Introduction to video and live-streaming

Welcome to our introduction to video-recording and live-streaming hockey games. This series contains a lot of learnings from the three years of live-streaming the Ottawa Lady Sens U22AA team and is put together as a resource for other teams to use.

We are all in this together. The better the quality and consistency of our video productions, the more visibility we will create for our players and elite female hockey.

For many years prior to live-streaming, I was involved in recording hockey games with a very simple set up (iPhone and tripod) and this worked well to capture the play for off-line analysis and review. Stepping up to add a live-stream challenged me to re-think the solution and over several years I have honed the set-up I will describe here. To be clear, there are many different solutions that can be used, with many different cameras and equipment, but this relatively simple set up has served us very well. Please experiment and take your feed to the next level, but I hope to create a foundation from where you can start and grow from.

Live-streaming games and creating an on-line audience is a little different to simply recording the game. Firstly, we need to consider things from the viewer's perspective and how they will enjoy the stream…

- What platform to stream on?
- Can they see the action clearly?
- Is the stream smooth and continuous or choppy?
- Can the numbers / name-bars be clearly read on the screen?
- Do any overlay graphics add value without obscuring or reducing the size of the main video feed?

From the viewer's perspective, watchability is more important than resolution every time. It is far better to deliver a smooth and consistent stream in 720p or even 480p, rather than delivering a choppy feed at 1080p. Given the very significant reductions in bandwidth required to support the lower resolution streams, it can often be a key choice for the streamer, based upon available network conditions, to deliver a watchable stream. We will talk more about this in the codecs and bandwidth section.

At some points, the discussions may get quite detailed and technical, but I hope to present this in a way that you can duplicate without having to delve too far into the technology. I hope you find this useful and interesting.

If you have any comments or questions, feel free to email me at kevan.jonesrus@gmail.com and I will see what I can do to help.

Table of contents

Camera and equipment

Equipment and network

The video solution that I use grew gradually from an initial recording-only set-up of simply an iPhone and a tripod. This is a great place to start and can be achieved at minimal additional cost. Modern iPhones (and Android phones) have great cameras in them and they work very well in the rink for hockey game recording. I have tried several different camera solutions from GoPros, DLSRs and consumer level video cameras, and whilst these can all work well, the simplicity and quality of the recording from a modern phone is difficult to beat. Moving up to a pro-sumer or pro level video camera can certainly enhance the clarity of the recording, but in many situations this does not necessarily translate into a better or clearer live-stream. Therefore careful consideration of budget and portability often leads us back to using a phone as the video source and this guide will concentrate on that option.

Live-streaming requires a real-time video feed into the network to your hosting platform of choice (YouTube, Twitch, etc.) and as such requires efficient video compression to achieve reasonable bandwidths with good picture quality. Having more compute power allows a better picture to be achieved in the same bandwidth via video compression and coding. As such, streaming directly from a phone or iPad may not be able to deliver a high quality stream in low bandwidth due to the processing power available on the phone - although this is improving all the time. Using an external computer (Mac or PC) allows us to provide significant coding and optimise the picture quality for a limited data rate. This is the solution that we use, with an Apple MacBook Air. The MacBook Air is chosen as a light-weight, lower power consumption computer that provides good processing power and can also be used for video editing after the stream.

So the minimum set of equipment for a live-stream and recording set up includes…

- Camera (iPhone in our solution)
- Tripod with "video" head and phone holder mount
- Computer (MacBook Air in our solution)
- Cable to connect camera to computer

One of the advantages of using a Mac and an iPhone is that these can simply be connected together with a cable (usually Lightning to Thunderbolt/USB-C) to create a USB video connection. This avoids the need to have a video capture card / adapter, which is needed for some cameras with HDMI outputs. Some video cameras also support USB streaming directly, but check with your device to see what is supported.

For our solution we add a few extra capabilities to enhance the overall output…

- Second Camera (another iPhone) to provide a clock overlay on a monopod stand
- Third Camera (another iPhone) to provide a backup recording - in case everything else fails!
- Portable power supply battery with 120V inverter
    - This allows uninterrupted portable power without having to run cables around the rink
- USB hub and battery pack to connect and power the additional cameras
- External keypad for hotkey shortcuts

Finally, in order to be able to deliver a live-stream, you need a good network connection. Depending upon your circumstances, this may be available via Wi-Fi or Ethernet cable in your locations, but for

flexibility on the road we travel with an Internet Hotspot. This is a small device (a little bigger than a hockey puck) that connects to the cell-phone network and produces a local Wi-Fi network to connect to. We use a Rogers Mi-Fi device and in most places this provides an excellent connection.

Video and technique

Filming hockey games presents us with a few choices on how to capture the game. It is relatively simple to use a wide angle lens as this gives a whole zone in a single shot and makes pointing the camera really easy.

This may be useful in some coaching reviews, but this does not deliver the most watchable feed for your viewers. If you are using a phone as a camera and want to stay with a fixed zoom level during the game, then a simple rule is to adjust the zoom to get the net on one side of the screen and the edge of the blue line as it meets the boards on the other (assuming you are shooting from the middle of the rink). This will make it easier to see the players more closely but also manages the amount of panning to follow the play to a reasonable level.

It is desirable to zoom in a little bit more than this, but I recommend working up to that, as it takes more concentration to keep the play in frame.

When the players pass through the neutral zone, they will naturally be closer to the camera, so the operator needs to pan to track the players and the puck across the ice. Overall this provides a much more watchable feed than the super wide angle "doorbell" camera!

If you are using a more advanced camera with a zoom capability, then you may wish to zoom out a little for the neutral zone and then back in again as the players enter the zone around the net. If you watch professional coverage on TSN or Sportsnet, you will see the camera operators doing this.

When setting up the camera before the game, check the white balance (if your app allows it) and your exposure level. Adjust these so that the colours look correct and the exposure is right such that the ice is not too "blown out" white or the picture is not too dark.

Connecting it all up

Firstly let's look at the simple configuration. This really is just as simple as connecting the iPhone to the Macbook using a Lightning to Thunderbolt cable. The camera app that I use on the phone (described in more detail later) allows streaming of video over USB and this gives a very simple solution. I believe that if you use a Windows PC rather than a Mac, then this can still be done, but not with the simple driver-less solution of Apple.
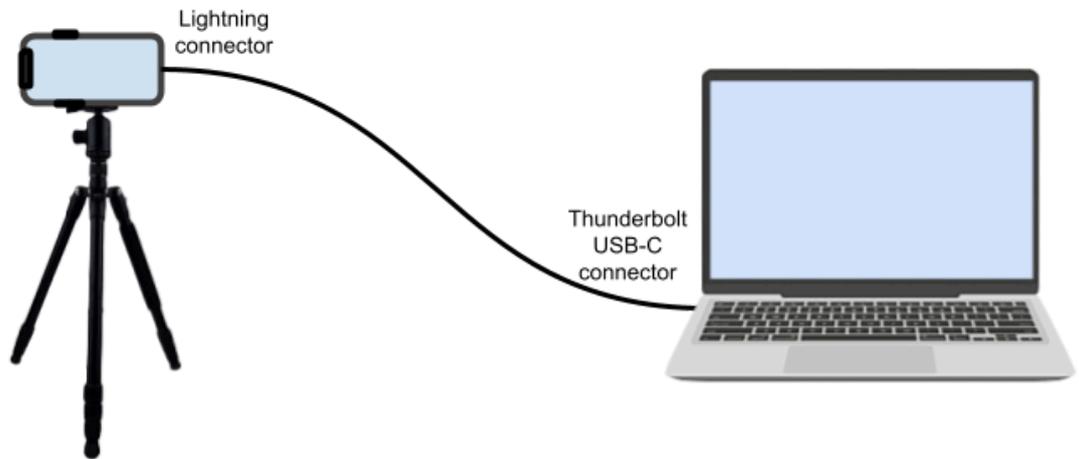
Whilst the simple set-up will get the job done, I also include details of my full set-up below. This adds the extra camera for the scoreboard, a keyboard for enhanced control and a portable power supply so that we are not worried about running out of power during the game.
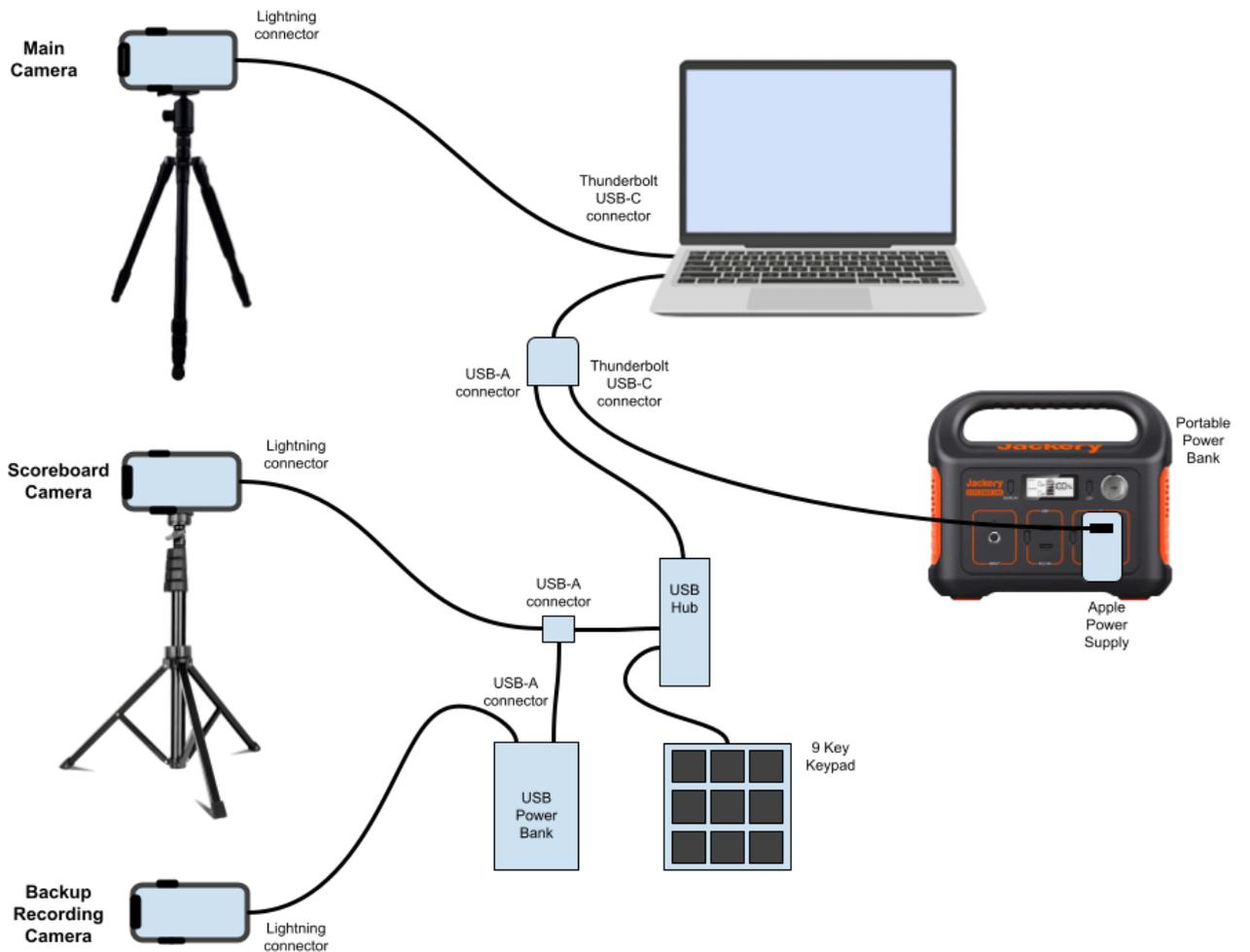

This looks complicated, but the functions of it break down quite easily.

The Apple USB/HDMI adapter provides a means to add power from the Macbook power supply that is connected to the Power Bank and still provide an additional USB connection.

In order to be able to add more than one USB device to this adapter, we need a USB Hub to create multiple ports. This is connected to the adapter via a USB female-male extension cable, to reach to where the Mac is set up. The 9 Key external keypad is connected to this hub to provide keyboard shortcut control of the software on the Mac. Since the USB hub is a passive (un-powered) design, we have used a USB power adapter cable to provide a data connection to the USB hub and also a connection to a USB Power Bank. This allows this Power Bank to provide power to the Scoreboard camera.

Finally, the backup recording camera also shares an output on the USB Power Bank to provide a power source (these are mostly older phones with tired internal batteries!). This camera is not connected in to the rest of the system, but is simply a local recording device, in case all the other equipment fails! I've only ever needed this backup a couple of times in 3 years…but I was glad to have it!

Here are a couple of pictures of the actual set up…
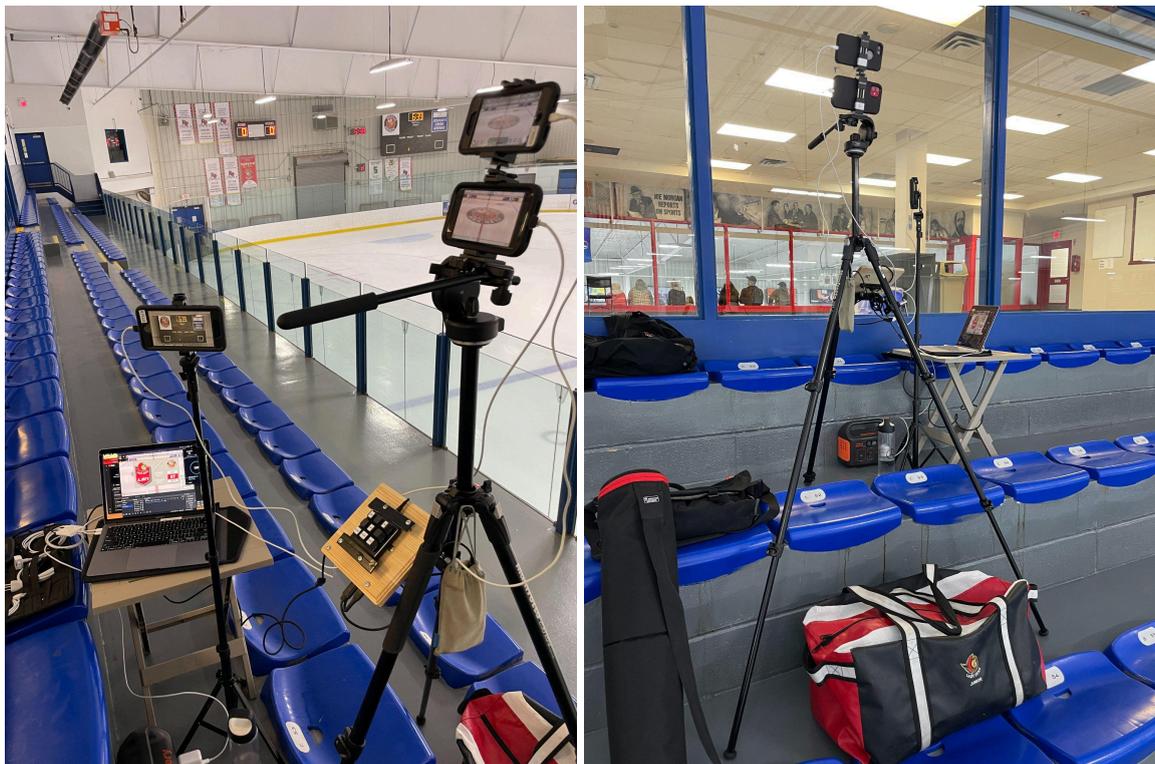
Network connection

Whilst we can still record without any network signal, to be able to stream we need a good connection to the internet. The key requirement for good streaming is your Upload speed, rather than the Download speed (which is the one we usually care more about). To maintain a clean HD (1080p) feed we need faster than (approximately) 7Mbps, but we can still deliver 720p in less than 2Mbps. If the network is very restricted then we can deliver an SD (480p) feed in about 1Mbps.

Depending upon your location there are several different options to get a network signal.
- In building wi-fi network, such as that provided by the rink
- Cellphone network hot-spot wi-fi
  - This can be from a dedicated hotspot device such as our Rogers MiFi device, but can also be simply from a hotspot connection to a phone sharing it's network connection.
- Wired Ethernet connection
  - Depending upon on your set-up, you may also have access to a wired Ethernet connection. This can provide a reliable link, but does require an additional adapter for the Mac to provide an Ethernet to USB conversion.

Whenever setting up to live-stream, one of the first things to do is to test the speed of the connection that is available. There are several different ways to do this, but my preferred application is Ookla's Speed Test app. This gives a set of performance measurements on the network connection in about 30 seconds to 1 minute and an example of it's output is shown below.
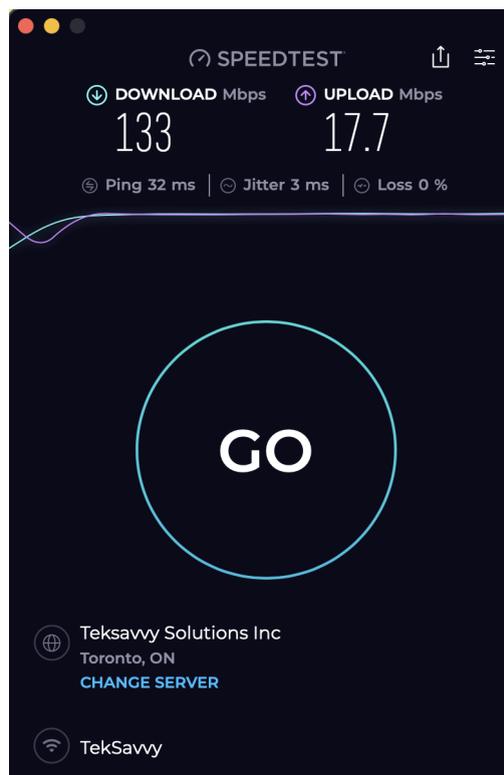
So the upload number on the top right is the important value (17.7 Mbps above). If you are using a dedicated hotspot, this may involve moving to different points in the rink to get a better signal. It's possible to site the hotspot up to about 50m away from the Mac and still maintain a good signal, provided you have a line of sight to the MiFi (no walls in the way). This flexibility allows for searching

out the best cell connection location for the hotspot, and still having the video set up in the best location to record. I often tape the hotspot to the wall where I find a good signal.

Having established a good network signal, we can use the bandwidth achieved to determine the optimum streaming settings. When we configure the streaming data rate, we should always allow for some variation in the signal speed. For example, if we measure an upload rate of 8 Mbps, we should be able to stream at about 5 Mbps without frame loss. A good rule of thumb is that we should not set the streaming rate any higher than (Upload Rate / 1.5). So if we have an upload rate of 8 Mbps, we can set up to 5.3 Mbps streaming rate. If we have an upload rate of 6 Mbps, then we should set the streaming rate no higher than 4 Mbps. This allows for some variation in both the streaming data rate (it naturally varies) and in the network connection. Here is a table, based on experience, of reasonable settings to get you started…

Network Upload Speed	Streaming Rate	Streaming resolution
>10 Mbps	6.5 Mbps	Full HD 1920x1080 (1080p) 60fps
8 to 10 Mbps	5 Mbps	Full HD 1920x1080 (1080p) 60fps
6 to 8 Mbps	4 to 5 Mbps	Full HD 1920x1080 (1080p) 60fps
3 to 6 Mbps	2 to 4 Mbps	HD 1280x720 (720p) 60fps
<3 Mbps	1 to 2 Mbps	SD 640x480 (480p) 60fps

In some locations you may have a very good upload speed available, so where should you set your streaming rate if you have a strong signal such as >20Mbps? Using the rule above you could set a streaming rate of 13Mbps. I would recommend against this for several reasons…

- You won't see a significant quality improvement in the stream - experience says that beyond about 7Mbps the clarity gains are small, unless you using a higher level video camera or streaming higher resolution such as 4k.
- A higher rate is more susceptible to frame loss if the upload rate sees a variation due to network interference or loading.
- A high rate simply uses more data - if you are paying for your data via a hotspot, this will increase your bill for very small gains in viewer experience.

Streaming platform

When we first started live-streaming we used the Twitch platform. This gave a decent service, but was much more focussed around gaming and social interaction. For the last few years we have been using YouTube. Whilst a little more effort to set up well, YouTube provides an excellent and flexible service. If you are streaming from a computer then you simply need a verified account to do so. If you are streaming directly from a mobile device then you need a minimum of 50 subscribers to get started.

YouTube provides an excellent set of tools to manage your streams and on-demand videos. One of the other nice aspects of streaming is that the video is already loaded on to your video site to allow users to watch on demand later. If the network conditions are challenging and we have chosen to stream a non-Full HD feed, then I usually upload the Full HD version that we recorded locally after the game, so that we keep only Full HD content on the site for on-demand access.

YouTube allows pre-scheduling of the streams which allows your viewers to see when you are going to be broadcasting next and they can configure alerts such that they are reminded to watch the feed. People who subscribe to your channel can also allow notifications to see when you go live and this all helps to build a loyal and interested audience.

YouTube allows you to configure your stream to allow DVR like capability. This is really useful for viewers who may want to wind back to review a goal or play, and then catch up to real-time again during the flood.

With YouTube you can also specify to automatically add a livestream to a playlist after the recording is finished. This is great as the automation can group all your games together to make them easier to find on your site later for on-demand playback.

Finally YouTube, like Twitch, allows the viewers to engage with the broadcast via Chat. Unlike Twitch, YouTube allows you to disable this feature completely if you prefer. Chat can be a good way to engage with your audience, but my recommendation is that if you are a one-person video team then you should disable this. We have experienced trolls posting inappropriate content during games on the chat and if you are also trying to stream, this can be very distracting. If you have a multi-person set up with a "producer", then they can monitor and moderate the chat effectively. YouTube provides tools to help moderation.

Streaming opens up a complex world of copyright management. Many hockey games will play music in the rink between the plays, however as a streamer you do not have the rights to re-broadcast that music. Most of the time in YouTube this will result in the copyright being identified and will stop any monetization of the feed. Furthermore, it may block the feed being visible in some regions of the world due to the local copyright. When streaming on Twitch this could create copyright strikes that would block the feed in real time - this is very frustrating. Whilst I have not seen this happen with YouTube, copyright strikes are possible and with three strikes the channel will be blocked.

To avoid this, we mute the music between the plays to ensure we do not get any copyright violations. This is one of the reasons we have the extra keypad (mounted on the tripod), to allow the microphone to be muted & unmuted during the game, without having to reach to the computer.

Having a commentator for the game can really improve the overall viewer experience. If you have someone skilled enough to video and commentate - or even better, a different person to just commentate - this improves the feed. I did not manage to be able to do this and so use rink sound only.

Camera software

Streaming camera software

In order to be able to use the simple direct USB video streaming from the iPhone to the Mac, we need to have a camera application running on the phone that presents a clean video feed over USB that can be used by the Mac. The application that we use is "Shoot Pro Webcam & Telestrator" from SquaresTV software. This is a great little app that provides some manual video controls and provides a clean USB video feed. The app produces a display on the phone that simply shows what the camera is seeing, no overlays, no controls. To get controls, simply tap anywhere on the screen and the control menu pops up. Configure the camera and then after a few seconds the menu fades away leaving the clean feed again. This is ideal for our purposes, as it lets you set the camera up before the game and then have a clean feed the entire time. Example screens are shown below…
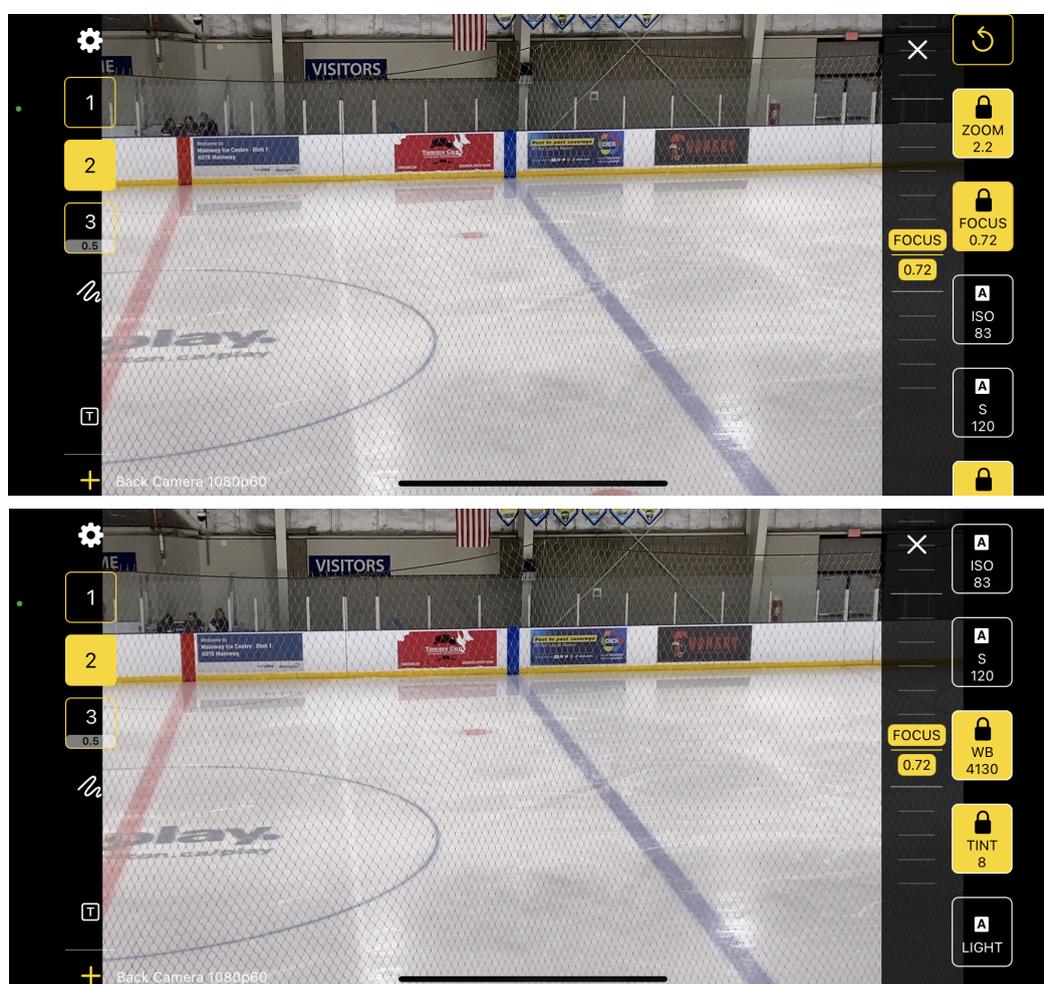
For each game I usually configure the white balance (WB) and tint to get the colours correct and then set the zoom for the game. I also lock these so they are not adjusted during the game. This can be particularly important for daytime games for rinks with windows, where the combination of natural light and electric light can cause some interesting shifts in white balance if set to automatically compensate.
Sometimes it is useful to manually set the focus too. This can be useful when shooting through mesh at the rinks as you can adjust the (fixed) focus point beyond the mesh on the players and the ice.

Because of the small sensor in the iPhone camera, there is a large depth of field which reduces the focus criticality of the set up, but a tweak here can sometimes increase the video sharpness. When there is no mesh (and even sometimes when there is) I usually leave this to autofocus.

Video recording software

We not only stream the video, we also record it direct to disk on the computer, so that we have a high resolution copy for the archives. For the streaming and recording to disk we use the open source software [OBS (Open Broadcaster Software)](#) which we will describe in much more detail in the following sections.

If you choose to run an independent backup camera in case everything fails, there are a number of options to choose as the video recording app on the phone. If using an iPhone like us, then the built in Apple Camera app is a good option, but if more control over the settings is desired then you can look to apps such as [Filmic Pro](#). This used to be a one time purchase app, but has now transitioned to a subscription model. There are other apps available for both iOS and Android phones.

OBS primer

[OBS (Open Broadcaster Software)](#) is an amazing and free piece of open source software, created and supported by the community for people to use. It is the de-facto standard tool for most live-streaming set-ups. It is available on all major platforms (Mac, Windows and Linux) so it is very flexible on which system and hardware you prefer to use. Since this is processing a lot of information in the form of the video, don't expect this run without challenges on low powered computers, however it runs very well on all modern Macs and most mid to high range PCs. If the PC has a graphics chip set from NVidia that supports the NVENC codec in hardware, this can significantly help in processing. On the Apple computers (at least the Apple Silicon ones) Apple provides hardware based codecs that keep the load off the processor too.

Scenes and Sources in OBS

To get started in OBS, we first need to configure the software to work with our camera of choice. With the iPhone and the Apple Mac, this is particularly easy. You simply connect the phone to the Mac with a USB-lightning cable and then configure a new video source in OBS.
In the "Sources" panel, select the "+" icon and then choose "Video Capture Device".
Give the new source a name, such as "iPhone 11" and click "OK".
Select the device from the list and set the preset resolution. This gives options for different resolutions right up to 4k (3840x2160). Even though we are running the camera at 1080p in the Shoot Pro app, My testing has shown that 4k selection here seems to give a slightly clearer picture. That said, 1920x1080 resolution is still fine for most applications and reduces the processing needs a little too.

If you are using a clock camera as well, add a second video source with a different name using the same process as above. Since we will only be using this for a small overlay, I usually reduce the resolution to minimise the processor workload. 1280x720 for both the Shoot Pro app and the OBS video source will work fine.

In the Sources panel there will now be a list of the cameras that you have configured. This list is like a stack of layers in Photoshop. Sources that are at the top of the list will appear on top of the other sources, so in our example we would always have the clock camera listed above the main camera so that we can overlay the clock. Again for each source there are also two icons…an eye icon and a padlock icon. The eye icon toggles whether the source is visible or not and the padlock icon allows the video size and position on the Canvas to be adjusted. When the padlock is closed, the video cannot be moved or resized, when the padlock is open this can be selected in the list and then resized on the Canvas. This is how we adjust the size and location of the clock camera to be where we want in the video window (canvas).

Within your scene panel you can also have graphic overlay files. These are added as "Images" in a similar way that the video source was added earlier. These will be discussed in more detail in a following section.

Codecs and bandwidth

One of the key things about live-streaming and recording video, is to understand the resolution and compression that is applied to the video signal. Uncompressed video requires massive amounts of bandwidth and so to make it usable with a reasonable network connection, we need to compress the signal for broadcast / recording. When we compress a video, some information is lost and so it is important to choose the best codec (compressor-decompressor) and its configuration to give the best picture quality for a given bandwidth. However, using a computer rather than a smartphone for this compression means that we have more processing power available, as well as dedicated hardware based solutions, to give us the best quality compression for a particular data rate.

When we started live-streaming with the Macbook Air, the M1 processor was new and the OBS support for the processor and embedded HW codecs was not great. Initially we selected the x264 software based codec. This is a well regarded codec and gives good compression to H.264 video standard (this is the most common HD compression standard in the world). The downside of x264 is that it is all done on the CPU and ultimately this limits the amount of data that can be handled. This resulted in us using a 1080p 30 fps (frames per second) feed. The H.264 feed was directly compatible with YouTube and so made for a good solution.

Recently the support for the M1 processor in OBS has been much improved and this now allows us better access to the hardware codecs embedded in Apple's processor. We have now switched to using a newer video standard H.265 (also known as HEVC) as this provides greater compression at the same picture quality…or better picture quality at the same data rate. YouTube has also recently added extensions to their streaming interface (still in Beta at this time) to allow ingestion of H.265 video streams. This allows for a seamless transition to streaming in HEVC. The biggest advantage of this solution is that we can deploy the dedicated hardware codec in the Apple processor and this means that we are not limited by the processing power of the CPU. This allows us to compress video at 1080p 60fps (twice the frame rate) while actually consuming less CPU. Using 60 fps allows for smoother motion on the video.

Another important point for the configuration of the codec is that for streaming applications, we are ideally presenting a constant bit rate to the YouTube video ingestion. To do this we select the CBR option (constant bit rate) as this will try to compress for a constant data rate stream. In reality it will still move up and down a bit, but this does tend to smooth the peaks out.
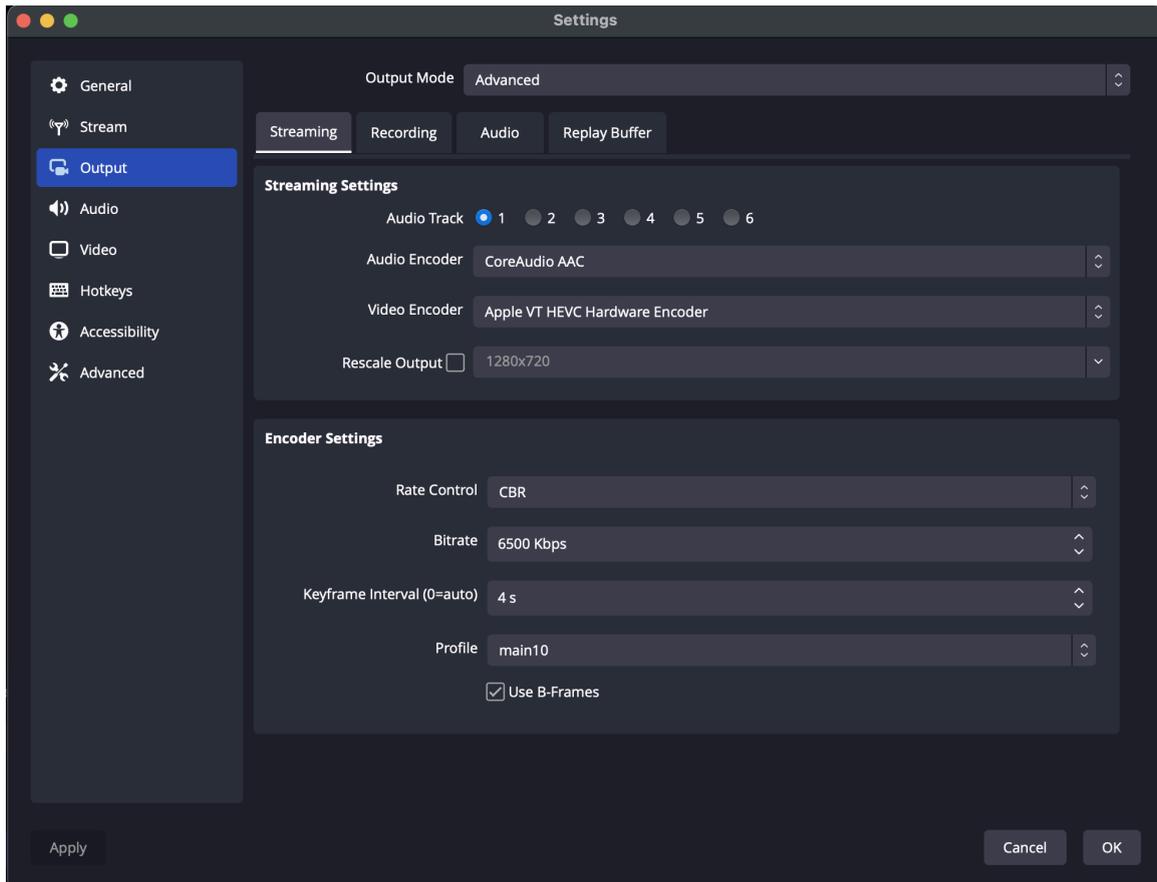
The Bitrate is adjusted based upon the available upload speed for the live-stream as discussed earlier.

The recording of the stream that is done locally to disk allows for higher bandwidth (larger files - better video quality) and so we select an H.264 hardware codec (maximum compatibility) for that, with with an adaptive bit rate (ABR) at 18Mbps for a good balance between picture quality and file size.

Settings

Output Mode **Advanced**

| Streaming | Recording | Audio | Replay Buffer |

Type **Standard**

Recording Settings

Recording Path `/Users/ottawajrladysens/Movies` **Browse**

☐ Generate File Name without Space

Recording Format **MPEG-4 (.mp4)**

Video Encoder **Apple VT H264 Hardware Encoder**

Audio Encoder **CoreAudio AAC**

Audio Track ☑ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5 ☐ 6

Rescale Output ☐ `1920x1080`

Custom Muxer Settings

Automatic File Splitting ☐ `Split by Time`

Warning: Recordings saved to MP4/MOV will be unrecoverable if the file cannot be finalised (e.g. as a result of BSODs, power losses, etc.). If you want to record multiple audio tracks consider using MKV and remux the recording to MP4/MOV after it is finished (File → Remux Recordings)

Apply   Cancel   OK

Settings

Output Mode **Advanced**

| Streaming | Recording | Audio | Replay Buffer |

Type **Standard**

Automatic File Splitting ☐ Split by Time

Warning: Recordings saved to MP4/MOV will be unrecoverable if the file cannot be finalised (e.g. as a result of BSODs, power losses, etc.). If you want to record multiple audio tracks consider using MKV and remux the recording to MP4/MOV after it is finished (File → Remux Recordings)

Encoder Settings

Rate Control **ABR**

Bitrate **18000 Kbps**

☐ Limit bitrate

Keyframe Interval (0=auto) **4 s**

Profile **high**

☑ Use B-Frames

Apply   Cancel   OK

Adding overlay graphics

One of the key advantages that OBS provides is a simple ability to add overlay graphics on top of the stream. This can be used to describe the teams playing in the game, upcoming events, as well as adding stream logos and other icons. The simplest solution for adding overlays is to create image files that are 1920x1080 resolution. This is also the base(canvas) HD resolution of the OBS set up and simplifies alignment with the display. Create the graphics with a transparent background and save them in a PNG format. This can be done in many different applications including Photoshop, Google Drawings and even Apple Preview.

Once the image files are created they should be stored locally on the streaming computer, ready to load into OBS. I separate these files into two groups…one which is used on every stream (e.g. LSN logos, mute/unmute, etc.) and one which has the gameday specific overlays. The ones on every stream are managed as "Helper icons" and gameday ones are handled individually. These gameday overlays are stored in a folder with the data and team name of the opponent for easy location.

Before every game we need to point OBS to use the correct graphic files. This is done file by file…
In the Sources panel, double click on the "Start graphic"
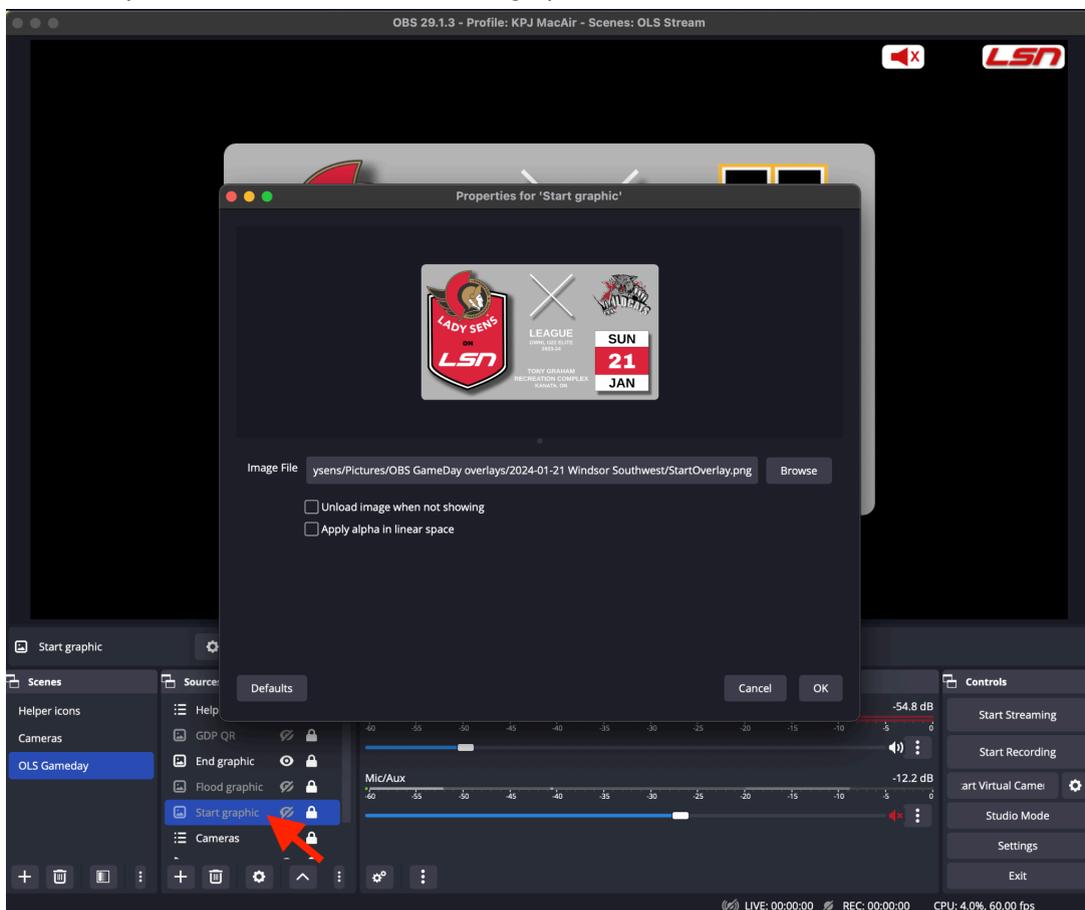
And then click on the Browse button as shown below…
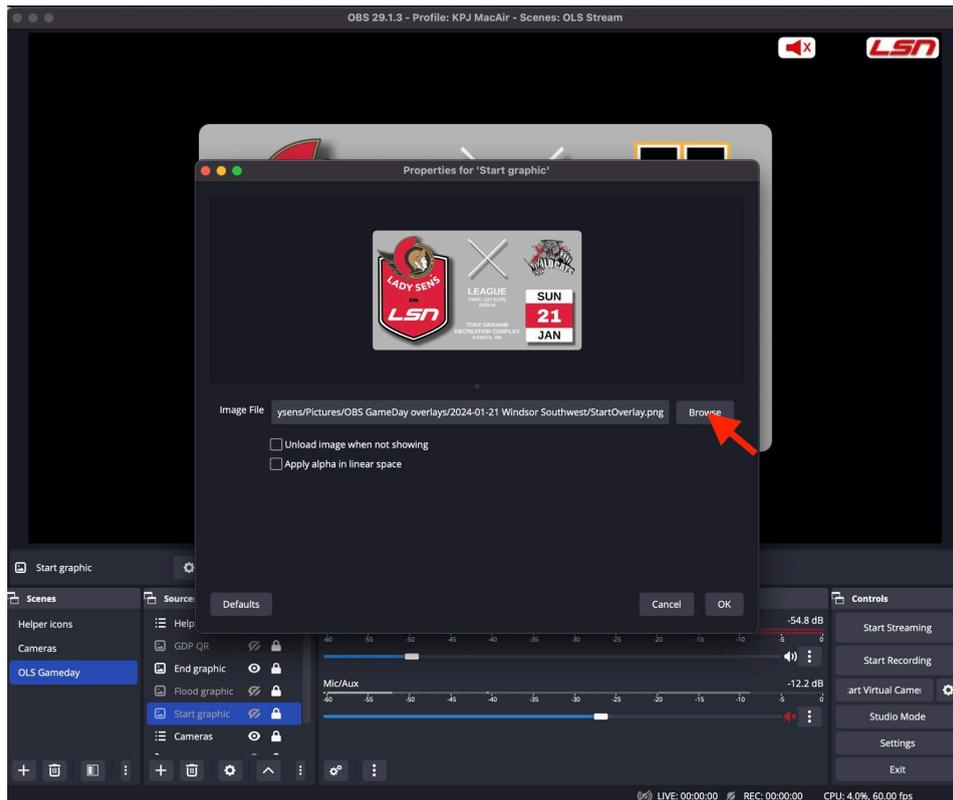
Navigate to the new overlay graphic file that you want to use, click on it to select it and then press "open".
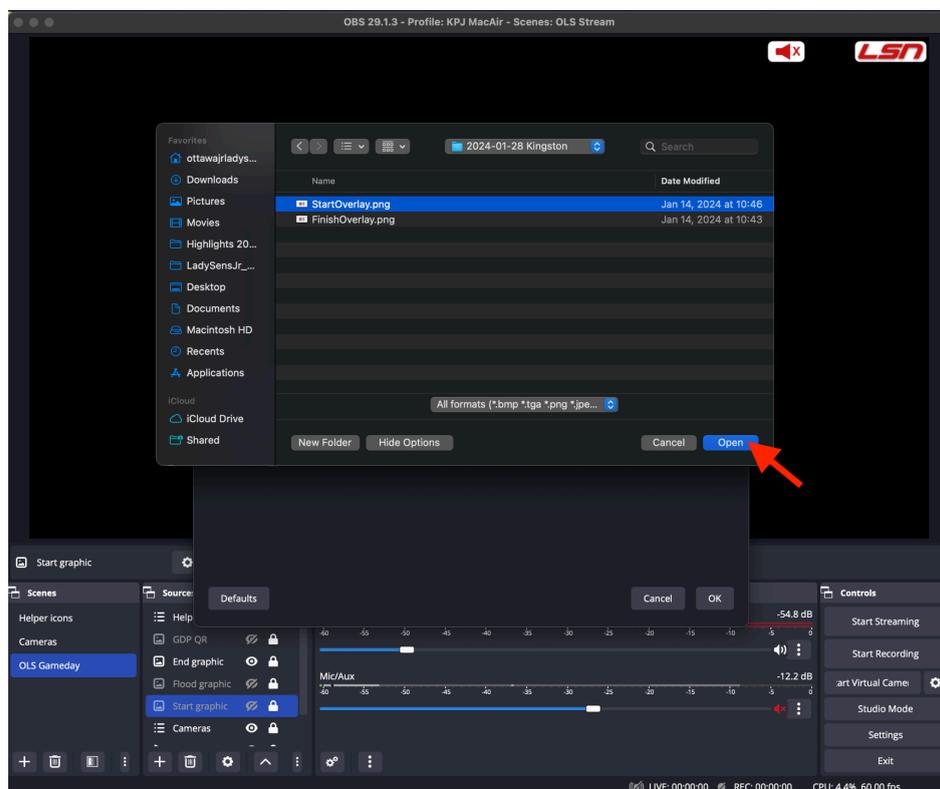
And then when the file is previewed in the window, select OK. This closes the dialog box and the "Start graphic" element is now pointing at the correct file. Repeat this for the other graphics that need to be updated. This is usually "Start graphic", "End graphic" and "GDP QR" as these are updated every game. It is OK to shut down OBS, as it caches these selections and will maintain them when OBS is restarted. This allows set up before going to the rink for example.

Hotkeys and Macros

I have hotkeys and macros set up to allow toggling of overlay graphics, mute/unmute, etc. using the numeric keypad. This allows the stream to be easily managed during the broadcast without having to reach down to the keyboard on the computer. The keypad simply acts as a programmable external keyboard. I have simply made this the equivalent of a "numeric keypad" with the keys being assigned the numbers "1" to "9". Other choices are valid, but this gives a simple to remember and easy to configure solution.

OBS has an included tool called "Advanced Scene Switcher" which can be found in the Tools menu. There are two tabs for configuration, one for General and one for Macros.
The defaults for the General settings are likely OK, but this shows my settings.
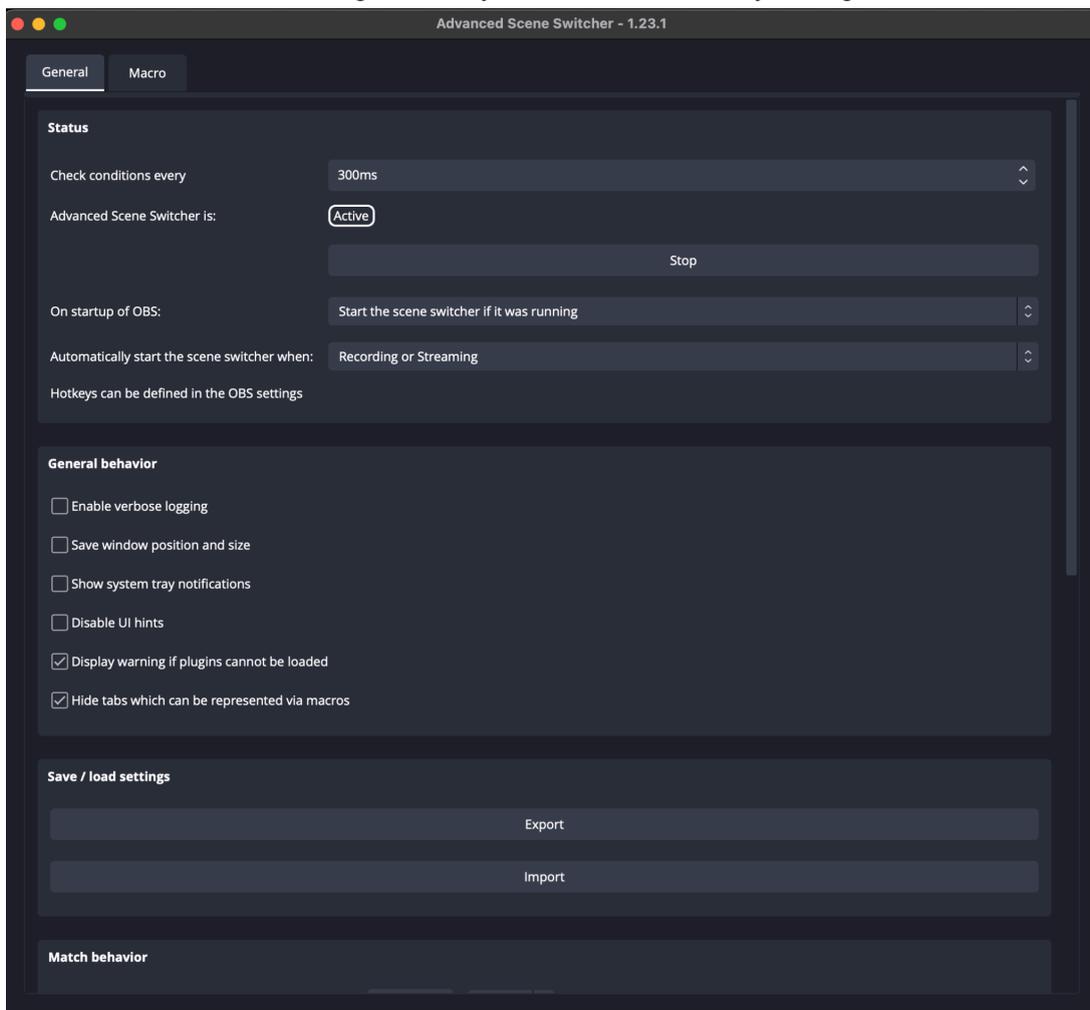
Having the tool check every 300ms ensures that there is not too much lag in the system responding to your keypresses. Then you need to set up the Macros that you want to use by switching to the other tab.
I show a couple of examples of Macros that I have used. One is for Mute the Audio and the other to unMute the Audio. The way that I operate the stream is that instead of a complete silence when the rink audio is muted, I have another audio track of royalty free music at a very low level. This provides some sound so that the viewer knows that stream is not frozen. The Macro effectively switches between these two sources.
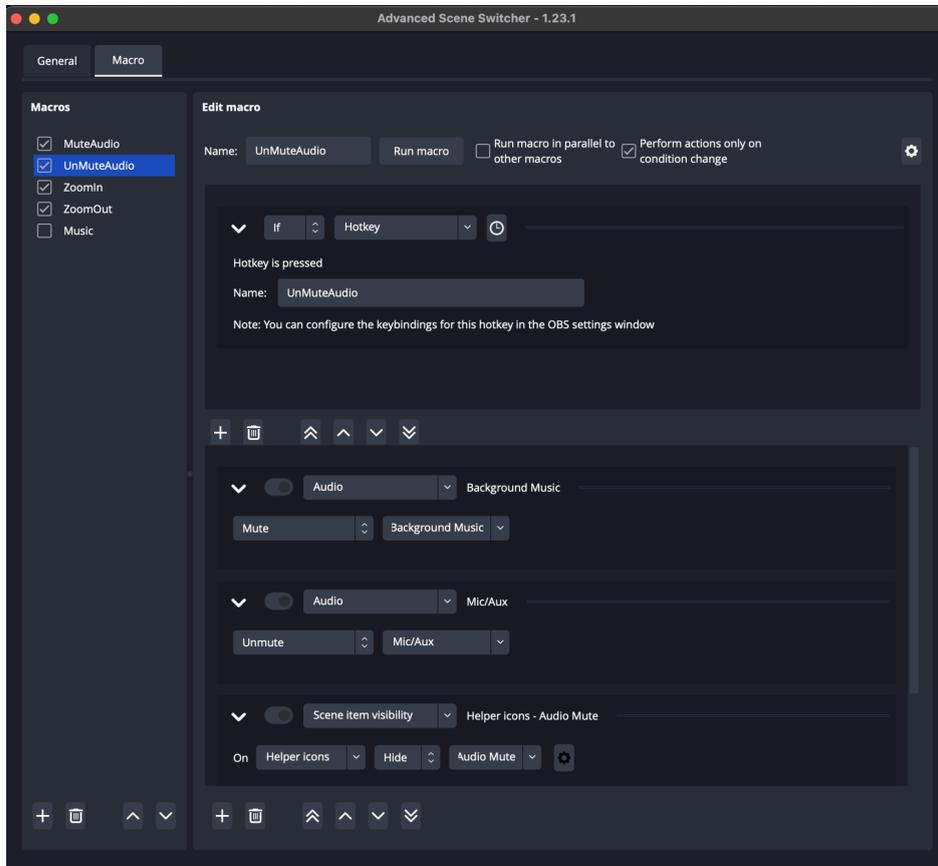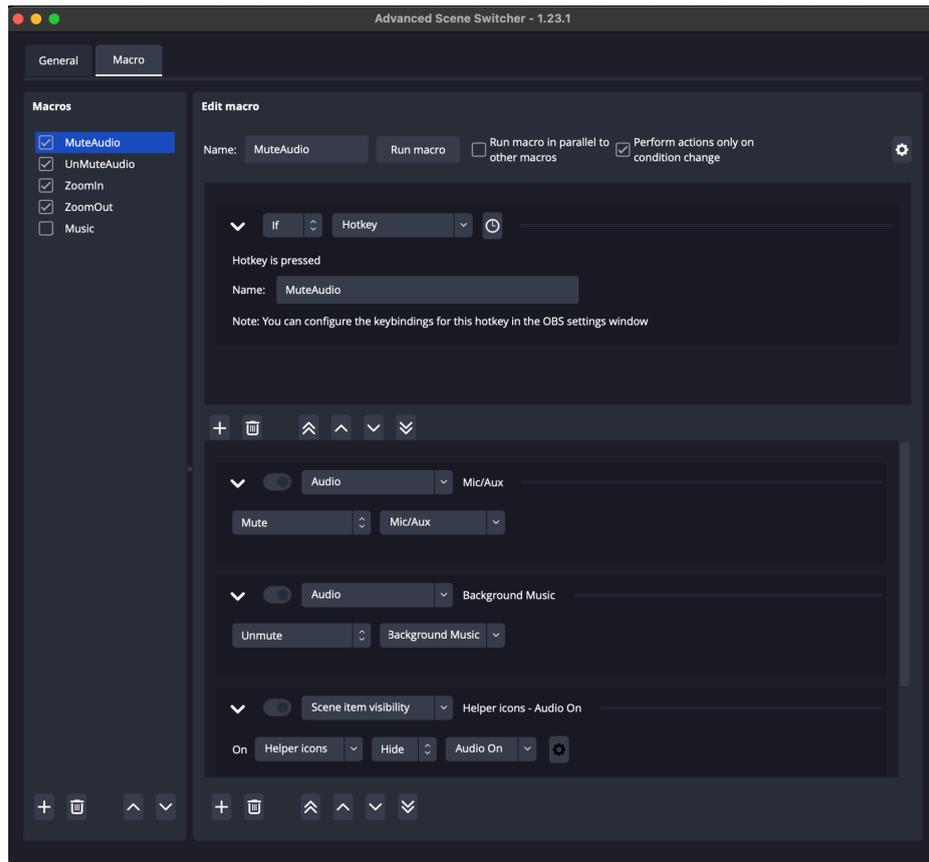
The same action also changes the on screen graphics so that the user (and the live stream operator) can see that the rink audio is on or off.

The Macro then needs to be connected to the hotkey that you wish to use. This is done in main OBS settings (from the button in the lower right hand side). Within the Settings window, select the Hotkeys menu item and you can see lots of actions that can be triggered by key presses. This list also includes the names of the Macros that you have created in the Advanced Scene Switcher. Assigning a key (e.g. one of our number keys that is also available on the keypad) to the macro ensures that when that button is pressed, the macro is executed.
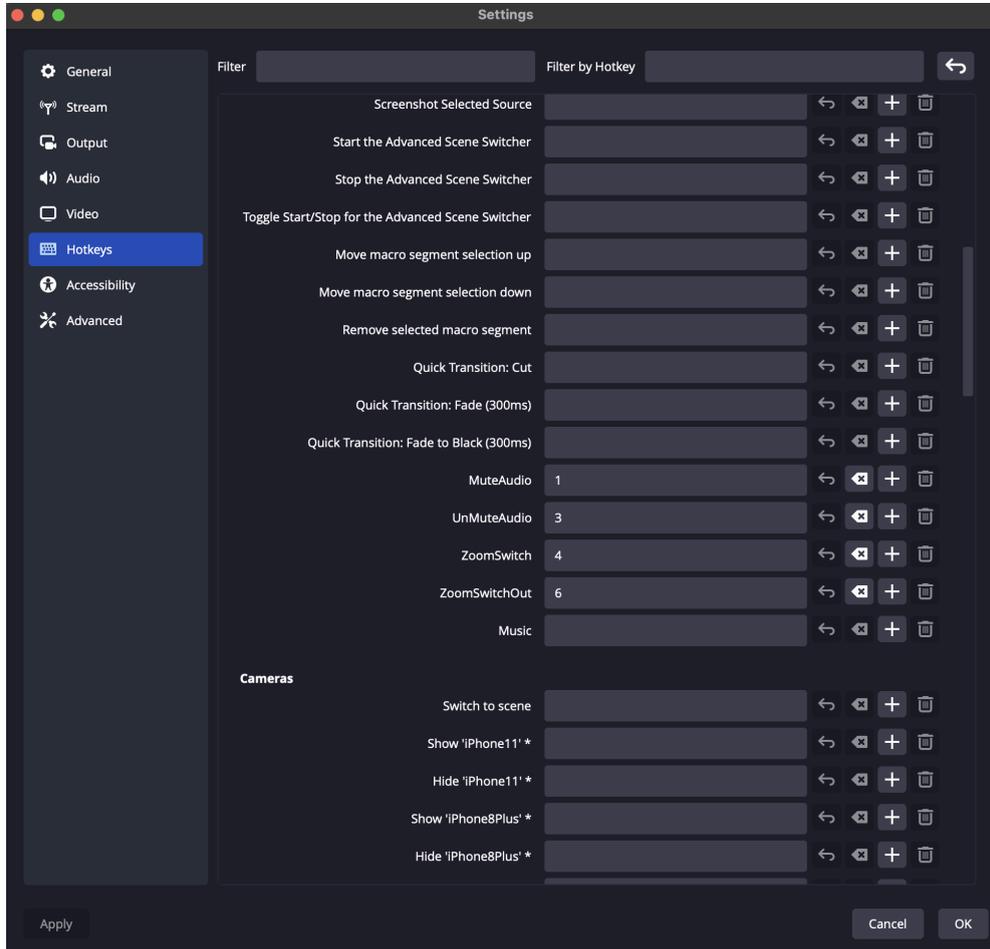
The Mute and Unmute are assigned to "1" and "3" respectively, so pressing the "1" key will mute the audio and "3" will unmute the audio. Now you can test your macros and make sure that they behave as you expect. This can be a bit fussy to set up initially, but can be a great addition to managing the stream.

There are some minor bugs with hotkeys (at least in the version of OBS that I am using) and it is useful to be ready with a couple of work-arounds. If the cursor has selected the Scenes or Sources panel, sometimes the hotkeys do not work as expected. I find that clicking on the open space below the audio switcher (shown by red arrow below) often allows the hotkeys to work properly again. This is certainly something to do before every live stream session to give the best chance of working.
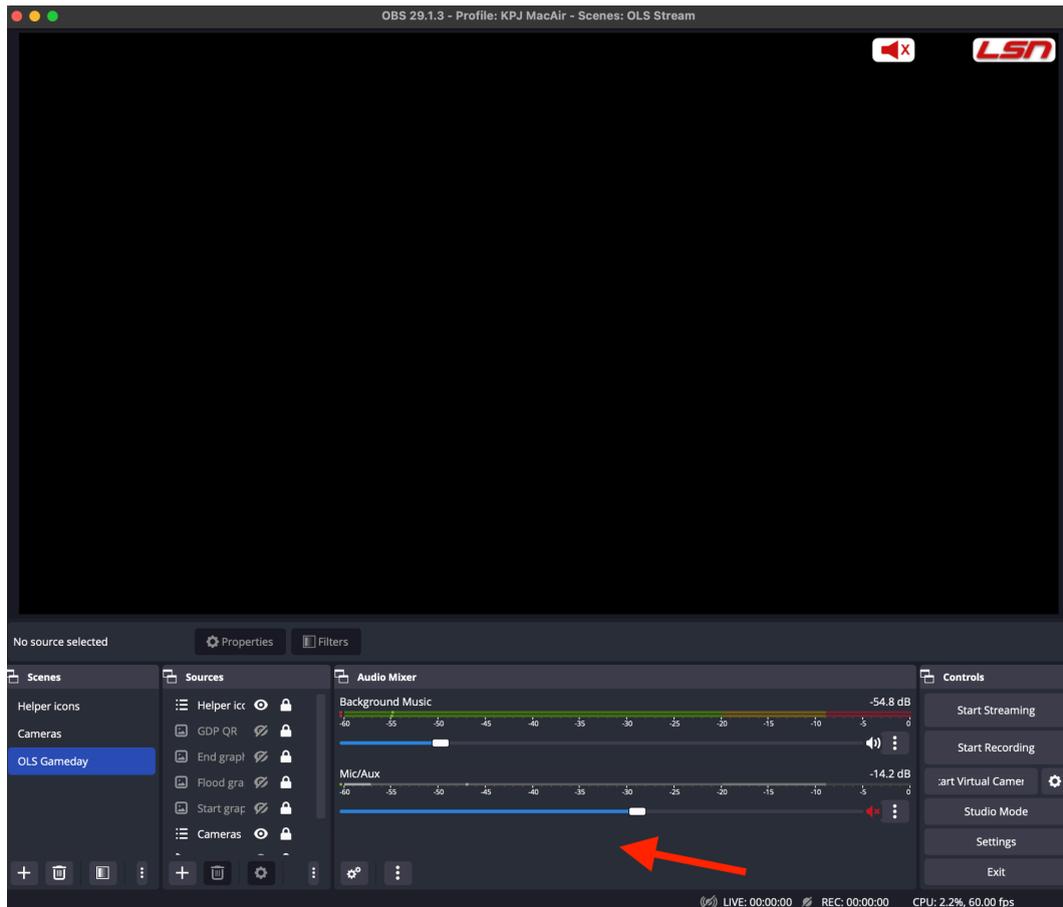
Occasionally (say 1 in 20 games) the hotkeys will randomly stop working during the game. In this case, it can often only be solved in real-time by going back to using the mouse/trackpad to click on the GUI to take the action (mute/unmute) that you want. If this happens, don't panic. During the flood, stop the stream in OBS (but leave it running in YouTube) and close and restart OBS. This generally fixes the hotkeys and you can then start the stream again. This will automatically restart the frozen video on the livestream and you can continue as normal. This does not happen often, but it is good to know the work-around for when it does !

Shoot live - running the live-stream

OK, so it's gameday and time to run the stream. This section assumes that you have the basic set up confirmed and all the cameras checked to be working, graphics ready and all we need to do is set up in the rink. Here is the process to follow at the rink…

Configure the set-up

- Open the Macbook Air and login
- Connect to the MiFi hotspot or local WiFi
- Speedtest the network connection to establish upload speed
  - Adjust network or MiFi location to get sufficient bandwidth
- Start up Google Chrome
- Select YouTube Studio - Create - GoLive
  - The select the pre-scheduled stream for the gameday
- Start up OBS
- Connect main camera directly to the Mac
  - I usually connect the main camera first, followed by the clock camera. If you do the other order sometimes the OBS source can get confused which is which.
  - Start Shoot app on phone and see the OBS preview show the camera output
  - Adjust the Zoom on the iPhone to get the desired shot of the ice
- Connect scoreboard camera and keypad to Mac via adapter
  - Start the Shoot app on phone and see the OBS preview show the scoreboard camera overlay
  - Adjust position of camera and settings using…
    - Zoom on iPhone. Adjust angle of camera on monopod to get level
  - Adjust in OBS using Cameras scene - Source - Choose camera - Filters - Crop/Pad to isolate the scoreboard and configure the overlay on the OBS preview
  - Click the padlock and resize the clock camera in the OBS window if required (it is usually OK from the last setting unless the shape of the clock is different).
- When the cameras are configured, select the "OLS Gameday" from the Scene menu and then click the mouse underneath the sound selections in the Audio Mixer. This is important, as it deselects the Scene menu which can sometimes interfere with the macro operation.
- Check the keypad shortcuts work
  - Test the overlay graphic toggling and mute/unmute
- Test the stream **without** enabling the YouTube dashboard to GoLive - this allows the connection to be tested and if needed the streaming rate to be adjusted
  - Return to the OBS window. On the bottom right of the window there should be a green box indicating a good stream (this will turn yellow or red if there are streaming issues). There is also a timer and a dropped frame counter. Usually I run this for about 5 minutes to ensure that we have no dropped frames. If we see frame drops during this test, we should consider reducing the Output rate to deliver a smoother stream for the game. To do this, you need to stop streaming, adjust the rate and then restart the test to confirm. Once this is good, we can stop the stream and wait until we are ready to go live.
- The set up is now ready to run and record the stream and we can pause until the Zamboni comes out to flood the ice before the game.

Run the stream and backup recording

- When the Zamboni starts the flood before the game it is time to start the stream and the recording, as this gives time for viewers to get the notification and start watching…
- Unlock the backup camera (iPhone8) and select the Apple Camera app
  - Select Video, pinch to Zoom to set the best framing of the ice and then press Record. This can run until the game is finished and is an independent backup solution in case other problems occur.
- By carefully arranging the windows we can monitor the stream and the number of viewers in the YouTube Dashboard, whilst also using the main OBS window on top to see how well the streaming is working.
- In OBS select the Start Streaming and confirm this is what you want to do
  - This will stream to YouTube and record locally to disk at the same time
- In the YouTube window, after a few tens of seconds you should see a preview of the video appear and the GoLive button will activate.
  - Press the GoLive button to start the stream
  - Click on the OBS window to pull it to the front again. By overlaying the OBS window on top of the YouTube window you can still monitor the stream
- In the OBS window, click on the space below the audio mixers to make sure that the hotkeys work correctly.
- You are now live, so operate the stream.
  - Add/remove overlays and mute/unmute the audio as required.
  - We usually mute any music playing in the rink to avoid copyright strikes against our YouTube account
- After the game, stop the stream in YouTube first and then select Stop Streaming in OBS
  - When this has completed we are done and can close the computer and pack up the equipment
  - Don't forget to stop the backup recording on the iPhone8 too.

Post processing

Edit and archive recording

After the game is over and the gear is packed away, it's time to process the game for long term archiving. The live streamed version is already up on YouTube, and provided the stream ran smoothly, this is effectively complete. If you want to make the video easier to access, you can add times in the description of the game that indicate the start of each period, or other key moments for example. If you enter the time in the format mm:ss (or hh:mm:ss) then this will automatically turn into a link that jumps the video to this time. This can be a useful shortcut.

We can now take the recording of the game on the laptop (which is in our maximum quality, e.g. recorded in 1080p high bandwidth when the stream might have been published at 720p) and edit if for long term archiving. Open up the recording in your editor of choice (QuickTime, iMovie, Final Cut Pro, Premiere Pro), trim off the non-game time from the start and end of the recording and remove the footage of the floods between the periods. This removes all the unnecessary video, but keeps all of the game that we want. This can then be exported (recompressed) at your preferred data rate. This gives an archive version that you can keep on an external disk and use to generate other clips, such as goals, saves or highlights of the game.

Upload to analysis package

Many teams use video analysis packages such as [Hudl Instat](#) or [TPE](#) to extract additional analytics from the recording and use this for player feedback. We have used Hudl Instat and after the game we need to upload the game to the platform (I assume this would be very similar for TPE). When you have a Hudl account the game can be uploaded using a form on their site. Fortunately this can use a link directly to the YouTube recording so that you do not need to re-upload the video (although you can if you prefer) and provide a link to the gamesheet. In the OWHL U22 Elite league, all the gamesheets are on-line and so this is very simple. Simply paste the appropriate links into the form and then submit. After processing the video, the statistics will appear for the coaches and players to review (usually available next day).