live.smokyvalley.org

How are we doing it?

Our streaming setup has evolved over time, but it started with the simple need to share concert and sports events with family members that weren't able to attend due to illness or distance (including parents enlisted in the military). It started small a number of years ago, and never really grew because of limitations of the services we were considering.

Most services included ads of some kind, and also limited us on how much video we could archive and even how many cameras we could use with their software. HD quality was important, but also cost prohibitive. Other services didn't integrate with our website very well either, so it was never easy to get people to the video stream.

That all changed when Google loosened its requirements on their Live Events service. We had recently migrated to Google Apps for Education (not a requirement for you by the way), and that had us looking into Google's somewhat obscure service offerings to see what they could do for us. Live Events fit the bill because:

- It didn't put a limit on bandwidth for our clients
- There were no ads
- We could use as many cameras as our bandwidth would handle
- There were no client count limits
- There were no archive limits
- We control who has access to what and when
- It works on *everything* that plays YouTube (phones, SmartTVs, AppleTV, Roku, Amazon Fire, ChromeCast, computers, tablets, etc.)
- It could handle all the connections we could throw at it.
- Did I mention it was free?

I should also mention at this point, that we weren't interested in charging people to watch our content. We're looking to provide a free service to our patrons. If you're looking to replace gate receipts, I'm not sure this will be the perfect solution for you, though there is some monetization possibilities with YouTube.

So now that you know why we settled on YouTube, let's get into the hardware requirements for YouTube Live Events:

Video Source
This can be a camcorder, a video mixer with multiple sources plugged in, or even a computer with some form of video-out port

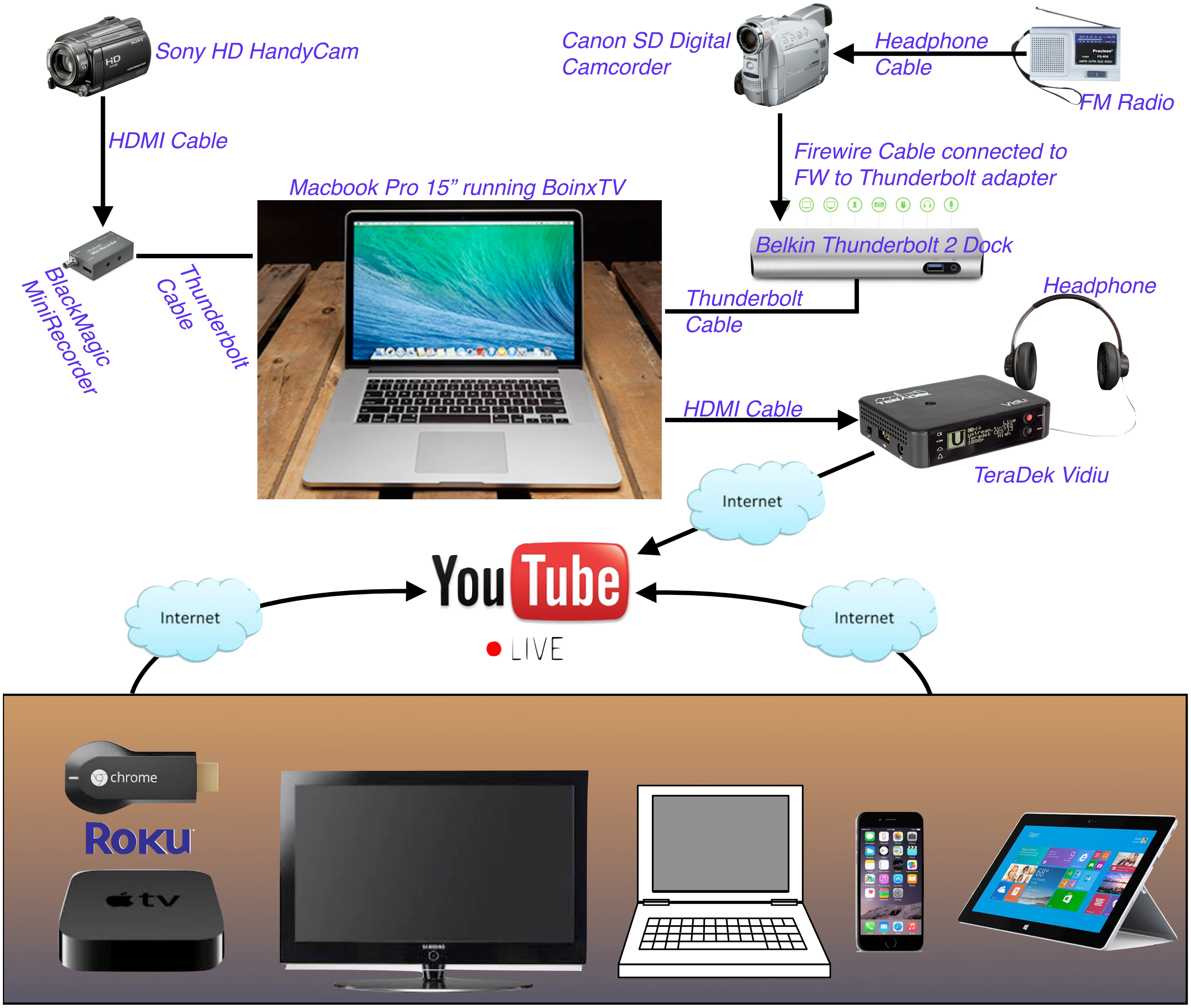
Encoder
Either a dedicated piece of hardware (the Teradek Vidiu is an example) or specialized software running on a computer. As we tried a myriad of services, they all required you to download and install their encoding software to use their service (iHigh Production Truck, UStream, etc.). Wirecast is a popular encoder (or has encoding capabilities) but we're cheap, and went with the free Adobe Flash Media Live Encoder. There are limitless software encoders for every platform under the sun, so find one that works with your service and on your computer. Encoders are processor hogs, so make sure your rig has enough horsepower to pull it all.

Internet Access

Make sure you have plenty of bandwidth at the location you'll be broadcasting from. Most encoders will have a bandwidth test function that you should use before every broadcast to ensure things are what they should be. We broadcast from our gym using copper ethernet plugged into the encoder. For football and concerts, we use wireless connectivity. Depending on the encoder, you may be able to plug a 4G cellular modem into it and broadcast using that (obvious rates apply). Just make sure there's enough bandwidth to handle the stream you'll be pushing.

Our Setup

That's the bare minimum; everything else is extra credit depending on your needs. Here's what our current rig is running and how it's all hooked up:



Here's a breakdown of each device and it's purpose:

Sony HD HandyCam - This is the 'Main' camera that we use to record the action, either on the stage, court, or field.

BlackMagic MiniRecorder - Acts as an HDMI to Thunderbolt adapter to allow video processing on the MacBook Pro.

Canon SD Digital Camcorder - This is the stationary 'Scoreboard' camera and also accepts the audio feed for events (We won't get into the technical reasons why it's configured this way; it was the easiest and works, though probably not the most 'sound' solution - pun intended.) This camera was dug out of a box of old tape-style cameras that are broken and no longer record. The Canon camera is connected via Firewire to a Thunderbolt adapter that's connected to the Belkin Dock.

Belkin Thunderbolt 2 Dock - Creates multiple ports via Thunderbolt including USB 3, Thunderbolt, HDMI and Ethernet. Probably not completely necessary in this specific setup, but the additional HDMI port and Ethernet port has saved some bacon.

MacBook Pro 15" - Runs BoinxTV for video mixing and creating the Picture-in-picture for the scoreboard, as well as showing text on the screen, etc. Boinx is also used to create an onsite recording that's saved to the local HD in case of technical issues that cause the stream to go down; in such an event, we can upload the file to YouTube. Live Video is dumped via HDMI to the Vidiu.

TeraDek Vidiu - Deck-of-cards-sized box that encodes HDMI video and transmits stream to YouTube. Supports a number of other services as well, but works very nicely with YouTube. Headphones are plugged in so the operator can hear what's going out on the stream. The device will transmit *any* HDMI source, so a mixer board could be plugged into it just as easily as a computer. Ultimately, one could stream with just an HDMI-capable video camera and this device.

Some clarifications

One question you might have is '*Why not just use the Adobe Flash Media Live Encoder application? It is free after all.*'

At the beginning of the 2014 school year, that solution was used, but it required some additional software to run in order for it to work the way we wanted it to. The Picture-in-Picture scoreboard is important to us (for the viewer's sake), so in order to do that, we used BoinxTV to mix the two video sources. That combined video signal was dumped to another app called 'CamTwist' (free for Mac) that turned a video source into a webcam. That 'webcam' was then encoded using Adobe Flash Media Encoder (free for Mac and PC) and pushed to YouTube. It worked, but it took *lots* of horsepower and pushed the MacBook Pro to it's limits.

A solution was found by offloading the encoding to the TeraDek Vidiu ($700). It eliminated CamTwist from the mix and required that only BoinxTV was used for mixing. It's a little spendy, but makes streaming dead simple, which is important when thinking of students streaming video.

Another question you might have is '*Why the FM radio?*'.

To give our viewers the 'full experience', color commentary was really important when watching the feed. Some districts use students or local college kids to do the play by play, but (people) resources being what they are, we opted to tap into the radio feed. The FM Radio was a quick and dirty solution that saved running cable to the broadcaster's booth during the football games and one that works well with basketball games. There's about a 1/4 second delay for the audio which is practically indiscernible. For concerts, we'll set mikes up and tap into those for streaming.

Final thoughts

We are an all Mac shop, so that's why there's such an emphasis on a Mac solution. Obviously, you don't have to go that way, but it does prove that a very descent product can be put out on a Mac solution, and if it can be done on a Mac, there's no reason whatsoever you can't get it done on a PC.

Going Mac makes resolving some issues interesting (it took sometime to find a good solution for HDMI to Thunderbolt, but we did and it works great). Software is another example; we had a license of BoinxTV laying around and that product has worked very well for us. As you're designing your rig, just remember there are usually connectors/converters/adapters out there to connect anything to anything; just Google it.

Another tip is to go back and watch the stream after the event. Doing so will help you identify any technical issues you might be having and not know about, but it will also tell you if the end product is enjoyable to watch. If it's not enjoyable to watch, then make changes accordingly; we added the scoreboard and commentary to make it more enjoyable and it's helped the watchability tremendously.

You can spend as much or as little as you want on a rig, but define what you want your audience to see first, then design around that. You may have to phase things in and let it evolve over time, but be patient and be creative with your solutions. If there's no money available from your district, consider outside sources (Boosters, alumni organizations, etc.) for support. There's also an obvious place for sponsors to be mentioned in the broadcast which could offset the cost.

Probably the most important resource you have to find is people to run the streaming setup. We plan to use students to do it, but we're small enough, and our students are involved in so many things, finding kids to run it has been a challenge. The commentary is handled through the radio, and for football, the main video feed was handled by the team (we just tapped into their camera). With basketball, the new HUDL system uses an iPad to record, so team managers are using that, requiring a dedicated camera operator. Fortunately, the rig is streamlined enough for one person to drive the camera and control the laptop, but it's one more student that needs to be found. There's no perfect solution to this issue, but you should be aware that it could be an issue since it's something new.

One final thing to keep in mind is the viewer's perspective. In english class, we were taught to 'write to the audience'. The same goes here (and was addressed earlier), but maybe more importantly, before they can read it, the audience has to find your composition. We're working on 'branding' live.smokyvalley.org to give consistency for folks to find our product. We use Twitter and Facebook to get the word out about upcoming streams. Use whatever communication lines you have to the public to share the word about how to find your stream to make it easy for folks to find it, but be consistent.

Taking on streaming can be an overwhelming and daunting task. If you have any questions or want clarification about how we're streaming, or if you just need help getting things going, please contact our Tech Director, Mike Rose at mrose@smokyvalley.org