Encoding Best Practices

***Questions and Answers from
Streaming Media's Encoding Best Practices webinar***

INDEX OF QUESTIONS

General encoding & codec questions

Encoding for streaming questions

1) [Can we encode one good quality file for web and allow the streaming network server software/hardware to determine the delivery rate based on the visitor's bandwidth connection? Do we need to deliver multiple bit rates for lower (standard) and higher bandwidth (HD) visitor connections?](#)

8) [Can you talk more about http live streaming?](#)

13) [It was said that Constant Bit Rate (CBR) should be used for mobile streaming, how does Variable Bit Rate (VBR) video and/or VBR audio cause issues when streaming to mobile devices?](#)

14) [Do you know of any scalable solutions for large quantities of smooth streaming compression?](#)

17) [What encoding settings are best for live streaming? I use Flash, VP6 350k 12 fps, de-interlaced](#)

25) [What's the best way to support live streaming to Apple mobile devices right now, with current technology and standards?](#)

26) [Are there encoding products available that produce multiple formats at the same time - I'm thinking live event here, and I need to support Windows desktops, Mac desktops, Apple mobile devices, and Non-Apple mobile devices.](#)

28) [Is Windows Media Encoder the only program that is available for us for live streaming our (church) services?](#)

34) [Do the rules for Variable Bit Rate (VBR) and Constant Bit Rate (CBR) also apply for mobile media streaming?](#)

Encoding for web and mobile questions

5) [What are recommended bitrates for file delivery over the Internet and to mobile devices? Is there a general range, say 300kbps, 600kbps, 1Mbps, etc.?](#)

33) [What is the difference in encoding H.264 for web and mobile and how do I encode for both?](#)

General/non-encoding questions

21) [What is a good/easy software to use to edit videos?](#)

29) [What is the best way to measure video quality - subjectively besides using A/B testing or MOS?](#)

1) Can we encode one good quality file for web and allow the streaming network server software/hardware to determine the delivery rate based on the visitor's bandwidth connection? Do we need to deliver multiple bit rates for lower (standard) and higher bandwidth (HD) visitor connections?

Generally one would want to encode to multiple bit rates. Keep in mind that VOD solutions often allow the user to select the bit rate and are often Variable Bit Rate (VBR) encodes.

In the case of streaming servers, new protocols are in various stages of deployment - Microsoft Smooth Streaming, Adobe Dynamic Streaming, Apple HTTP Streaming - which involve server polling of the "client" (viewer/player) to dynamically switch the file packets being sent. Older technology is often a single polling at the start of the stream. The newer dynamic protocols may need GOP (Group of pictures) alignment between files. Generally files for streaming servers should be Constant Bit Rate (CBR). In all these cases multiple bit rates should be available for distribution. There are some Content Delivery Networks (CDNs) that use technology that can create and distribute multiple files from a single source.

2) How do Hollywood studios get such good quality at very high compression? For example, how do they get 8GB on a 4.7 GB disk at HD quality?

The quality of the source is a major factor, as noise found in many lower quality sources is the enemy of good compression. Codecs tend to preserve noise since they perceive it as "action." Also expert compressionists will test "challenging" sections (fast action, heavy motion FX, subtle gradations) to ensure their settings will handle such sections. In addition the quality of the compression algorithm, and more generally the compression program's exposure of the potential settings, allows for lots of user customization to the specific issues each source may have.

Episode has excellent user control of the codec settings and, additionally, excellent processing filters that can handle many of the issues found in "less than ideal" source material. Generally we'd recommend testing two minute areas - give or take - to give a 2-pass Variable Bit Rate (VBR) encode time to do a proper "look ahead" pass to see how it's allocating the bits. Considering their sources may come from files as big as 4:4:4 Uncompressed, they may have files many times larger than 8GB to fit on that 4.7GB DVD Video disk. Actually for DVD Video they'd more likely be using 8.5GB DVD. HD would be on up to 50GB Blu-ray. BTW, DVD Video is Standard Def. HD quality would be on Blu-ray.

3) **Will the royalties for H.264 Baseline Profile change after 2016?**

MPEG-LA, which is a consortium, will determine licensing royalties. So far it seems that thresholds may involve the number of views and they are set very high so that only "very high volume" distributors will be impacted. Also whether the distributor is charging for its services may have impact. This is, of course, is subject to change. We've seen no differentiation over profile but there may be differentiation between internet and non- internet distribution.

Here's the link to MPEG-LA's own explanation and there may (or may not) be changes when the 2015 term expires.
http://www.mpegla.com/main/programs/avc/Documents/AVC_TermsSummary.pdf

This site has a very good Q&A on typical questions.
http://www.betanews.com/article/10-questions-for-MPEG-LA-on-H264/1274306999

4) Hardware vs. software encoding: are there differences in quality or other advantages?

Quality is primarily dependent on codec implementation and processing filters.
Hardware gives you a dedicated device with a built-in feature set which may not tie up other equipment. Software encoding is very much hardware dependent as well although hardware is allocated based on the computer or network's access to processors, memory, and network configuration.

While this can be complex in some cases, Episode 6 simplifies this greatly yet maintains extensive configuration flexibility. Whereas hardware may be a high quality, very fast "locked box," Episode 6 allows you to add both Mac and Windows computers, as well as submission from computers not part of the network cluster, extensive control over settings implementations, processing filters to handle the challenges unique to specific sources or add watermarks, time code windows, levels adjustment just to name a few. Imagine the scalability ranging from your single desktop to using the resources of your ever-expanding network at your facility. We might say that unlike a dedicated hardware encoder, Episode gives you near infinite hardware and software scalability and a simple interface to manage it all.

5) **What are recommended bitrates for file delivery over the Internet and to mobile devices? Is there a general range, say 300kbps, 600kbps, 1Mbps, etc.?**

It depends on your target market. Lower speeds may reach the broadest market if it includes people who are still on slower ADSL lines or Satellite Broadband. That can range from rural areas to commonly underserved communities. It can vary by city and country depending on their infrastructure. If you're targeting a product or service to major metropolitan areas that are in higher income brackets that might allow you to use a fairly high bit rate for example.

Targeting mobile phones may well depend on the handset and the providers. What an iPhone 4 can handle on a network that can handle high data rates may be different than what an older Nokia handset can handle. In fact, when it comes to mobile devices, some big metro areas may require lower speeds because of network saturation. This is why it's hard to give specific guides.

The clearest thing I can suggest is to know your target, know who you're willing to include, what the ROI is on offering a number of different bit rates vs. achieving your goal.

Some people actually use YouTube encodes as a guide since Google endeavors to reach a wide audience. You can download and examine the bit rate for their 240, 360, 480, 720, 1080 encodes as well as those that target the iPhone. Note that over time YouTube changes its encoding methods as the market evolves.

6) Is there a reference book, URL, or other resource that helps define all of these buzz words used in encoding?

In this fast changing technology, books become dated quickly. I find this site the best to learn about encoding.
http://www.streaminglearningcenter.com/

Jan Ozer, who runs the site, has written about Episode.
Producing H.264 files for Flash distribution with Telestream Episode Pro
http://www.streaminglearningcenter.com/articles/producing-h264-files-for-flash-distribution-with-telestream-episode-pro.html

Telestream also has a great guide with Episode specific suggestions written by Intelligent Assistant
http://www.telestream.net/pdfs/user-guides/Simple-Encoding-Recipes.pdf

7) We have all heard about problems of streaming video online for a variety of heterogeneous devices, displays, networks, speeds, qualities, frame rates, manufacturers, etc! Why is there a disinterest in the industry in using the H.264 SVC encoding, as it brings solutions for the problems above? I am PhD student working on SVC and I have personally seen them to perform better that simulcasting. I believe this is the answer to the problems.

SVC holds great promise for adaptive bit rate streaming. The key advantage, as I suspect you know, is the ability to use a single source file to derive and deliver the other bit rates as needed. SVC is still very early in the process of being adopted. Basically it's a case of interoperability. It's seems that its biggest and earliest inroads will be in video conferencing. Vidyo is using it. Google Video Chat as well. We've

heard speculation that this may be part of Apple's FaceTime or, at least, would be a logical part of its road map. Ultimately there has to be server and player support for it and this may come over time. Keep in mind that use is often driven by market forces with each aspect, Encoder, Server, Player needing to support it and have wide distribution in order to make the ROI on implementation viable. For example, if Flash Player doesn't have it on their short term roadmap that can impact its support in other parts of the market.

8) Can you talk more about http live streaming?
In relation to Telestream products you can use Wirecast to send RTSP to Wowza 2 server, which then uses the source for HTTP streaming to iOS.
See this document for details
http://www.wowzamedia.com/resources/tutorial_ffd.pdf

Of course one of the promises of HTTP streaming is that because it's sending small packets (chunks) of information using HTTP, no special streaming server would be needed. The stream is presented as chunks at different data rates that come from an M3U playlist. Hence its dynamic nature as the viewer's conditions change. Another advantage is that firewalls or proxy servers that let through HTTP will allow the stream through.

9) What are your recommendations for formats to use for general viewing? (Our clients generally ask for WMV files because their customers view the videos on PCs.) Do you offer software for this purpose; if so what is it and how much does it cost?
I'd recommend Episode. Episode 6 will be scalable from single desktop to a workgroup cluster to a dedicated server farm. Even a small post facility with a few Macs and Windows computers can be clustered for great speed advantages.

These days, with rare exception I'd recommend H.264 .mp4. It works in Flash, Silverlight, Quicktime, HTML5 and even Windows Media Player 12 on Windows 7 and, with the proper profile and specs, can be viewed on Apple iOS devices. H.264 is an extremely efficient codec (quality relative to bit rate). The upcoming Internet Explorer 9 will be supporting HTML5 H.264. Microsoft's own Expressions Encoder supports encoding to H.264. The problem with WMV is that if the material is to have any longevity, a business will eventually face the cost of re-encoding their entire library. Unless a business has explicitly banned the use of the Flash browser plugin, Microsoft Silverlight, Apple Quicktime and has not updated to Windows 7 and is willing to risk the costs of having to re-encode everything in the not-too-distant future, I see no other reason to use WMV. Generally WMV is best used if the file only has a short shelf life and IT has restricted Browser plugins and OS and Browser updates.

The current version of Episode 5 and the soon to be released Episode 6 support both WMV and H.264 of course and you'd even be able to offer the option of doing both a WMV and H.264 encode.

10) Do you have recommendations for sources that provide a general overview of encoding topics for novice and intermediates?
In this fast changing technology books become dated quickly. I find this site the best to learn about encoding.
http://www.streaminglearningcenter.com/

Jan Ozer, who runs the site, has written about Episode.
Producing H.264 files for Flash distribution with Telestream Episode Pro
http://www.streaminglearningcenter.com/articles/producing-h264-files-for-flash-distribution-with-telestream-episode-pro.html

Telestream has this great guide with Episode specific suggestions written by Intelligent Assistant
http://www.telestream.net/pdfs/user-guides/Simple-Encoding-Recipes.pdf

If you're also interested in live streaming (which is live encoding of course) as well as encoding, this is a good site but it may be difficult to find basic info.
http://www.streamingmedia.com/

Telestream has a great live streaming product called Wirecast. While the link below is to an Apple seminar which I conducted, the product is also available for Windows.
http://www.seminars.apple.com/seminarsonline/videoliveweb/apple/index.html

and Telestream has a great Wirecast blog which is perfect for novices and intermediate level.
http://blogs.telestream.net/wirecast/

11) What is a common bitrate and resolution for an enterprise environment?

It may depend what your intent is. If it's encoding, it really depends on your audience targets. Episode 6 will be scalable to your network as far as workload and speed. Generally higher quality encodes may use more resources depending on the codec.

If it's viewing that may well depend on the constraints given you by your IT people. Their concerns may be load (number of viewers for example) and available bandwidth (other network use at the same time). Ideally a network may have no problem supporting HD frame sizes of reasonably high bit rate H.264. People do this at home with iTunes for example. Alas, in an enterprise environment only your IT specialist knows for sure. The challenge as a compressionist is to get the best possible looking file given the IT department's constraints. To do that requires a tool that allows you lots of control over the settings and processing filters and takes advantage of your network's resources to hit that target quickly. What's "common" is what IT permits. Episode would certainly be that tool.

If you still want a sense of "common" I'd recommend downloading YouTube videos for 240, 360, 480, 720, 1080 and you'll see what bitrates they're using. Since YouTube is the most commonly viewed video site on the Internet it often influences the level of quality people expect at a given frame size. Then you have a "target guide" depending on any limits IT gives you.

12) Is it possible to restore stored H.264 clips? If so, why is this not widely used in the broadcast world for long term storage or archiving?

If you mean, is it possible to rebuild a clip to source/master quality or something close to that, the answer would be no. H.264 is a lossy codec. Data thrown away cannot be retrieved. It's certainly possible to encode to near visually lossless for delivery or even, in some cases, as a source for further targeted compression. Each round of compression compounds the loss.

Certainly H.264 is being used for acquisition with AVCHD and AVC Intra and those can be stored as masters. Even those won't match 4:4:4 uncompressed coming from a camera such as a Sony F-35 though.

The best archives maintain the full integrity without any further loss of data. Even in the case of visually lossless or near lossless codecs, when the source is pushed further in compositing or color grading artifacts can be introduced. Therefore most broadcasters (and feature film makers) want the archive to be at least at the quality of the acquisition or final "treated" master used as the source to make the deliverables.

13) It was said that Constant Bit Rate (CBR) should be used for mobile streaming, how does Variable Bit Rate (VBR) video and/or VBR audio cause issues when streaming to mobile devices?

Streams need to be at a constant bit rate (CBR) for reliable streaming. VBR (variable bit rate) may have peaks which can interrupt the stream due to the fact it demands a higher data rate or more data in the buffer/cache (time needed to fill the buffer) at a given moment. The viewer might see stops in playback as a result. In fact streaming servers generally want CBR to control data sending (near exact amount of data sent is always the same at a given moment in time).

This tutorial from Jan Ozer gives a good overview. Note that the settings screenshots are from Episode.
http://www.streaminglearningcenter.com/articles/streaming-102-codecs-plus-vbr-and-cbr-and-i-b-an-p-frames.html

Note his comment on CBR vs. VBR specific to cell phones and under what case, generally VBR is used (progressive download).
When is this a problem? Well, if you were producing video for a tight, relatively low bit rate connection like a cell phone, this variability could interrupt playback. Also, when distributing via a streaming server, which meters out the video to the remote viewers as needed, the consistent data rate of a CBR encoded file is easier to administrate. For this reason, the generally accepted best practice is to use CBR when producing for streaming delivery, and VBR when producing for progressive download.

14) Do you know of any scalable solutions for large quantities of smooth streaming compression?
It depends on what you mean by smooth streaming. Smooth Streaming is a Microsoft technology (http://www.iis.net/download/SmoothStreaming) in which the server sends small fragments drawing from multiple source videos so that the data rate changes depending on the changes in bandwidth on the client/viewer side. More broadly this is referred to as adaptive bit rate streaming. That packetizing is usually happening with a server side utility.

If you're looking for a deeper understanding of Microsoft's Smooth Streaming you can get an overview here
http://www.microsoft.com/downloads/details.aspx?displaylang=en&FamilyID=03d22583-3ed6-44da-8464-b1b4b5ca7520

Episode is immensely scalable from a single desktop to large enterprise network configurations. With Episode you'd certainly be able to do multiple encodes of a single source needed to create files for such streaming. Depending on the size of the network cluster, encoding can be faster than real time and/or you can queue multiple sources with multiple encodes each.

If you simply mean files that stream smoothly from the various streaming servers such as Flash or Windows Media servers HTTP streaming (which also involves packetizing) or even progressive download, the same scalability holds. In fact, with Episode, you can target codecs and settings to multiple streaming and progressive download environments. The key to getting smooth playback depends on matching appropriate settings to the capabilities of your target audience.

15) How do you see HTML5 affecting me as an encoder and how it will affect the end consumers' applications?
That depends. There's a bit of a codec war over what HTML5 will support. For the most part, H.264 holds the vast majority. This means if you're already encoding H.264 .mp4 for Flash, Silverlight, QuickTime, or WMP12 there may be no effect at all. The same files would work in HTML5. It will be up to the web programmers to implement the video tag. You'll note that both Vimeo and YouTube already allow you to view their sites with HTML5 compatible browsers. Internet Explorer 9 will be implementing HTML5 H.264. iOS devices (iPod, iPhone, iPad) all use HTML5 H.264. The same H.264 file works in Flash as I mentioned above so the codec can cover all the bases as they currently exist.

The one holdout is Mozilla-sbased browsers such as FireFox. They want HTML5 with an open source codec such as VP8 WebM. That would have a major impact as all source media would have to be re-encoded. The market may not respond well to that.

In either case Telestream Episode will have you covered as it currently supports H.264 in .mp4, .mov and the .f4v container (amongst others). In addition Episode will be supporting WebM so if you are faced with having to re-encode your sources, Episode will have you covered.

HTML5's impact on consumer applications is a bit more complex. Flash still has many features that are well developed, such as interactivity, that are still being developed on the HTML5 side. There are many companies that are starting to support both for video playback on their websites. Interactivity will be the HTML5 challenge. We're hearing there are startups that are beginning to develop HTML5 interactivity and will likely target the growing iOS market first. SproutCore, an open source framework used in Apple's MobileMe web apps may be the catalyst for such companies. It's very early on this front.

16) Do you have a product that will work cross platform, for example H.264 (+) and HTML, etc.
Telestream Episode can encode to H.264 with all common containers and it will be supporting WebM as well if that demand should come to pass.

17) What encoding settings are best for live streaming? I use Flash, VP6 350k 12 fps, de-interlaced
Your encoder settings should be based on your target audience's bandwidth and computer, your upload bandwidth, and your content (talking head vs. fast action sports, for example).

If you are targeting very old computers (maybe 5 years old or more), that haven't updated their Flash plugin since 2007 (when it became H.264 compatible), are doing a talking head interview (where low frame rate won't hurt low temporal resolution and you maintain some quality) and viewers are on slow DSL modems, used a small frame size such as 320x240, then the settings you are using would be fine.

On the other hand, your viewers with modern computers, standard DSL speeds or cable modems that can easily decode the more efficient H.264 codec, might be concerned about the small frame size, possible loss of quality during transitions or other fast motion, less than fluid motion.

To put it simply, your settings are best for people on very old computers who don't update their Flash plugins and have very slow broadband connections. Is that your target market?

These days it's rare (except in some rural areas) to find internet downstream speeds below 768kbps on DSL and many support speeds of 7000kbps give or take. Cable modems often start at 5000kbps and many support 15,000kbps or more as well as those with FIOS service.

Of course it's possible your upstream is limited as many providers give you much lower upload speeds than download speeds. That would be a severe limitation on your end and I would look for a better service provider package if streaming is integral to your work (or even your hobby).

These days I'd consider a base setting with codec H.264, 500-800kbps upload, 24-30fps, frame size 480x360 - 640x480. The exact settings within the range depends on content and bandwidth limits.

Of course maybe you need to reach a range of viewers with old computers with low bandwidth to the latest computers with high bandwidth for HD streaming.

Telestream has and excellent live streaming encoder called Wirecast. (http://www.telestream.net/wire-cast/overview.htm)

As an encoder it can do multiple simultaneous encodes at different bit rates so you can easily have low, medium, high, HD settings. You can even encode to different server types at the same time such as Flash, QuickTime, Windows Media (on the Windows version of Wirecast). Wirecast supports H.264 currently and they'll be adding VP6 support when 4.0 is released. (http://www.telestream.net/wire-cast/new.htm)

Not only is Wirecast an extremely flexible encoder, it's a production switch in software. It's capable of handling multiple cameras and multiple types including webcam, DV, HDV or higher end cameras. It can even take feeds from the desktops of other computers on the local network as "camera" sources. It can

also do chroma key, titles, 3 source picture-in-picture over backgrounds, and can handle multiple audio sources.

Of course the power of the software is only constrained by the power of your computer. An 8 core desktop will handle more than an entry level dual core laptop but even a simple laptop and webcam can be quite powerful when you add your own titles and do a picture in picture of you and the program you're talking about with a desktop presenter feed.

18) What are the best practices in encoding to a single media source can be delivered to multiple platforms. (iPhone/iPad, Android, web, etc)

Currently the closest there is to a "universal" codec is H.264. The closest to a "universal" container for that codec is .mp4. It'll work in Flash, Silverlight, QuickTime 7, Windows Media Player 12, HTML5. With the HTML5 embed tag it can be delivered to iOS devices and Android as well. You'd probably want to limit profile to Main Level 3.1 but some devices may need Baseline Level 3.0 or lower (iPod Classic for example). You'd also probably want to limit frame size to 640x480 and data rate below 1.5Mbps. Android OS/Handset device compatibility may depend on your lowest common denominator (OS 1.6 maybe) and oldest handsets. Ultimately you'll have to decide what the lowest common denominator is but the above info should head you in the general direction.

Telestream Episode has presets that you could use or otherwise customize to hit everything in your target. If you ended up deciding you needed to do multiple encodes that too would be easy with Episode. You can apply multiple target settings to a single source file.

19) Why is it that sometimes I can scrub through the encoded video but other times I can't? What component of the encoding process affects the ability to scrub?

Often this is due to "I frame" (key frame placement). In Episode one can use "Force Key frames" to ensure a minimum key frame distance for scrubbing.

Note that B frames (bi directional frames) are the most difficult to construct as you scrub, so decreasing the number of B frames can also improve the experience. Keep in mind that increasing I frames and/or reducing B frames can impact quality. More I frames means more data allocated to those frames than others. B frames tend to be most efficient regarding data use so reducing those also may mean a higher data may be needed to maintain quality.

20) What do you think about the Matroska (MKV) container?

A good place for general information (http://www.matroska.org/technical/guides/faq/index.html)

While it can have advantages over AVI and, with its open source nature it may be a good choice for VP8/WebM, a critical issue is compatible players. Macs, for example, may be limited to VLC mplayer or the Perian QuickTime plugin and many users may not be aware of or inclined to go through the effort to install it. Of course this may change over time depending on the adoption of VP8 so it really depends on your viewer target.

21) What is a good/easy software to use to edit videos?

That really depends on your intended goals and your current and future skills. Some people are quite happy with Microsoft Windows Movie Maker and Apple iMovie. The latter has a nice feature set for beginners. Others look to Sony Vegas, Adobe Premiere, Avid Media Composer, Apple Final Cut Express and Final Cut Pro. They're all good and "easy" for the skilled operator as workflow is important to any professional. Right now Apple Final Cut series seems to have the widest use but the others are certainly competitive and may be suited to your specific workflow.

22) What is a recommended way to convert video to other formats like AVI, MPG, etc?

I'd recommend Telestream Episode available for both Mac and Windows. It can handle nearly all major input and output formats including those destined for Broadcast or other "high end" use.

23) What software would you recommend to compress videos to a smaller size? One hour of HD is just slightly bigger than a DVD 4.7GB.

See above. One hour of HD, depending on the source, can be many times bigger than a DVD. DVD Video is standard def so the result inherently won't be HD. HD would be Blu-ray at 25GB or 50GB capacity.

24) Where can we find a matrix of the compatible codecs?

"Compatibility is in the eye of the player" to use a paraphrase of a common expression. Players, browser plugins and codecs are updated frequently enough that such a matrix would be dated very quickly. Many players also have plugins which themselves are updated. The question you must ask as a compressionist is: What players and plugins does your potential viewer have and how frequently do they update? You need to follow industry trends.
Currently H.264 codec in .mp4 wrapper probably has the widest compatibility but there are specs such as Profile, Level, data rate, frame size and encoding methods that can have impact on that though.

25) What's the best way to support live streaming to Apple mobile devices right now, with current technology and standards?

One good method is to use Telestream Wirecast to stream to a Wowza server. Properly set up, the server will be able to stream to both Flash and iOS clients simultaneously. Wowza has posted a tutorial online for this.
http://www.wowzamedia.com/resources/tutorial_ffd.pdf

26) Are there encoding products available that produce multiple formats at the same time - I'm thinking live event here, and I need to support Windows desktops, Mac desktops, Apple mobile devices, and Non-Apple mobile devices. That is, right now I use two computers to produce a Flash stream and a Windows stream. Wondering if I could get away with one computer and one encoding product that "does it all." Live is key.

Telestream Wirecast can stream both QuickTime and Flash and, on the Windows version, Windows Media. It can stream to any or all simultaneously. It can stream multiple data rates as well and record to hard drive. The number of streams and the quality are dependent primarily on the power of your computer. Basically an 8 or new 12 core computer will handle more than a 2 core for example. It's very easy to produce a Flash and Windows stream on one computer with Windows version of Wirecast.

27) Does Telestream have any plans for Wirecast support of WebM?

Currently Wirecast 4 is adding VP6 support as well as moving from Apple to MainConcept H.264. In order from WebM to warrant the R&D testing and implementation, we'd need to see the market support for it in streaming servers and players. That time may come but we're still observing the market implementation and demand.

28) Is Windows Media Encoder the only program that is available for us for live streaming our (church) services?

I can see you are using Windows Media to live stream. Mac users can expand their screen using Windows Media Player, but Microsoft stopped development for it on Mac over half a decade ago and it is not native to modern Intel Macs. While Telestream makes Flip4Mac for WMV playback in QuickTime, I think you'd reach a wider audience with Flash.

With Telestream Wirecast you can stream Flash as well as WMV and can even do it simultaneously if you'd like. You'd also get software video switcher features so you could setup multiple cameras if you'd like.

29) What is the best way to measure video quality - subjectively besides using A/B testing or MOS?

Quality is not only objective measurements but subjective evaluation. There are various tools on the market that can take measurements. Some are quite expensive. If you're comparing two results I think

A/B testing is best. Keep in mind viewers aren't likely doing A/B tests so ultimately you'll have to judge by reaching the best quality given the targeted audience.

30) Is .mp4 a container? I understand .flv is a container and YouTube is Flash-based. How is Apple iPhone supporting YouTube if they don't support Flash?

.mp4 as well as .mov work in Flash. Flash is extension agnostic when it comes to H.264. Flash has supported H.264 since December 2007. YouTube is using H.264 for all its encodes. H.264 with HTML5 embed tag works on iOS and these is what YouTube is doing. Since the same codec works in Flash and HTML5, YouTube need only use one codec. For example, H.264.mp4 will work with both Flash and HTML5.

31) What is the difference between high profile, baseline and main profiles? Is it advisable to transcode between these profiles?

This chart lists H.264 profiles and if you scroll down just a bit you'll see the features they support
http://en.wikipedia.org/wiki/H.264/MPEG-4_AVC#Profiles

Higher profiles generally demand more resources to decode than lower ones. Various devices have limits. Some people are using High Profile for web encodes now and others keep to Main. Some smartphones will work with Main profile but some older or less powerful ones require Baseline.

Telestream Episode allows you to choose the profile. This is not the case with all encoders. In Episode, changing the profile will change some of the settings available to you can see in the above charts, certain features are available only at above a given profile.

32) What is the best option for de-interlacing?

Telestream Episode has one of the most powerful, user configurable de-interlacers in any encoder today. The reason for that is there is no universal "best option." Often de-interlacing is dependent on content.

Many de-interlacers either duplicate the dominant field, fast but low quality since you're throwing out half the information or, blend both fields, which, while ok for motion but is less sharp.

A good de-interlacer should have options for Edge Detection which can yield sharp edges, and Motion Compensation which results in sharper edges than blending but retains smooth motion. Basically this means depending on the edges of the subject matter and the motion involved.

A good de-interlacer should also detect which frames are interlaced and which are progressive when you have mixed sources. Another feature of a good de-interlacer is the ability to detect and de-interlace moving areas where interlacing is most visible during motion. It should also include the ability to set a threshold if needed.

As you can see, a flexible and powerful de-interlacer should let you, the compressionist run tests and make choices based on the content of the material. Episode gives you that kind of de-interlacing feature set.

33) What is the difference in encoding H.264 for web and mobile and how do I encode for both?

Mobile files generally have greater restrictions given the hardware limitations. One challenge is that different mobile devices have different restrictions.

Often a single file can be used for both if it falls within these restrictions. For example: the iPhone 4 supports H.264, up to 720p, up to 30fps, up to 1.5mbps, Main Profile 3.1. A file matching these specs would also play on the web. On the other hand if you used High Profile it would still play on the web but not the iPhone 4. Of course other handsets may have different specs.

Telestream Episode has a wide variety of presets that would work on various mobile handsets. In addition you can easily customize them. With Episode, if you need to, you can encode for multiple handsets and for the web as well if you wanted to target each to the maximum of their playback ability without forcing other handsets or the web video to the "lowest common denominator" limits of the weakest handset. This is why knowing your target is key.

34) Do the rules for Variable Bit Rate (VBR) and Constant Bit Rate (CBR) also apply for mobile media streaming?

Yes the same rules apply although perhaps even more conservatively. With rare exception you'd want to use CBR for streaming. Of course if you mean progressive download you could use VBR but the caveat is that with variations in connectivity speed and stability, high peaks are much more likely to create buffering or stalling. To be conservative you may want to stick with CBR in most cases.

35) What's the best codec for high quality Blu-Ray encoding?

Blu-ray can use VC-1, MPEG-2, or H.264. H.264 is the most efficient codec. It can achieve higher quality than comparable bit rates of the others.