

3Play Webinars | DIY-Captioning-Webinar

LILY BOND: Welcome, everyone, and thank you for joining this webinar entitled DIY Workflows for Captioning and Transcription. I'm Lily Bond from 3Play Media, and I'll be moderating today. I'm lucky to be joined by Ken Petri, the Web Accessibility Center Director at Ohio State University, who has a fabulous presentation prepared for you today.

His presentation will be about 45 minutes. And then we will have around 15 minutes for Q&A. And with that, I will hand it over to Ken, who has a great presentation for you.

KEN PETRI: Thanks, Lily. So I'm Ken Petri. I direct the Web Accessibility Center at Ohio State. It's a small office, but it has kind of deep roots.

We're part of Disability Services. And I report to the ADA Coordinator's office. And so my scope is pretty broad.

But we have a really small staff. It's just me and a GA, who I'll talk about in a second. We've been doing stuff on captioning, and have had captioning materials up on the web for quite while.

And I'm going to talk about those two. They're, of course, free, and anybody can go and get them. And if you have suggestions about any of them, please give us feedback, and I can update them.

So the talk has, I guess, a really long part and then a really short part. And this is the really short part. I wanted to talk just a little bit about, in terms of DIY, what our in-house service is right now.

So OSU recently contracted with a vendor for production of captions for some of the courses that are solely online through our office of This is Education and e-Learning. But we've had, for a while, a service called Transcribe OSU that started as a grant, and for the first year or so, under the grant, was a free service, just offering transcription about four years ago.

And when funding ran out, we decided to sustain it. And now it's kind of in a cost-recovery mode. And I'll talk a little bit about the details of it.

If you go to the website that's right there, go.osu.edu/transcribe, it'll redirect you to our wiki page for Transcribe OSU. We're a vendor internally at the University. People can access us by contacting our email and filling out a form. And then they can pay through internal means.

It's an OSU-only service. It's run by my GAA, my graduate administrative assistant. And we hire undergrads to do transcription. And for a long time, they were doing just transcription. And we had anywhere from three to as many as seven, depending upon load and turnover.

In my mind, the advantage to in-housing something is that the OSU dollars sort of stay here at OSU. The students benefit from that, because they're getting paid for it. And we pay, I think, relatively competitively with other jobs on campus.

And in terms of output, some of the students will gravitate to certain kinds of courses or certain kinds of material that are in their wheelhouse. So there's a possibility that they'll have domain knowledge when they do the transcription. And because they're on campus and using our campus systems for exchanging files and things like that, it's a lot easier to have personal contact. Like I mentioned, this is an OSU-only service, and it's actually only OSU main campus. We have done some of the regional campuses, but predominately, it's for the main campus.

So costs-- we charge \$60. So we had to go into cost recovery, like I mentioned. And we charge \$60 per hour of transcribed media. So if they just have a video and they just want a transcription of the video, or if they have an audio file and just want a transcription of that audio file, it's \$60 per hour of media-- so not \$60 per hour of our labor, but \$60 per hour of media.

And if they want a timed caption to go along with that, we can give that back to them in a number of different formats. And that's an extra \$10, so \$70 if we go ahead and caption it. So the way this breaks down-- those are pretty cheap prices.

One is because the labor is really cheap. Not necessarily something I'm proud of, but that's the reality of having undergraduates as labor. They can work totally on their own and can do these transcriptions at any time that they want.

And so that's an advantage. And so we like to have that flexible schedule to kind of retain them, because there is some itineracy. We figure that there's a ratio of about five hours of taking the time to get the transcript for every one hour of media. That's what we budget as part of our costs.

Then, there's a review process. Every transcript that we produce goes through a review by a

second person. And then if there's captioning, so there has to be a timed transcript, it takes about an hour for that.

And some of these numbers are not solid. So sometimes, we'll get something that's particularly tough. And it will take quite a lot longer to do it.

But we figure if we come in at \$60, there'll be some easy files. There'll be some tough files. We can kind of make up the discrepancies.

Since last year this time, we've done about 215 hours of media. And since we just started doing the captioning, mid of last year, about a third of that media is captioned video. So I can talk more about this service later, if people are interested.

So most of the talk is about the kind of DIY aspect of this, and then kind of technical what you do aspect of this. Presentation is a little bit Windows-centric, less so, actually, than what I thought. There's really only one or two pieces of software that I mention that are Windows only.

Concentrating this, at least in this workflow, on using YouTube for transcription and timing of captioned material, and Express Scribe for transcription of audio. So Express Scribe is a package I'll talk about in a second, but it's dictation software. And it's free.

I'll talk about the difference between the Pro version and the free version. It's a significant one, and it's not very expensive to buy the Pro version. So I'd say, buy the Pro version.

We use Microsoft Word or LibreOffice, which is the open source correlative of Microsoft Word, for its spell-checking ability. And it also helps us break the caption up, chunk the caption, and get line lengths right. And I'll talk a little bit about how we do that as we move through this.

There's a piece of software-- this is a Windows-only piece of software-- called Subtitle Edit that is just part of the tool kit. It can provide automatic translation. It uses Google Translate to do automatic translation of captions. But really the thing that it's best at is, if you need the caption in a different format-- so say you're moving from an SRT file or some other time-text format, and you need to get it into TTML, because that happens to be the format that you need it in-- Subtitle Edit does those conversions very slickly. And you can throw pretty much any imaginable caption format at it, and it will do the transcoding for you.

I don't know what kind of [INAUDIBLE] I want to give at this point about these two tools. But at

some point, you end up with-- you've got a video file, and you've maybe got a secondary audio file that's your audio description file. And you have a timed caption file. How do you stick all those things together into a single file?

And for that we use two tools, one that is both Windows and Mac, called mkvmerge GUI. And another one that's, right now, at least the GUI version of it is, PC only, but there's a command-line version which you can run on the Mac called MkvToMP4. And I'll give you more detail about what those do in a second, and give you a demonstration.

If you have to provide a burned-in caption-- so an open caption, one that can't be closed-- any video converter, there are lots of things out there that will do this kind of work, but any video converter-- that's the name of the thing-- will do it pretty effectively. It seems to be pretty stable, and it works both on the Mac and on the PC.

There are a couple things that we play with on occasion. If you're doing a lot of subtitling, and you're on the Mac, and what you're targeting in particular are things like iTunes U, definitely check out iSubtitle. It's a really nice piece of software that does what mkvmerge GUI and mkvToMP4 do.

It'll take your subtitle file and your video file, and you can add, actually, more subtitle files to it, or more caption files to it, so you can have them in multiple languages. And then it'll merge them together into an M4V container, which is the format that iTunes U or iTunes likes to play. So if you've got, on your campus, an iTunes U account, in order to get captions in that thing, you have to have the captions in the file that gets uploaded.

They're not a separate file. They're baked into the M4V container that gets uploaded. And iSubtitles is good at doing that.

Handbrake is also one of these kind of emerging tools. It seems on occasion it works better on the Mac than it does on the PC. There's PC and Mac versions of it. And I've been really frustrated by it numerous times. It said it will do things that it doesn't actually accomplish. But if you're on the Mac, it is free. And it's worth checking out. It's kind of the correlative of mkvmerge GUI.

So the basic workflow for moving from source video to a publishable format is, you need to get a transcript. So that's probably the most time-consuming part of the whole process, is getting the transcript. You need to break the transcript up, or chunk the transcript into-- if you're going

to do a caption-- into screen-sized pieces. We call that process chunking.

And then you need to synchronize the transcript with the video. In other words, you need to create, for every chunk, a time scan in the video that it will align to-- so a time that that chunk comes on screen, and the time that that chunk goes off the screen. That process we call synchronizing. And then going to mention later in the slides, going to talk a little bit about audio description, what an audio description is.

And I don't mention translation, but, I mean, that's clearly if you're trying to hit an international audience, or you've got material for a foreign language class, you're going to want subtitles in languages other than the original language. And iSubtitle can help with that, or Subtitle Edit can help with that a little bit, just to get you to the point where you've got something to work from. But none of the automated translation solutions are going to be probably reliable for high quality end content. So really, doing the translation of the file is a kind of manual affair.

Just to note here, you really want to have the transcript timed before you do the translation. So you would time it in its original language. And then you do the translation, and fit the translation into those timed chunks.

And then we're going to talk about how you glue all this stuff together, which really depends upon what you're going to target as your endpoint. So are you going to put the video, embed it into a PDF? Are you going to put it into a PowerPoint?

Is it going to be web-based video? Or is it going to be a standalone file? So depending on what your target is, that's how you're going to glue the parts together.

And then publishing, which I guess I kind of already discussed there. So does it go on the web? Does it go into a PDF, a PowerPoint, or some other kind of standalone media? Is it something that's inside of another piece of media? Or is it just a standalone file?

OK, so getting a transcript-- I would imagine if people have been doing this for any amount of time, they've run into Express Scribe already. There's a free version of it, and there's a Pro version of it. And the main difference that I see is the Pro version of it allows you to play back video inside of the tool itself.

So the free version will play back an audio file. And you can use it to just generate a raw transcript from audio. But the Pro version allows you to just drag your video into the thing, and it will play the video inside of the Express Scribe environment. And especially for transcription

of complex things, it's really nice to be able to see the thing that you're getting the transcript for-- see the video, and see the people interacting. It's much easier to be able to note that this guy here is Stewart Cheifet than it is to just recognize his voice on an audio track.

So I think the Pro version is like \$30. Sometimes they offer bulk discounts. We use this mostly for transcription of audio only, but we will do video on occasion using this tool.

It has keyboard-assignable hot keys. So to speed up the transcription process, you can assign hot keys for a whole range of different things. But the ones that we tend to map always are starting and stopping the video, which, unfortunately, there's not like a toggle for pause/play.

There's Stop, which pauses the video. And then there's Play, which has to be a different mapping for each. So we mapped the pause to-- and this actually may be the default-- I'm not sure-- but we mapped the Stop to Alt Z. So the Alt key plus Z.

We mapped the Play to Alt plus A. And we mapped the five seconds back to Alt plus Q. So they kind of line up on the left-hand side of the keyboard. You can get to them really quickly, and just hit Alt-Q, Alt-Q, go 10 seconds back, and let it play forward.

Another great feature of this is it has the ability to slow the playback down by a percentage. So in the screenshot here, you can see that it's playing back at 59%. If you look down to the right-hand corner, there's a little playback speed indicator on the screenshot there. And it's circle says 59% in the circle.

It doesn't just slow the playback down. It also changes the pitch when it slows it down. So you don't get someone sounding like they're talking in five octaves lower than they normally would when you slow it down.

The pitch of the voice stays for the most part the same. It's just the speed that slows down. So that's a really nice feature, and pretty much a necessary feature if you've got somebody who's talking really fast.

There's also a mini player. So I can put my video in this thing, and then hit the mini player. And then I can tell the mini player to stay on top of all other windows.

I can kind of squeeze it up to the corner of my screen, and have the mini player playing in a kind of compressed version, which you see on the screen here. And all the shortcut keys will stay the same, no matter what program I happen to be in. So if I'm taking the dictation, writing

the transcript into, say, Microsoft Word or something, I can still hit Alt-Z, Alt-A, and Alt-Q to have those functions available.

So it doesn't matter that Word is currently the focused application. Those hot keys are still live and available. So I can adjust the speed, or whatever other functionality I want inside of Express Scribe. So it's a really terrific little tool.

Another way you can get a transcript is if you have a video, and the video has either no copyright, or it has the sort of copyright restrictions that will allow you to upload it to YouTube. If you can upload to YouTube, and now if you upload into your own account, every video that goes in there will receive an automatic caption. Now, you can see in the screenshot here that there's kind of nonsense text in the middle of this thing.

So it says, "Knollwood need to look at the doc election and tell you that on purpose." OK, well, I can't imagine in what universe that would make any kind of sense. What he's actually saying here is, this is Kelly Flock, who's the inventor of this game, he's saying, no need to look at the documentation.

And then, there's a speaker change. And Stewart Cheifet gets on. And he says, woop, I can tell you did that on purpose, because Kelly's doing some kind of trick with a skateboarder on the ramp.

OK, so if you end up with an automatic caption that looks anything like this, my advice would be chuck it. It's not going to do you any good to try to fix it, or download it and fix it, or even worse, try to fix it inside of YouTube. It's basically garbage. You're going to be much better off by just starting from scratch.

However, if you're narrating a PowerPoint for, say, a class that you've put together, and you're doing a screen capture of the PowerPoint presentation, and you're speaking clearly and slowly, chances are that the YouTube automatic caption will be something that's pretty close to what you're saying. And it'll be just a few edits to allow you to edit.

So what you can do is, after it gives you the automatic caption, you can download the automatic caption. And then you can strip out the caption timings. So there's no way to download the caption just as raw text. You're going to get all of the timings along with it.

So if I go to where the Action button is, and I click on Action, it'll say download in VTT format,

in SRT format, or in SBV format. Those are just some basic caption formats. All those caption formats have each of those chunks of text along with the timing.

And it can be hard just to strip out the text alone. I wish YouTube would just add a Download as Text. That would be great. But they don't have one at this point. They probably will soon. But right now, there's not one. So you have to download one of the caption files.

But-- and I'm going to switch out of this presentation here for just a second. I'm going to show you how easy it is to actually strip these out. So let's see, where am I?

So now on the top of the screen is a piece of software called Subtitle Edit. And this is a Windows-only piece of software. But I've dragged and dropped the caption file that I downloaded from YouTube into this thing.

And this is actually a corrected caption file. But I've dropped it in there. If I had a decent automatic caption, that I got from YouTube, that I downloaded from YouTube, but I still had all the timings in it-- because I don't really want the timings.

So the timings, when I download it from YouTube, is going to look like this. It's going to look like some text that has-- this is the SRT format. It's got these timings in it.

And I suppose you could go through here, and you could manually delete all these. But if you've got a long file-- this is a short one. This is only 58 lines. Even getting those manually out of the 58 lines would be kind of a pain in the butt.

So what you can do instead is use Subtitle Edit. And Subtitle Edit will strip all of the caption timings out for you. You could go to File. And you can go down to Export Plain Text. And then you've got some options.

So it says, I can un-break the lines if I want. I can leave the line breaks in if I want. I can show the line numbers, or take the line numbers out.

I can show the time codes, or take the time codes out. But what I would want to do is, take out all the time codes. And if I wanted to, I can even merge all the lines so it's just one, long transcript like that. I probably wouldn't want to do it in this circumstance, but there might be circumstances where I'd want to do that-- if you just have a single speaker, and you wanted to worry about the line breaks later.

Anyway, so this will allow you to take any caption format that you can imagine, and just rip out all the timings from it. And so if you're downloading from YouTube, this can give you the raw transcript, that you can then fix up, and then re-upload to YouTube, and let YouTube do the timings, which we'll talk about in a second.

Anyway, so this is one of the uses of Subtitle Edit. It's probably the main thing that we use it for. You can tell that it flags lines that may be too long for the start and stop times. That's what these little red symbols mean.

So it says, this particular line is only on screen for 1.69 seconds. And it determines that in 1.69 seconds, it's going to be really hard to read that line. So that's what it's telling you there.

This isn't really a transcription tool. It's more like a kind of caption file cleanup tool. But it's pretty handy.

OK, the other way to get a transcript would be to-- I mean, there's a myriad ways. But another effective way is to actually use YouTube to get the transcript. And so you can kind of see ideas moving back toward YouTube.

Where people had external tools, it's now pretty clear that one of the best ways to get a caption outside of just contracting somebody else to do it for you-- which is, of course, the easiest way, but also less expensive way-- is to use YouTube. Upload your video. It gives you an automatic caption.

If you want, you can use that caption. Or you can just not use the automatic caption at all if it's garbage, and use this as a transcription tool. So here in the screenshot, you can see that I've taken the speed of the playback of the video that I uploaded, and I've turned it down to half speed. So it's going to play back at half.

And it does the same sort of thing that Express Scribe does-- it'll change the pitch so that when it plays back, it plays back in a kind of normal-sounding voice, just slowed down. And then it has some hot keys down in this region, where the other red circle is here. I can go five seconds back by hitting, I believe it's the shift and left arrow key. And I can pause it and play it manually-- toggle between pause and play manual-- by hitting Shift and the Space bar.

So you just start playing it. And you can use these simple controls to work your way through it. It also has a don't play unless I'm not typing feature. So that's that second check mark, says, "Pause video while typing."

So while I'm typing, it's not going to play the video. When I get done with my little typing, it'll start the video automatically. And then as I start hearing something that I can catch up with, I start typing. And it pauses. And if you're coming back at the video, doing the transcript of the video, and your speed is set to half speed, this is a really terrific way to get your transcript.

The other thing I would recommend at least exploring is, there are a number of clipboard manager tools. So a clipboard manager will allow you to save multiple copies of text. So normally your clipboard, you copy a piece of text and you paste it. And that's all you can do.

What a clipboard manager does is allows you to copy five different things, 10 different things. And then when it comes time to paste, you can select which one of those things you want to paste in. Some of them even have features where you can paste in by a particular hot key.

So you might have something like Control Shift 1. That'll paste the first thing that you copied into the clipboard. Control Shift 2 will paste the second thing. So this kind of works like a macro.

If you've got multiple speakers, you can copy the speaker identification text. In this screenshot here that's the Stewart Cheifet colon chunk that's in that little transcript window. I don't want to keep having to type Stewart Cheifet, so I might just copy Stewart Cheifet into the clipboard manager, and then associate it with a hot key, so that I can paste it back in every time he starts to talk. So all those kinds of things will speed up your transcription.

Even so, you're typically looking at about four to six times as long to get a transcript as the video is. So just kind of keep that in mind. It's a relatively painstaking process. You want to be highly, highly accurate when you do that.

Now, when it comes time, so say you've got your transcript and you've got it inside of YouTube, or you've got it from using Express Scribe, or some other tool. You can put it into Word. And there are a couple settings that you can use in order to help the chunking process.

So what you want to end up with are on-screen captions that have certain attributes. They need to be no longer than a certain number of characters. And you want them to be semantically complete.

And they need to have speaker identifications in them. And maybe there has to be captions that have sound effects. You can use Word to help you at least get the chunks about the right

length.

So what we do-- what our transcribers do-- is they'll use Word, and they'll take the transcript broken up, just broken up into speakers or into paragraphs, and Paste the transcript into Word, and then set Word to have right and left margins of 0.32 inches. If you set the font to Courier New 10.5 font, it turns out that that line length that appears on screen inside of Word is the same as what will fit at one time on one screen in YouTube. So we arrived at this through some experimentation.

So this gives you a way to control how long your chunks are. And in the chunks themselves, you need to kind of follow some basic rules for how they're formatted. It turns out that any individual line is about 42 characters. So when we're looking at the previous screen, that was the length of a chunk. So that's how much total text YouTube can fit on-screen at one time.

But on each line in that-- so you might have two lines in a single chunk that appears on-screen at one time. Like in the screenshot to the right, on the left-hand side, you got "was convincing the world he didn't exist." Well, that's two lines, but it's one chunk. So it's two lines on the screen at one time. If you take that line and you expand it out, so it's just one long line, "was convincing the world he didn't exist," that pretty much fits inside of one of those lines in Microsoft Word.

To get a good line break, though, you want each individual line to be about 42 characters. And so that's what we're trying to achieve here-- each line around 42 characters, about one to two lines per chunk. And that's the kind of mechanics of it.

There's sort of the human element of it is, you want each of these chunks to be as independent as possible. So if somebody comes in, and reads just "take your stinking paws off of me, you," and then that's all you hear, it sounds incomplete. So the better, semantically whole, understandable chunk is, "take your stinking paws off of me, you damn, dirty ape."

So you want to get your chunks so they're as semantically complete as possible. We tend to break on periods. So if there's a period, we'll create a chunk break.

And we try to break the lines so that they read as fluidly as possible. So "the greatest trick," and then a line break, "the Devil ever pulled," is easier to read than "the greatest trick the," and then a line break, "the Devil ever pulled." and in this case, it's "the Devil ever pulled was." OK, so you're trying to get the chunks to be semantically complete, and the lines to be as easy

to read, easy to scan, as possible. Because they appear on screen for a very short time, typically.

You also, in a caption file prior to timing it, in the chunked file, you need to have speaker identifications. There are lots of different ways of doing this. This is what we do. This is not a hard and fast rule.

But if you do it this way with right angle bracket, right angle bracket, the person's name, and then a colon, when you take that caption file before the timings and uploaded to YouTube, when YouTube does its automatic synchronization of the file, it will ignore those things. So you're more likely to get an accurately timed transcript if you use this format for identifying speaker changes. And this, I mean, if anybody has looked at real-time transcription, you've been in the audience where someone who's deaf is in the audience, and they've got a real-time transcriber in there, you'll see that this is a very common format for identifying speaker changes.

And then there's sound effects. So sound effects is typically in brackets. They can either be after a person's name in the second block in there, where it says Rick Astley crooning.

That "crooning" is a sound effect. It's identifying that he's crooning. I don't know if that's the best sound effect for Rick Astley's voice, but that's the one we gave it.

And up above, there's grunting sounds. So that's the sound effect. So it's sound effect, sometimes called audio cue, is some sound in the video that's important enough that you would need to have a description of it so someone who was deaf or hard of hearing could understand the video well.

So what used to be the hardest part now-- and this doesn't always work, but the vast majority of times it does work-- is now the easiest part. So syncing up-- so how do you take those individually timed chunks, and then give them time codes? Well, all you really have to do is upload them to YouTube.

And so I'll just really quickly review how that process works in my browser window somewhere. My god, we're really, really, running short of time. I'm going to haul through the rest of the stuff. Where's my browser?

So I can click on Add New Subtitles in YouTube. And then I can click on Upload a File. And I can take my chunked file and upload it. And then I can just click on Set Timings. And it will

create the timings for me. And I will then get a nicely timed transcript, that, if we're lucky, it'll look something like this.

YouTube Tube also allows you to, now, you can manipulate the times for each of these chunks by just clicking down here. Actually, you have to go into Edit mode first. I get these drag handles, and so I can change the timings this way. And even down below, it's got a little sound wave file here. So you can use the sound wave file as kind of a visual way to identify when someone's starting to talk, and when they're stopping to talk, when there's silence, and not silence.

If YouTube, for whatever reason, won't work-- you've got copyrighted video and you can't upload it because of the flags when you try to upload it, you could use something like Movie Captioner or Magpie. Magpie anymore seems to me to be pretty crash-y. But both of those tools will allow you to import a caption, and then listen to the video, and then manually click a hot key that sets a time for each of your caption chunks.

OK, some words about audio description. So audio description is-- and this is a quote from the description key from the Digital Caption Media Project website-- "Audio description is a verbal depiction of key visual elements." So in a video, it's things like opening titles in on-screen text, screen changes, scene changes, costumes, character appearance, including the character's emotional state if it's important to understanding the video, and actions, and gestures.

So these are things that someone who is blind or with low vision might need to understand what's going on in the video-- audio description. In self produced video, the best plan is, if you can do it, to not have to do it-- so to create the video in such a way that you're describing enough of what's happening that you don't need a secondary audio track to do it. Typically, an audio or secondary video track-- that's the format those things are in.

An audio description file is a secondary audio track or a secondary video track. But it can also be a text track. It can be a text track that has timings, and is just in the format of SRT.

So you would have a text track that has your audio description, which you would say in an audio description, just in text format. And that gets associated with the video. For production of it, DIY'ers can use Audacity. It's a free program.

It will allow you to record either the entire thing, just kind of watch the video and put your audio description in there. Or better, record each individual piece of audio description, and then

string them together with silence, so that the end product is the same length as your video. That's what you need to get-- you need to get the audio file to be the same length as your video file, your [INAUDIBLE] audio file, your audio description file. There's also commercial products. Audition and Sound Forge are two of those.

Now, to glue the things together, you're going to have bits. So we're going to have your caption file, which has timings in it. And you're going to have your video, which is, if you can do it, pretty much the best format is to have MP4 H.264 encoded, and then a secondary audio format, MP3.

And I'm just going to show you what this process looks like to glue it together. I'm going to use this workflow. So I'm going to use mkvmerge GUI to mux or to multiplex all these files into a single mkv file. And then I'm going to use MkvToMp4 to convert it to an M4v file for upload.

All right, so this is mkvmerge. What I did is, I just took my video from my desktop, and I dragged it in here. And then I also put in the other things that I needed.

So I wanted, A, my audio description file, and I wanted my two caption files. And I just dragged them into this top window. They appear down here, and I can set properties on them.

So for example, in this particular video that I did, I have two caption files. I've got one in Spanish, and I've got one in English. So I can just click on this, and it allows me to set the language of that particular caption file, and/or to identify the language of it, and to give it a name.

I would click on Start Muxing after I get all those situated. So I've also got an audio description file in here, right there, that gets a track name of English with Audio Description. I click on Start Muxing, and it says, "Mux it," and that's it. And when I mux it I end up with this file here.

And this file right now-- we'll play it. I can play it inside of VLC, which is a common video player. And you can see when I start to play it-- I'm going to play much of this-- but you can see I have both an English audio track that's the main audio track. And then I have a track with audio descriptions embedded in it. And then in subtitles, I've got both my subtitles. I've got English captions and Spanish captions available.

So if I have VLC, I can just upload it from there. If I have a different target, so say I need to upload it to iTunes U, I'm going to need to go through a secondary process, and turn it from

that MPV file into its final form as an M4V file. And I am using mpvToMp4 to do that. And this allows me to set the output format, which I did over here.

I set the output format to-- somewhere up here, it allows me to set the output format to M4V rather than MP4. MP4 and M4V are very, very similar formats. M4V is the format that iTunes likes, but most other players will also take it in as well. I'm not going to do this process, but I can show you really quickly what the output of the process is. It's this file here, which I can then open up in, say, iTunes.

And when I open it up in iTunes, it's going to take up the whole screen, probably. No. Yes. I don't want a quick tour. There it is.

And you can see if open this up down here, I've got my English subtitles. And I've got two audio tracks. It lost the fact that this secondary audio track is an audio description track, but I've got the two audio tracks in here.

So the other way that you might want these files is in a format that has burned-in captions. And for burned-in captions, we use this tool, which is called Any Video Converter. You just drag your video file in there. You click on the Subtitle field, and navigate to where your subtitle is.

Make sure that you have the format chosen as HTML5 MP4 for output. And then when I click Convert, it will burn in the caption. So now the caption can't be closed.

But if I'm embedding the video inside of a PDF, I need the caption to be open, because by default, the video player inside of PDF doesn't have any ability to open and close the caption. It needs a burned-in caption.

Same thing with PowerPoint. There's a tool for PowerPoint called Stamp that allows you to have an open and closed caption. But it's a pretty buggy tool. So if I'm going to have a video inside of a PowerPoint-- I'm not redirecting to a video in YouTube or something, but I actually have it in the PowerPoint-- I'm probably going to use a video with a burned-in caption.

For the web, there's nothing else that you really need to do. You've got all the parts. You've got your video, your original video. And you have your secondary, or you have your caption file. You've got those things already. For burn-in, which is what I just showed you, we tend to use Any Video Converter-- seems to be the most reliable product. There's a Windows version

and a Mac version of it.

OK, so what goes where? It for web you need to or you should use an accessible player. So where do you find an accessible player? We have a matrix that's available here, and that I have in a browser window somewhere. There it is.

This is a matrix that we created for a different project that has a bunch of video players listed along the top, and tells you what features they support. So I was involved in the creation of Able Player, which is kind of like a Swiss Army knife of video players. It will do pretty much anything you want it to do.

And its goal is full accessibility. So it's fully keyboard accessible. You can add audio description as a separate track, or you can add audio description as text. It has a live transcript feature.

But there are number of them here. These are all things that can be found without having to pay for them. And I've got a number of them in here that I haven't completed the listings for yet. Hopefully, I'll have this matrix completed by the fall.

I hope I'm not leaving anything out. That's what I had at this point. And I didn't leave a whole lot of time for questions. But please ask them. Thank you.

LILY BOND:

Thank you so much, Ken. That was a fabulous presentation. And people have been asking lots of questions. So we'll get into the Q&A now.

I just wanted to mention a couple of our upcoming webinars on a quick start to captioning, and HTML5 video accessibility, both in August. And then in October, we have 10 Tips for Creating Accessible Online Course Content. You can register for those on our website.

So Ken, to start out, I'm going to ask you a question that came in. How do you deal with turnover with your student transcriptionists?

KEN PETRI:

Yeah, what we try to do-- that's probably the thing that we have the biggest difficulty dealing with. Not only do you get turnover, but you also get students who are flaky, unreliable. You have deadlines that you need to meet, because the clients have solid deadlines they have to meet.

But the students not always-- they don't always care. They're students. They've got other things on their minds.

So what we tend to do is have some in reserve. So when students come to us, we tell them at the outset that we can't employ them more than, initially, 10 hours a week. And they usually want more than that, but we say we can't employ them more than 10 hours a week.

So we can sort of employ more students and have kind of a buffer. So if the workload goes over 10 hours per week, we can ask them if they'd like more hours. And typically, they will.

That seems to be pretty effective. One other thing that we do is, we lend them computers. And we tell them that they can use these computers for their regular work, and they can just basically have them.

So there's a huge incentive to kind of stay, because as soon as they leave, the computer goes back to us. And that's kind of a bribe. But it works OK.

LILY BOND: Thank you. Another question here-- there's actually a couple questions about kind of who pays for this to happen, and if you have a minimum amount that is charged for each video, and that kind of question about the funding.

KEN PETRI: Yeah, so a lot of the stuff that we did initially, after our grant funding ran out, we did for projects that were research projects, where they had a grant, and they were paying for it out of their grant monies. Some departments-- so for while, chemistry had us on retainer, and we were doing a number of videos that they were putting online. And the Chemistry Department paid individually for that.

We don't have-- there's really no central funding on campus for captioning and transcription. I think ultimately, that's going to be a big liability. We need to find a way to fund some of that centrally. But at this point, there isn't an avenue for it. So it's come mostly out of people's pockets, out of their research funds or out of departmental monies.

LILY BOND: Great, thank you. So the tools that you were talking about seem to be Windows-centric. And people are wondering what the options are for Mac, and whether it can be just as easy of a workflow on Mac.

KEN PETRI: Yes, so the only tool that's not in there, that there's just no kind of equivalent tool that I've seen for Mac is Subtitle Edit. And it's a nice tool to have, but it's not it's not necessarily an absolutely crucial piece of the workflow.

The mkvmerge GUI is now a product called MKVToolNIX, or something like that. And there's

GUI. There's graphical interfaces for both Mac and PC now for that tool.

Express Scribe is a Mac and PC program. Any Video Converter is Mac and PC. Of course, YouTube is just a browser.

So most of those tools are available on both platforms, as is Movie Captioner. So if you're a fan of Movie Captioner-- I don't know how many people have used it-- but it's a full-blown captioning tool, but it also allows you to do timings. Movie Captioner is both Mac and PC.

An advantage to using the Mac is, if you're doing a lot of stuff for iTunes U in particular, iSubtitle is worth looking at. iSubtitle allows you to take a video in a caption file or multiple caption files, and merge them into M4V files suitable for upload to YouTube. And it's a Mac-only product.

So I mean, we work mostly on Windows. All the students get Windows computers. But they're definitely almost equivalent workflows, minus maybe one tool for the Macintosh.

LILY BOND:

Great, thank you. That's really helpful. There are a lot of questions about the implications of copyright law on YouTube captions. I don't think we have the time to get into that right now. But we did do a webinar on copyright fair use and YouTube that we will paste a link in the chat window to, which can be helpful for the people asking about copyright law.

KEN PETRI:

So I can just say one thing-- that we've seen-- if you upload something that's like a sports broadcast, immediately after you upload it, it'll be flagged. And you'll get a takedown, or basically what they do is, they leave it in your account, but it's not visible. It's visible to you as a kind of "you can't play this video" thing.

But a lot of stuff that you would think would be copyrighted is, in fact, not flagged by YouTube when you upload it. So what we do, if we have a situation where we know at least theoretically something's going to be copyrighted, we'll upload it to a private account, and not publish it at all, just so it's only available to that person, use YouTube for getting the caption file, and then immediately take the video down, and then use the raw video that we first got, and add the caption file into it, and then put it into the course management system. And I mean, that might be skirting the letter of copyright law. But inside of a university situation, I think you could probably chalk that up to fair use.

Now, the one liability is, if you start uploading a lot of stuff that has copyright protection in YouTube, they'll turn, if you've had your account time extended-- and there's a link, actually,

on our website. So if you go to the DIY Caption website, go to OSU.edu/diy-captioning, there's a link in here to how to get extended time in YouTube. So if you're still stuck at, like, 15 minutes, you can make a request.

And if you've been a YouTube user for a while, and you don't have any egregious copyright violations, they'll extend your time. And I can upload two-hour videos to YouTube without any issue now. So just FYI-- if you do start uploading captioned stuff, they may take your account time privileges away for you temporarily.

LILY BOND: Thanks, Ken. I think we probably have time for a couple more questions here. If anyone has to go, the Q&A session will be included in the recording. So if you missed the last couple of questions, you can watch those later. So Ken, people are wondering how much you pay the students, and where the funding for the students comes from.

KEN PETRI: OK, so the funding comes out of my budget initially. We pay them \$10 an hour. The funding comes out of-- which is sort of like campus going rate. It's kind of highway robbery, but it's the campus going rate.

We pay them \$10 an hour. And we scale it on work completed, rather than the time that it actually took them. So we just say, look, you got a video. If it's an hour-long video, it's going to take you five hours to do it. You can bill us for five hours of your time, whether or not it takes-- because we have no way of monitoring it. So we say, that's what you get.

If they go over that, they can't go for more than that. If they go under it, they can bill for it. So there's an incentive to do the work quickly.

It initially comes out of my budget. And then we're a vendor inside of OSU system, and so when somebody makes a request, it's a purchase order. And so we may be running a little bit in the red temporarily, but then the third requesting unit will pay for it, and it fills back up my account.

LILY BOND: Great, thank you. And so a final question here-- people are wondering what transcription conventions you use. Whether you kind of developed your own conventions, or whether you have, like--

KEN PETRI: I mean, I don't know about transcription conventions per se, but captioning conventions-- most of the stuff is a version of this site here. So it's the DCMP-- described and captioned media

program. It's just DCMP.org.

And they have a thing called Captioning Key that has best practices for captioning. They also have a section of it that's called Description Key, which has best practices for described audio, too, for audio description. And it has full guidelines and tons of examples.

And we tried to take what's useful from those examples. I can't say we are 100% in conformance. But they're not really written as laws, anyway. They're kind of like, these are some best practices. And so we've kind of evolved ours from theirs.

LILY BOND:

That sounds great. I'm sure that resource will be helpful to people. So I think that's about the time that we have.

I want to remind everyone that we will be emailing out a link to view the recording as well as the slide deck, for those who were looking for that-- most likely tomorrow, so keep an eye out for that email. And thank you, everyone, for joining. And Ken, thank you so much for a wonderful presentation. It was highly appreciated by everyone.

KEN PETRI:

Well, thanks for letting me do it. I hope people benefited.