop Audio" and "Mic/Aux"). These sources does not appear in the Sources window because they are like special sources automatically created by OBS. They however act as any other audio source. You can by the way create additional similar audio sources by clicking the ✚ in the Sources window and select `Audio input capture` or `Audio output capture`. The first one acts like the "Desktop Audio" source while the latter acts like a "Mic/Aux" source.

In the next steps of this guide, you will see how to use theses features to set up a simple yet working stream for your musical performance.

Video configuration

1. Once you opened OBS for the first time, you should see a scene called `Scene` in the Scenes window. You can see it is empty as it has no source attached to it.

2. Now we need to add a video source in our scene. You can use both a webcam or a smartphone as a camera.

    - Using your webcam:

        - Click the + in the Sources window, and choose `Video capture device`
        - Select `Create new`, set whatever name you want and click OK. The properties window of the video capture device appears. Your webcam should then appear in the devices list. If not, refer to your webcam's manual.
        - Depending on your webcam you should have several settings. Let's stay with defaults and click OK. If you need to re-open the properties windows, just double-click the created video capture device in the Sources window.
    - Using a smartphone's camera.

        - You have several solutions to use your phone's camera as a webcam in OBS, such as EpocCam (for iOS and Android, free version with low quality, paid version for full quality), DroidCam (Android only, free) or NDI HX Camera (iOS only, free, you will also need the OBS-NDI plugin to make it work)). The steps to get these apps working in OBS are quite simple and can be easily found on developers websites or on youtube. Notice that most of these apps require your phone to be Wifi connected to your home network, so you cannot use it as a 4G internet hotspot at the same time.
        - Once you added your phone as a video source in OBS, you might see it also appear as an audio source in the OBS audio mixer with the "video capture device" name you set earlier. You most like want to disable this audio source as you will get your sound from a proper mic or your audio software as we will see.
3. Once you created your video source, make sure it fill the whole area of your scene by stretching it to the correct dimension. To stretch it, select the source in the Sources window, then a red frame should appear around your view. Click a corner and stretch.

4. Good, you can now go to the next part to fid the best audio setup that fits your situation.

3. Which audio setup do I use?

First you need to determine the kind of audio setup you are using:

- My performance is purely acoustic and/or I just need one mic to capture it
- My performance requires an audio software but no external sound card
- My performance requires an audio software and an external sound card

This will lead you to the appropriate instructions to prepare the audio part of your live stream.

My performance is purely acoustic and/or I just need one mic to capture it

If you don't necessary need a computer for your performance, you might consider streaming from a smartphone as it is very simple to set up. Just download the app corresponding to your streaming platform, do what they tell you to do and you should be good. If you need to stream to a custom RTMP server, you can use Larix Broadcaster because it's free, simple to use, and available for iOS and Android. If you don't have a proper microphone, many phones these days still have a decent quality, sometime even in stereo, so why not!

You may prefer to use a computer if you don't have a smartphone, or if you use a webcam with it's built-in mic, a USB microphone, a mini-jack microphone or an XLR microphone you nee to plug into your external audio interface. We assume here that you do not need a DAW. Choose the case corresponding to your situation:

- If you use the **built-in mic of your computer** or your **webcam**, the **build-in mic input** of your computer or an **USB microphone**, refer to Case 1: OBS + Mic/Aux source.
- If you have to **plug your microphone into your external audio interface**:
    - **macOS user**: refer to Case 1: OBS + Mic/Aux source.
    - **Windows user**:
        - External sound card with **MME/DirectX drivers**: Case 2: OBS + Desktop Audio Source
        - External sound card with **ASIO drivers**: Case 4: OBS + OBS-ASIO (Windows only)

My performance requires an audio software but no external sound card

Choose the case corresponding to your situation:

- **macOS users**: refer to Case 6: OBS + BlackHole (macOS only).
- **Windows users**:
    - If you use default **MME/DirectX drivers**, refer to Case 2: OBS + Desktop Audio Source
    - If you use **ASIO drivers** (such as ASIO4ALL, FL Studio ASIO installed along with FL Studio but compatible with any audio software, or ReaRoute with Reaper), you have two solutions:
        - The **ReaStream VST plugin** solution might be the simplest one as you keep your audio software configured as usual with your favorite ASIO drivers. Refer to Case 3: OBS + ReaStream VST (Windows only).
        - You can also choose to keep your audio in an ASIO context using an additional **ASIO mixer** such as VoiceMeeter Banana, Asio Link Pro or Jack Audio (all free). This is the most tricky way to go and can be hard to configure properly to get less latency than with MME/DirectX drivers or ReaStream. Because latency is why you use ASIO drivers I guess. You can refer to Case 5: OBS + OBS-ASIO + ASIO mixer (Windows only) if you want to give it a try, and go back to Case 3 if you are unsuccessful.

My performance requires an audio software and an external sound card

This might be the case for most of you. The idea is quite simple: we want to turn your main audio software output into an audio input in OBS. You can do that virtually, using your external sound card software (only for compatible models/drivers) or a 3rd party virtual audio device, or you can do it physically, by wiring one of your external sound card's output to a physical audio input. Here are some details for all of these ways, each leading you to the **appropriate audio configuration**.

A virtual loopback might be preferred as it does not need extra piece of hardware, but depending on your case a hard-wired loopback can eventually be easier to achieve. Choose the option you feel the most comfortable with.

Virtual loopback

- Your sound card supports multi-initialization from multiple applications, and has a built-in **loopback feature**. It is the case of all 3rd gen Focusrite Scarlett, 1st gen 8i6 Focusrite Scarlett, most RME interfaces (using TotalMix), some Motu interfaces (using CueMix), the Roland Quad Capture, the Behringuer XR18, and some other devices I'm not aware of. This should be the best way to go as it is quite simple to configure and should add little to no additional latency. The loopback feature (generally accessible from your external sound card's mixer software) will allow you to route your audio output back into one of its (either virtual or physical) audio inputs. You can then easily select this input in OBS. First refer to your external sound cards owner manual to see how to use the loopack feature. Then, depending on your OS:
    - **macOS user**: refer to Case 1: OBS + Mic/Aux source.
    - **Windows users**: refer to Case 4: OBS + OBS-ASIO (Windows only) if your external sound card uses ASIO drivers, or Case 1: OBS + Mic/Aux source otherwise.
- Whichever the external sound card you use, you can still create a virtual loopback using **additional softwares** (but it can increase your audio latency):
    - **macOS users**: refer to Case 6: OBS + BlackHole (macOS only)
    - **Windows users**:
        - The **ReaStream VST** plugin solution might be the simplest one as it should work for any case and it allow you to keep your audio software configured as usual with your favorite sound card drivers (ASIO or other). Refer to Case 3: OBS + ReaStream VST (Windows only).

        - If your external sound card uses **MME/Direct drivers**, you should be able to catch its output directly by selecting it as a "Desktop Audio" source in OBS. Refer to Case 2: OBS + Desktop Audio Source
        - If your external sound card uses **ASIO drivers**, you can try to use an additional ASIO mixer such as VoiceMeeter Banana, Asio Link Pro or Jack Audio (all free). This is the most tricky way to go and can be hard to configure properly to get less latency than with MME/DirectX drivers or ReaStream. Complexity goes even higher if you need your external sound card to output/input audio to/from external audio effects, synths, mics involved in your performance... You can refer to Case 5: OBS + OBS-ASIO + ASIO mixer (Windows only) if you want to give it a try, and go back to Case 3 if you are unsuccessful.

Physical (hard-wired) loopback

- Your sound card does not provide a built-in loopback feature, but you have **two unused audio outputs** (or one stereo output) and possibly two unused audio inputs on your audio interface (can be an SPDIF in and out -- SPDIF is better for quality as the signal stays digital -- or two balanced jack/XLR outputs and two line inputs). If so, you can "manually" create a loopback with real cables from real life, by plugging your available output into your available input. The input can either be one of your external sound cards inputs, your computer built-in line mini-jack input (although not advised for quality reasons) or even a secondary external sound card (see next point). Notice that using an input from your only external sound card should only work if this sound card supports multiple initialization from multiple applications. As this feature is very rarely made explicit by sound card manufacturers, just try and see if it works. To proceed, in your audio software, find a way to double the output of your master (using a bus or return track, or an "External audio effect" in Ableton Live) so it can be sent to both your usual master/monitoring audio output, and the extra audio output available on your sound card. Then wire the selected secondary output back into an available input. Then:
    - For **macOS users**, refer to Case 1: OBS + Mic/Aux source
    - For **Windows users**:
        - If you **loop your audio back into your computer built-in line-input**, refer to Case 1: OBS + Mic/Aux source
        - If you use an **available input on your external sound card**, refer to Case 4: OBS + OBS-ASIO (Windows only) if your external sound card uses ASIO drivers, or Case 1: OBS + Mic/Aux source otherwise
- You have **two external sound cards**, so you can use one with your audio software and the other one as an input for OBS. Your case is then similar to the previous one. You need to wire one of your main sound card output to the secondary audio interface input, then select this input in OBS. To do so:

  - For macOS users, refer to Case 1: OBS + Mic/Aux source
  - For Windows users: refer to Case 4: OBS + OBS-ASIO (Windows only) if your secondary external sound card uses ASIO drivers, or Case 1: OBS + Mic/Aux source otherwise

4. Audio configuration

Please read carefully 3. Which audio setup do I use? to know which of the following case(s) will match your setup.

- Case 1: OBS + Mic/Aux source
- Case 2: OBS + Desktop Audio Source
- Case 3: OBS + ReaStream VST (Windows only)
- Case 4: OBS + OBS-ASIO (Windows only)
- Case 5: OBS + OBS-ASIO + ASIO mixer (Windows only)
- Case 6: OBS + BlackHole (macOS only)

Case 1: OBS + Mic/Aux Source

You are into Case 1 if:

> - You just want to you a mic to capture you performance
>   - You use a USB mic, a built-in mic, the webcam's built-in mic or the line input of your computer.
>   - Your mic is connected to an external sound card
>   - You use macOS
>   - You use Windows and the MME/DirextX drivers for your sound card
> - You use macOS with an audio software and
>   - An external sound card with multiple initialization from multiple applications and a built-in loop-back feature
>   - An external sound card compatible with multiple initialization from multiple application, and enough available outputs and inputs
>   - Two external sound cards
> - You use Windows with an audio software and
>   - a secondary external sound card with MME/DirectX drivers as a looback input, or
>   - an external sound card with ASIO drivers and you loop your audio back into your built-in computer line-input.

If your whole audio come from a microphone, it may be already automatically selected as Mic/Aux source in the Audio Mixer.

If not, you should be able to select it or any appropriate (non ASIO) audio input into the OBS Settings panel, "Audio" tab, "Mic/auxiliary Audio" device. drop-down list.

You can also click the gear ⚙ near-by the Mic/Aux source in the Audio Mixer, then click Properties and select your correct or audio input in the "Device" drop-down list.

You should then see the Mix/Aux VU-meter reacting to the captured audio. If so your audio is properly configured, good! You are now ready for part 5. Streaming configuration.

Case 2: OBS + Desktop Audio source

You are into Case 2 if:

> You use Windows with an audio software and MME/DirectX drivers, with or without external sound card.

It is as simple as case 1 but with a little variation.

1. Open your audio software, make it play some sounds, and remember the audio output it's using
2. Open OBS and go to its Preferences (Settings panel), then into the Audio tab.
3. In the Devices / Desktop Audio drop-down list select the same output as your audio software.
4. Click OK then in the OBS audio mixer you should see the VU-meter of the "Desktop Audio" source moving relatively to the sound your audio software is outputting. Sometimes it seems a big buggy and you need to click the gear ⚙ of the Desktop Audio source, click Properties, then re-select the same output you just selected in the previous step. You can also try restarting OBS (don't worry, it will automatically save your session).
5. Awesome, you are now ready to go to part 5. Streaming configuration!

Case 3: OBS + ReaStream VST (Windows only)

You are into Case 3 if:

> You use Windows with an audio software and ASIO drivers, with or without external sound card.

This case work for any kind of ASIO driver, regardless the use of an external sound card or not. It can actually work also with any other kind of driver (MME/DirectX/WASAPI) but you should rather refer to Case 2: OBS + Desktop Audio Source if you use one of those.

The following instructions are based on ReaStream, a VST plugin made to send audio and midi in real time over a local network. Here we will use it on a same comuter as a bridge between your DAW and OBS.

1. ReaStream is included in a suite of free plugins called ReaPlugs VST FX Suite you can download here. Choose the 32bit or 64bit version depending on your audio software (Ableton Live 10 is 64bit only, other might depend).
2. Run the installer. In the "Choose Components" screen, you only need ReaStream (stereo) but you can select other plugins if you want to try them.
3. In the "Destination Folder", make sure you are using one of the following (see list below), otherwise the VST wont be available in OBS. Click Install and you're done. If you usually use a custom folder for your VSTs, just re-install ReaStream to your custom folder once you got it installed in one recognized by OBS:
   - C:/Program Files/Steinberg/VstPlugins/
   - C:/Program Files/Common Files/Steinberg/Shared Components/
   - C:/Program Files/Common Files/VST2
   - C:/Program Files/Common Files/VSTPlugins/
   - C:/Program Files/VSTPlugins/
4. Open your audio software and add the **reastream-standalone** VST plugin on your master track. You should see a firewall alert saying that the plugin want to access your network. Just allow it on both private and public networks, just in case, and click OK. Now on the plug-ins interface, select "Send audio/MIDI" then choose/type `\*local broadcast` (without quotation marks) in the IP field.
5. Open OBS, click the + in the Sources window and select "Audio **input** capture". Give it any name you want, and click OK.
6. As Device, choose an audio input you are NOT using. We actually don't care of the audio input itself, as we want the audio coming from the VST we will add in the next steps. So just use an unused audio input. If you can only

select some used inputs (such as your default microphone), then is is still OK, we will figure it out later. Click OK.

7. Now right click on your newly created audio input capture source, and select "Filters".
8. If you had no unused audio input available, you want to turn down the volume of your used audio input without turning down the volume of the OBS audio source itself. To do so, add two "Gain" Filters and set them to -30dB.
9. Ad a new audio filter by clicking the +, and select "VST 2.x Plug-in". Give it the name you want and click OK.
10. In the VST 2.x Plug-in drop-down list, select "reastream-standalone", then click "Open Plug-in Interface". Windows should prompt you again the firewall security message. Just allow everything one more time.
11. Make sure the Identifier is the same as the one you set in Ableton ("default" by default), and you're done.
12. Now if you play some sound in your audio software you should get it into your OBS audio input capture device. If so, you're good to go to part 5. Streaming configuration. If not, see below for some troubleshooting.

What to do if ReaStream in OBS does not receive sound from my audio software?

- Make sure you are using the same sample rate (44.1Khz or 48KHz) in your audio software and OBS (Settings > Audio tab)
- Right click the Audio input capture, open "Properties" then select another input device. For some reasons, it might not work with some devices (especially the "Default" one).
- In ReaStream in your audio software, replace `* local broadcast` by `127.0.0.1` . It is a special IP address saying "this very own computer".
- Try changing to another identifier in both OBS and your audio software. Sometimes, just clicking in the Identifier field and hitting "Enter" can re-instanciate the plugin an make it work.

What to do you it seem like there is some latency between the received audio in OBS and the captured video? Just check the related troubleshooting part of this guide.

Case 4: OBS + OBS-ASIO (Windows only)

You are in case 4 if:

> You uses Windows with ASIO drivers, whether using or not an audio software and/or an external sound card.

1. Check the version of OBS you are using: in OBS, click "Help" in the menu bar, then "About. The version number is just below the "OBS Studio" label.
2. Close OBS, then download OBS-ASIO. Here are the direct download link depending on your OBS Studio version: 25.x.x and above (preferred), 24.x.x or 23.x.x.
3. Then install the plugin.
4. Re-open OBS.
5. Click the + in the Sources window, and select "ASIO" to create a new ASIO source. Give it the name you want and click OK.
6. In the appearing window, choose your ASIO device:
    - If you use an external sound card with a built-in loopback feature or if you physically hard-wired an audio input of your external sound car back into one of its inputs, then choose this external sound cards driver
    - If you use two external sound cards, choose the secondary sound cards driver (the one in which you looped back your primary external sound cards output)
    - If you created a virtual ASIO input device using VoiceMeeter Banana, Asio Link Pro or Jack Audio, ReaRoute, choose the corresponding ASIO driver.
7. Choose "Stereo" in the Format field
8. In OBS Channels 1 and 2 (stands for left and right channels), select the appropriate ASIO inputs you looped your audio back into.
9. Click OK then in the OBS Audio Mixer you should see the VU-meter of the created ASIO source moving relatively to the sound your audio software is outputting. If so, you are good to go to part 5. Streaming configuration!

Case 5: OBS + OBS-ASIO + ASIO mixer (Windows only)

You are in Case 5 if:

> You are desperate.

This is by far the hardest case of all, but also the most documented online. I honestly could not make one of the following solutions work properly. It always resulted in unplayable latency or audio crackles. I think it should work as expected (with very few extra latency) if you use an external sound card as an output only (your performance does not require external microphone or instruments), as you can see in some tutorials on Youtube, but otherwise it just did not worked for me. As I consider ReaStream (see Case 3) as a way more effective solution for now, I will simply list different solutions I found and I will let you look by yourself for some proper tutorials online.

- VoiceMeeter Banana (free) is maybe the most user friendly and most documented solution.
- Asio Link Pro (free) should be the most powerful and flexible solution, bu also the hardest to set up (thanks to a 20 years old GUI). For the record, it's an old piece of software which you had to pay for. The developer passed away a few years ago so nobody could buy it anymore. In 2019 his nephew decided to create a "legit" crack allowing anyone to use Asio Link Pro for free.
- Jack Audio should be actually simpler to configure even though it does not provide a GUI (graphical user interface). All is done with command lines, but you need very few to get your setup working properly.

If all these solutions seem too hard to get through, just go with Case 3.

Case 6: OBS + BlackHole (macOS only)

You are in Case 6 if:

> You use macOS and an audio software, and maybe an external sound card with no built-in loopback feature.

BlackHole is the new "virtual audio device for macOS" replacement for SoundFlower. It works the same but is likely more stable and still in development. If you still want SoundFlower, look here, then instructions are the same than for BlackHole.

1. Download and install BlackHole following official instructions (easy)
2. Then open "Audio Midi Setup" app located in Applications/Utility. Once opened, click "Window" in the menu bar, then "Show audio devices".
3. Click the + in the bottom left, then "Add a multi-output device"
4. Click on the multi-output device you just created, then, depending on your case:
    - If you uses no external sound card, tick the "Use" box for **each** of the following devices:
        - "BlackHole" (will be sent to OBS)
        - The built-in output you will hear you back through.
    - If you uses an external sound card, select it in the "Master device" (or "Main device") field at the top of the window, then tick the "Use" box for the following devices:
        - "BlackHole" (will be sent to OBS)
        - The desired device you will hear you back through (should be your external sound card)
        - Even if you don't use it, tick your computer built-in output (normally not necessary but it's a workaround to face some issues, as stated here).
5. In your audio software, select the Multi-output device you just created as your audio output.
6. In OBS, open the Preferences/Settings, panel, go to the Audio tab, and select BlackHole as your Desktop Audio device. Click OK.
7. You should then see the corresponding VU-meter in the Audio mixer reacting to the audio sent by your audio software. You're done! Ready for part 5. Streaming configuration.

5. Streaming configuration

Testing your internet connection

First of all, you need to know how good is your internet connection, to know if it is suitable or not for video live streaming. You have to make sure it is **fast** enough **stable** enough.

To know if it is fast enough, you can test it using Speedtest on the computer or smartphone you will use for your stream. Mind that for the best results, you should close all your opened websites, disconnect all your devices but the one you will test the connection with, and use a wired (LAN/ethernet) connection when possible.

Here we're interested in your **upload speed**, so the amount of data your device can send to the internet every seconds. In many cases it is much slower than the download speed (and people generally just remember their download speed) so it's important to test it.

In order to live stream audio an video at a decent quality, the bare minimum required upload speed is about **2Mbps**. It can still work below but in low quality and with possible drop outs. Also, be careful: a fast connection does NOT mean a stable connection.

If your connection sometimes randomly drops off, or get super slow for a few minutes or this kind of problem, that would mean that your internet connection may not be stable enough for a live stream and you should consider pre-record your performance, and make the resulting video streamed from another computer or video platform, which is out of the scope of this guide. I found now way of testing stability easily other than your personal experience, so it's really up to you.

WTF is a bitrate and how do I choose one ?

The other critical part here is the **video bitrate** of your stream. A bitrate is the amount of data a stream needs for each second. The higher the bitrate is, the better the stream's quality is supposed to be, but the higher your internet connection speed need to be. If you dont want to go further in explanation, just remember to use a video bitrate of **850Kbps**. It's quite low in term of quality but it should be okay for most internet connection.

Now let's get a bit more technical. Remind your upload speed you just tested. If your internet connection upload speed is 2Mbps, you should be able to stream with a maximum bitrate of 2Mbps (including both video and sound). But for security reasons, and because both an internet connection and a video bitrate are never perfectly stable, we will limit our bitrate to the half of your upload speed. So if your measured upload speed was 2Mbps, your total stream bitrate should not exceed 1Mbs including both video and audio.

As we want to keep the best possible audio quality, we will stay with a 160Kbps audio bitrate. The rest can be used for the video. So in our example of a maximum 1Mbps (=1000Kbps) stream bitrate, we have 1000-160=840Kbps available for the video bitrate. about the same 850Kbps value I gave earlier.

Let's write a global formula to calculate a safe video bitrate:

$$\text{Video bitrate} = \frac{\text{Measured upload speed}}{2} - \text{audio bitrate}$$

with "audio bitrate" = 160Kbps for a better audio quality.

Please don't use a video bitrate above 4000Kbps even if your internet connection can handle it! There will be no real benefits in term of quality for a video stream of 1280x720 at 30 fps, so a higher bitrate basically means waisted bandwidth and energy. You can have further precisions about which bitrate to choose on this page.

Configuring OBS

1. Open OBS, then go to Settings panel, and set the configuration as follow:

   i. In the Stream tab,:

      - Service: select the streaming platform of your choice.
      - Other settings: please refer to your platforms documentation or some online tutorials. Generally speaking, the streaming platform will give you a unique link you have to copy in OBS, as well as entering your platforms credentials.

   ii. Output tab:

      - Video bitrate: set the calculated bitrate as seen above. If you don't know what to use, set it to 850Kpbs.
      - Encoder: if available, choose "Hardware" (might need an Nvidia graphic card). Otherwise choose "Software (x264)"
      - Audio Bitrate: 160 (we always want the best quality here)

   iii. Video tab

      - Output (Scaled) Resolution: **1280x720** (OK for most people)
      - Downscale filter: Bicubic
      - Common FPS Value: 30fps

2. Hit OK to save your preferences.

3. Check one last time in the audio mixer if your audio is working, then start streaming!

4. To monitor how your stream is going, refer to the bottom bar, showing the number of dropped frames and a colored square indicating the connections status. Your amount of dropped frames should be near 0%. If it is not the

case, then you might have some internet connection issues or your computer is too slow. See the Troubleshooting part for some advices.

Troubleshooting

This page consists in a list of advices to solve most problems, split in three categories:

- Troubleshooting OBS dropping frames
- Troubleshooting audio and video not in sync in OBS
- Troubleshooting audio crackles and latency issue

Troubleshooting OBS dropping frames

- If you're using an audio software and that your sound is NOT crackling, then your dropped frames should come from an internet connection issue. You want to improve your internet connection and/or lower the bitrate of your stream:
    - Prefer a LAN (wired) connection rather than WIFI when using a computer
    - Lower the video bitrate of your stream
- If you hear crackling while dropping frames, then your computer might not be powerful enough. You want to lower the CPU usage:
    - Close/kill all ongoing useless applications/process during the performance (web browser, Steam...).
    - In OBS Settings panel, go in the Video tab and set the Downscale Filter to `Bilinear`
    - In OBS Settings panel, go in the Output tab, tick "Enable Advanced Encoder Settings" and choose a faster "Encoder preset". Depending on your computer hardware, presets can be named from slower to faster or from quality to performance. Both mean the same: choosing a faster or more performant encoding preset will result in a lower video quality but in more available CPU resources
    - You want to make OBS and your audio software use less CPU (see Troubleshooting audio crackles and latency issue).
- See this page for more informations

Troubleshooting audio and video not in sync in OBS

After a few minutes of streaming, you might notice a slight **delay between the video and the audio** in OBS. I noticed that sometimes, even if you see a delay between your VU-meter or monitored audio and your camera in OBS, your stream is still correctly synchronized. To be sure of that, just connect to your own stream using VLC or a web browser.

If you are experiencing a true delay between your video and audio in OBS, there are a few things you can do. Latency can be frequent especially if you use your smartphone as an external webcam for OBS. It can also happen many other cases for many different reasons.

- Be sure that you are using the same audio sample rate (48KHz or 44.1KHz) in every involved software/hardware (external sound card, audio software, OBS...)
- Click the gear ⚙ corresponding to your audio source in the OBS Audio Mixer, and select Properties. If there is a box labeled "Use device timestamp", try to check or uncheck it and click OK.
- If your audio is always in advance compared to your video, you can retard it by clicking a gear in the OBS Audio Mixer, select "Advanced Audio Properties", and increase the value of "Sync offset" corresponding to your audio source. A value of 1500ms means a retard of 1.5 seconds is applied to your audio.
- If your video is always in advance compared to your audio, you can retard it by following these steps:
    - Right click your camera source in the OBS Sources window, and select "Filters". Click the + under "Audio/video filters" and select "Video delay (async)".
    - Set the desired delay (a value of 1500ms means a retard of 1.5 seconds is applied to your video) and click OK.

Troubleshooting audio crackles and latency issue

Here we are talking about the latency between you playing and what you hear back. We are not talking about latency in OBS.

- Don't be afraid to restart your computer once you installed something new. Restarting OBS and/or your audio software can also help.
- Mind to **use the same sample rate** everywhere in your audio software, external sound card and OBS settings. We will prefer 48KHz but 44.1KHz is fine too (and a bit lighter for your CPU).
- You can improve the latency of your audio software by lowering your buffer size. A 128 samples buffer size will lead to two times less latency than a 256 samples buffer size. However it is more CPU intensive, so you might experience crackles.
- Make your audio project as light as possible: disable unused inputs and outputs, delete unused tracks, VSTs and plugins, export to audio (or "freeze" in Ableton jargon) the tracks you don't play with during you performance...

- To improve CPU performance (allow more CPU resource to your audio software), you can go into OBS settings panel,