EXHIBIT 29

Video Compression: A Codec Primer

by T. Jay Fowler

A couple of years ago, while working at the University of Minnesota's Digital Media Center, a professor approached me with a videotape in one hand and a floppy disk in the other and said, "Transfer this footage to the disk. I'll be back in a half-hour."

Her less-than-dazzling grasp of video technology nearly made me fall off my chair, but as digital-video capture and editing became my specialty, I realized that many people are unfamiliar with the time required to capture and compress video, and the large file sizes required to store it.

The truth is that the capture and compression process is slow and cumbersome, and raw, captured video is huge. How huge? Ten seconds of raw, uncompressed NTSC video (which is the standard for television) will fill as much as 300 MB of storage space. That would be more than 200 of my professor's disks, just for 10 seconds.

So how do people create video small enough to play over the Web? In the past few years, there have been big advancements in the development of *codecs*, which allow video to be reduced to a reasonable file size for use on CD-ROMs and for Web delivery.

Codec is an abbreviation for compression/decompression. A codec can be either a software application or a piece of hardware that processes video through complex algorithms, which compress the file and then decompress it for playback. Unlike other kinds of file-compression packages that require you to decompress a file before viewing, video codecs decompress the video on the fly, allowing the client to view the file from its compressed original.

We will take a brief look at the main ingredients of video compression, including the difference between compression schemes, hardware and software codecs, video-compressing freeware that works just fine, and architectures (which are like all-in-one packages that handle the entire compression and decompression process).

Codec Compression Schemes

Codecs work in two ways - using **temporal** and **spatial** compression. Both schemes generally work with "lossy" compression, which means information that is redundant or unnoticeable to the viewer gets discarded (and hence is *not* retrievable).

Temporal compression

This method of compression looks for information that is not necessary for continuity to the human eye or ear (remember that videotape plays back sound as well as pictures). It looks at the video information on a frame-by-frame basis for changes between frames. For example, if you're working with video of a talking head (a clip of a person sitting or standing with little motion), there's a lot of redundant information in the recording. The background rarely changes, and most of the motion involved is simple head movements and the movement of the area around the mouth. The compression algorithm compares the first frame (known as a key frame) with the next (called a delta frame) to find anything that changes. After the key frame, it only keeps the information that does change, thus deleting a large portion

of your file. It does this for each frame until it reaches the end of the file. If there is a scene change, it tags the first frame of the new scene as the next key frame and continues comparing the following frames with this new key frame. As the number of key frames increases, so does the file size.

Spatial compression

Spatial compression uses a different method to delete information that is common to the entire file or an entire sequence within the file. It also looks for redundant information, but instead of specifying each pixel in an area, it defines that area using coordinates.

Both of these compression methods reduce the overall file size. If this is not sufficient, one can make a larger reduction in file size by reducing colors, frame rate (that is, how many frames of video go by in a given second), and audio quality. Depending on the degree of changes one makes in each of these areas, the final output can vary greatly in quality.

Hardware vs. Software Codecs

Hardware codecs are the most efficient way to compress and decompress video files. They are faster and require fewer CPU resources than their software counterparts. In order to capture clean raw video, most machines require a hardware codec that allows the video file to be fragmented and distributed rapidly on your hard drive. These hardware codecs are expensive, but deliver high-quality results. Using a hardware-compression device will deliver high-quality source video footage, but requires viewers to have the same decompression device in order to watch it. Hardware codecs are used often in video conferencing, where the equipment of the audience and the broadcaster are configured in the same way.

Software codecs are less expensive, and freeware versions are readily available. Most commercially available digital-video packages have built-in codecs that output files of decent quality. The drawbacks to software codecs are that they are CPU-intensive and take a long time to analyze and compress files. There are freeware versions of commercial compression-software packages available online that will allow you to use basic compression features (but won't allow you to have advanced control over your clips).

Free Software Codecs

The following codecs are available for free:

Cinepak has been around for several years, and has been the default choice for many digital-video editors. It is widely used on CD-ROMs, it works on many machines and has middle-range output quality. It is included free with the QuickTime software and is widely supported in various architectures. Its drawbacks are that it has a very large compression time and that it doesn't work well with slow data rates.

The video codec supports a high data rate, and because of this it's not really suited to Web use. It works well for video intended for CD-ROMs. The quality-to-data rate ratio is low, and therefore it requires a higher file size for good quality. It uses a temporal compression scheme and is also available free with QuickTime.

The animation codec is suited best for work with computer-generated images. High-quality compressed files are quite large. Because this codec uses a pixel generalization technique, it does not like to compress material captured from videotape. Files compressed from videotape source material will be large due to video noise.

Motion JPEG was introduced in QuickTime 2.5 and is a lossy, spatial compression codec. It produces large file sizes at 100 percent quality, but will create smaller file sizes than Cinepak at lower data rates, making it a good choice for Web use.

Architectures

Now that we've discussed codecs as the actual compression/decompression device, we need to talk about how the file is actually viewed on a client machine. There needs to be a standard (or at least semi-standard) format for the client machine to interpret the compressed file.

An **architecture** is a software package that allows information to be traded in a standard format. Architectures in digital video support file, storage, and playback formats, and allow you to specify which codecs are used.

There are several architectures for Web clients currently available. A few of the more popular architectures are QuickTime, VDOLive, and RealVideo.

QuickTime is a simple cross-platform architecture (for Macintosh, Unix, and PC). Depending on the version of QuickTime players and plug-ins the client has installed, the user may or may not have to wait for an entire video file to download before beginning to view it. QuickTime codecs and the player are available free. Because QuickTime has been around for several years, it is widely used.

VDOLive allows the user to see video files while downloading, making adjustments for connection speeds and dynamically providing a (seemingly) constant delivery of the video files. VDOLive uses streaming technology and requires a special server to deliver the files, although versions of the player are available free.

RealVideo also uses streaming technology, but usually there are several files for each clip that are targeted to a specific data-transfer rate. The user selects a file based on her connection speed, and the file that's served is optimized for that speed. RealVideo requires special server software, although some versions of the player are free.

Arm yourself with an understanding of architectures and codecs. You'll then be able to make decisions that will allow you to create good media for the Web. Delivery will not be instantaneous (or even within my deluded professor's half-hour), but by using the appropriate codecs and architectures, you'll deliver the high-quality content that your audience won't mind waiting for.