Compression Primer

TABLE OF CONTENTS

Getting started with compression

This primer is all about helping you prepare high-quality compressed videos. The best starting point for achieving great results is to understand why and how compression is done. While the mathematical basis of compression is very complex, the basic principles are easy to understand. Uncompressed video files are too large and the data rates are too high to transport effectively over a local area network or the Internet, so you need to compress them.

BITSTREAM TYPE	DATA RATE (KILOBITS PER SECOND)	TIME ON ONE 700MB CD-ROM
Uncompressed high definition (1920 x 1080 29.97 fps)	745750	7.5 seconds
Uncompressed standard definition (720 x 486 29.97 fps)	167794	33 seconds
DV25 (miniDV, DVCAM, DVCPRO)	25000	3 min, 44 seconds
Typical DVD	5000	18 min, 40 seconds
VideoCD	1167	80 minutes
Broadband web video	100-2000	3 hours, 8 minutes at 500 Kbps
Modem web video	18-48	48 hours, 37 minutes at 32 Kbps

Data rates and storage requirements for different DV media types

The ratios between data rates used for authoring video and for distributing compressed video can be huge. For example, converting from digital video (DV) to DVD typically entails a 5:1 compression, and converting an uncompressed source to a modem-compatible video requires about a 9000:1 compression. That leaves only about 0.1% of the original data left. You may find it surprising that video on the web looks and sounds as good as it does!

In addition, different types of compressed video vary in bit rate. For example, in web video, there is about a 100:1 ratio between the low end of dial-up modem rates and the high end of broadband rates.

Type of video (format) and compression type are two different things: one video format may support multiple compression types and ratios, and a type of compression may be used in multiple formats. However, most standard video formats include requirements that specify compression types and data rates.

DVD formats

Americans watch compressed video on DVD more than on any other media. (Digital broadcasts over cable and satellite are close runners-up.)

Modern DVD-R burners can put 4.7GB of data on a disc (called a DVD-5). Manufactured DVDs can use a dual-layer format that can hold 8.54GB (called DVD-9) of data. There are many different types of DVD, with higher-capacity formats being continually introduced. A DVD-5 disc can hold nearly 5GB of data, which is enough for a high-quality, two-hour movie. Consumer DVD burners available now can write to even higher-capacity discs.

The type of DVD that plays in a DVD player is most accurately known as DVD-Video. There is also DVD-ROM, which is a computer-formatted disc that requires a computer to play it—in effect, it's a large capacity CD-ROM. There are also hybrid DVD-Video/DVD-ROM discs, which play in regular DVD players, but also include information available for computer playback.

DVD-Video discs are mastered in either National Television System Committee format (known as NTSC, the format used in the U.S. and Japan), or Phase Alternating Line format (known as PAL, the format used in Europe and most of Asia). A set-top player (such as a home DVD player) designed to play one format typically won't play discs in the other format; however, computers can typically play both types. DVDs can also be region-coded, so they only play in DVD players or software intended for a certain region. Region Coding is an optional feature.

VideoCD formats

VideoCD is an older disc-based format. It has much lower resolution and higher compression ratio than DVD, and so it provides lower quality. However, it uses standard CD-ROM discs, so VideoCD discs can play in a variety of devices, including most personal computers. A manufactured VideoCD can play in any DVD player.

Like DVD, VideoCD discs can be either NTSC or PAL. VideoCD is very popular in PAL countries, especially in Asia. As with DVD, software players support both formats, and many set-top players support only one.

CD-ROM formats

Starting in the early 1990s, CD-ROM video was a popular medium. While the web has largely replaced CD-ROM as a medium for interactive content, people still use CD-ROM video for many applications, such as computer games and kiosks. Compared to web video, CD-ROM offers much higher bandwidth than broadband connections, and is capable of DVD quality on fast computers.

Internet formats

The web is the medium providing the most encouragement for development of video compression technology. The bandwidth available on the web is much less than on discs, so it is critical to make every bit of data count. Also, unlike a consumer electronic device, the web enables developers to update player software at any time. So, the web has become a technology battleground for producing the best technology, as quickly as possible.

Web video can use either real-time streaming mode or progressive-download mode. Real-time streaming files require specialized streaming server software and are limited in quality by the bandwidth available between the server and the end user. Progressive-download files load from standard web servers, and may take a long time to download, but may also provide higher and more reliable video and audio quality. Understanding the differences between these modes, and the strengths and weaknesses of each, is a major focus of this document.

Compression and mobile devices

Mobile devices, such as personal digital assistants (PDAs) and modern cellular phones, are a relatively new medium for compressed video. These devices have small screens and slow processors, even compared to a laptop computer, but the ability to play video back on extremely small and lightweight devices can have great value.

Most compressed video formats can be played on some mobile devices, including Adobe® Flash® Video, MPEG-4, Microsoft® Windows Media, and RealMedia. Dedicated mobile formats, like Kinoma, are also available.

In many ways, video on mobile devices, especially cellular phones, is reminiscent of the early days of video on desktop computers, in that technology is still evolving. Today, there are no universal formats or settings for mobile devices. Instead, content has to be optimized for a small class of devices or even a particular model. When the optimization is successful, the results can be great.

Understanding the fundamentals of compression

Compression tools and formats hide the mathematical complexity of compression from the user, but to achieve the best results, you need to understand the fundamentals of why and how compression tools and formats operate.

How people see and hear

Our senses of sight and hearing are powerful, but have limitations that make compression possible. The human eye and brain are skilled at perceiving high levels of motion and detail at the same time, or high levels of motion and color at the same time. In the same way, hearing is powerful, but has its limitations. We may easily hear a sound in a quiet environment, but not hear it in a loud one. Human ears are best attuned to the frequencies of human speech and are less perceptive of higher or lower tones. Compression technologies take advantage of the strengths and weaknesses of human senses by reducing data that isn't likely to be perceived.

How compression works

Fundamentally, compression works by summarizing a sequence of images and sounds as efficiently as possible. Video that has been compressed doesn't provide an identical representation of the source file back to the user—the data rates would be far too high. Instead, the encoded files describe the important details of the content as tersely as possible while still providing a reproduction that, to the human sensory system, provides the experience of the original as accurately as possible.

A matched pair of a compressor and a decompressor, better known as a codec, performs the compression. A compressor is part of the encoding process, reducing the amount of data required to store the video. A decompressor works on the receiving end, decoding the compressed data so that it can be presented to the viewer. It's important that the pair be matched because the decoder (decompressor) needs to understand the encoder's (compressor's) summary of the data.

Types of compression

Video codecs can use spatial compression, temporal compression, or a combination of both.

Spatial compression

Spatial compression, also called intraframe compression, affects only a single frame at a time. Most compressors only use spatial compression, so there aren't any interdependencies among frames. This type of compression makes random access to any point in the video and editing easy.

Spatial compression finds redundancy within a frame—either in areas of flat or similar colors, or areas of a frame that look like other areas of the frame—and saves space by encoding these similar areas by reference, rather than by a literal description of color attributes for each and every pixel.

Many codecs are based on a technique called discrete cosine transformation (DCT). These codecs deal very well with smooth gradations in an image, but they may not be as successful at encoding random details or sharp edges at lower bit rates. Any compression technique that encodes images at too low a data rate, can cause visible degradation of the image. This degradation results in visible flaws called artifacts. Two of the most common kinds of artifacts are ringing (in which a sharply defined edge may have a halo or smudge) and blocking (in which a smooth edge can become blocky in appearance). One common area of trouble is small text with sharp edges. Some codecs use very different base technologies, like wavelet, and therefore deal better with sharp edges.

Temporal compression

Temporal compression, also called interframe compression, adds the ability to use other frames as a reference for the current frame. In video, each frame tends to be similar to the frame preceding it. Temporal compression encodes only the changes from one frame to another (with the exception of key frames, as described below, which contain a complete representation of the data in a frame, without reference to previous frames).

Temporal compression gets a lot of research attention and has the most room for future improvements.

Frame types and compression

You can reveal many fundamental concepts of compression by looking at the kinds of frames in a file. Depending on the codec and options used, one, two, or three kinds of frames may be in a file.

A keyframe, or I-frame, is a complete image that is compressed using intraframe compression. An I-frame isn't based on any other frames in the file. The first frame of a movie is always an I-frame.

A delta frame is compressed using interframe compression and contains only those portions of the adjacent frames that are different. If two frames are mathematically identical, and the second one is a delta frame, it will require only a few bits to encode. But if more than half of the image is different, most codecs automatically generate a new I-frame. For typical content, a delta frame might be only 10-20% the size of the same frame at the same quality as an I-frame.

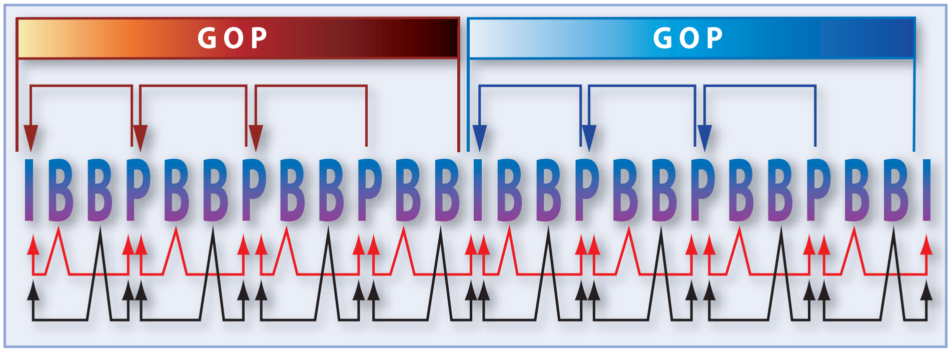
There are two main types of delta frames: P-frames and B-frames. The content of a P-frame (predictive frame) is based on the previous frame; the content of a B-frame (bi-directional frame) is based on the previous and subsequent frames.

I-frames are important for accessing random points in the video because they're the only frames that can be decoded all by themselves. To access a delta frame, a player must not only decode that frame, but the I-frames to which it refers. As the frequency of the I-frames increases, the number of frames that needs to be decoded for a random access operation decreases.

B-frames are mainly used in MPEG formats. A B-frame can use image elements from the frames on either side of it, which improves compression efficiency. However, B-frames cannot use image elements from other B-frames. This restriction prevents frames from being based on frames that are based on the first frame—an impossible recursive loop.

Most authoring codecs, like Motion JPEG and DV, only use I-frames. Editing is easy with I-frames because you can reassemble frames without regard for the other frames they might be based on.

Codecs vary in how well they encode complex frame transitions. For example, fades up and down are very difficult for some codecs, but some newer codecs have code built in to handle fades.



A typical 15-frame Open GOP as used in a NTSC DVD. Note that in an Open GOP, the trailing B-frames can refer to the first I-frame of the next GOP.

Color modes

While it might seem esoteric, the different ways that codecs describe color matter a great deal in practice. They can be a source of confusing terminology in the video world.

There are three basic types of color in today's codecs:

- **Monochrome** In a monochrome mode, only luminance (brightness) matters. This color type works great for black-and-white movies. Most video codecs don't have an explicit grayscale mode. Notable exceptions are Cinepak and JPEG (although, not all tools expose the color space control).
- **Red, Green, Blue (RGB)** Computer screens and televisions are natively RGB—the screen has red, green, and blue phosphors for CRTs and planes for LCDs. RGB isn't very efficient for video content, so RGB codecs aren't used for broadcast video delivery anymore. However, RGB is widely used for screen recordings and motion graphics.

- **Y'CbCr** (also called YUV, although YUV properly only refers to how color is used in NTSC video) These codecs encode luminance and two color-difference channels (the same technique used in the Adobe Photoshop® LAB mode). Because there are three colors, you can convert Y'CbCr to RGB for display. By separating out the chroma and luma channels, you can keep the critical luma information at full resolution and use less resolution for the chroma information. This process greatly reduces data rate and processing requirements. You can convert RGB to and from Y'CbCr adequately, but there are always some colors that are available in one mode but not the other, and there can be rounding errors as part of the conversion. By working in tools that support 16-bit-per-channel RGB, like Adobe After Effects® software, you significantly improve the quality of this conversion.

Color depths

Color depths define the precision of the colors and determine how many gradations of intensity there are. Color depth is measured in the number of bits per channel (bpc) and fall into the following categories:

- **8-bit (indexed) color** (often called 8-bit or 256 color) Early color computers used indexed color. Instead of each pixel having a measurement for red, green, and blue, each pixel was assigned a code, and a look-up table translated that code to actual RGB values, which the computer could display. Some codecs supported even lower modes, like 4 or 16 colors, providing a very simple compression scheme by reducing the number of bits required to store the image. The 8-bit color depth doesn't work well for video because even the simplest scenes use more than 256 colors. The only significant format that still requires 8-bit indexed color is GIF.

- **5 bpc** (often called Thousands of Colors or High Color) After the 8-bit era, the most common consumer bit depth was 5 bpc. On Macintosh® computers, this was exactly 5 bits per red, green, and blue, with one bit reserved for an alpha channel. Microsoft Windows® gave an extra bit to green, which improved luma quality (brightness information), but left no room for an alpha channel. The 5-bpc color mode was used for some very early video-compression codecs. Today, it is mainly used for screen recording applications where it may provide better compression and performance than 8-bpc color without any obvious degradation.

- **8 bpc** (also called Millions or True Color) Sometimes called 24-bit or 32-bit mode. The 32-bit mode still uses 8 bits per channel but with an added alpha channel.

- **10 bpc** This mode is used in high-end Y'CbCr authoring systems (such as Digital Betacam and D1) and some analog capture cards. It provides 1024 gradations instead of 256, which reduces banding (uneven transitions in gradated colors), especially with multiple generations of effects. It isn't used in any delivery formats, although some authoring codecs support it. The 10-bit Y'CbCr mode provides four times more accuracy per channel than 8-bpc color when converting to or from 16-bit-per-channel RGB in After Effects.

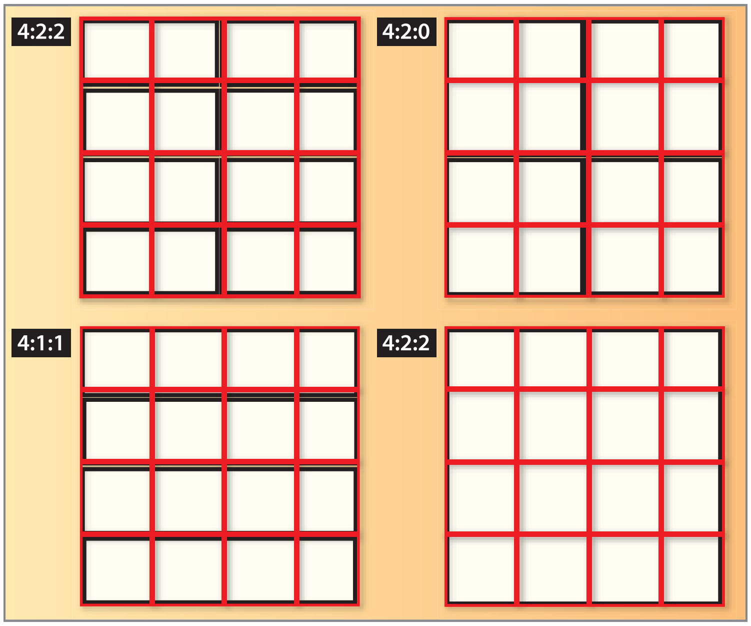
- **16 and 32 bpc** Some high-end applications use 16-bpc or 32-bpc color to preserve more color detail. These modes are especially useful when you have a multi-step authoring process, need to apply visual effects to your images, or when you convert from and then back to Y'CbCr. After Effects 6 and later can process 16-bpc color. After Effects 7.0 introduces 32-bpc color capabilities. Video that is saved at 16 or 32 bpc is not currently delivered to end users.

Color sampling

Y'CbCr formats support color subsampling. Color sampling records fewer pixels of chroma (color) information than luma (intensity) information. The terminology for color sampling is confusing because it uses an X:Y:Z format. The first value is the number of luma samples that are described—this is almost always four. The second value is the number of chroma samples per luma sample on the first, third, fifth, and succeeding odd lines. So if this value is two, there are two chroma samples per four luma samples horizontally. The last value is the number of chroma samples per luma sample on the second, fourth, sixth, and succeeding even lines. If this number is zero, the chroma sample from the first line is used.

- **4:4:4** Chroma is sampled every pixel with no subsampling. Y'CbCr delivery codecs don't use this format because it uses a lot of bits and doesn't provide any additional apparent quality. In authoring applications, it is sometimes internal. RGB is always 4:4:4.

- **4:2:2** This format is used for high-end video authoring, in formats like DV50, Digital Betacam, D1, and Motion JPEG. A few delivery codecs like H.263 use it as well.
- **4:2:0** The vast majority of delivery codecs use this format. There is one chroma sample per 2 x 2 block of pixels. This chroma sample looks nearly as good as 4:4:4, but at half the bits per frame. It is also used for PAL DV (but not PAL DVCPRO).
- **4:1:1** NTSC DV and PAL DVCPRO use this format. It contains the same number of bits per frame as 4:2:0 but with one sample per 4 x 1 block of pixels. This format is better for interlaced content because each field gets its own sample; however, it can result in visible chroma artifacts in areas of high saturation, which can be a problem with bluescreen or motion graphics source.
- **YUV-9** This format has one chroma sample per 4 x 4 block of pixels. This format results in noticeably degraded quality, so it isn't used in modern codecs. Legacy codecs that use it include the Indeo family and the original version of Sorenson Video. (Sorenson Video 3 uses 4:2:0.)



The most common types of color subsampling

Audio compression

While audio may use only a small fraction of the bits in a compressed file, it is responsible for half of the experience.

Sampling rate

The most fundamental factor in audio is sampling rate, which is measured in Hertz (Hz), or cycles per second. Audio CDs use 44100 Hz, or 44.1 kilohertz (KHz), which provides excellent reproduction.

SAMPLE	USED	SUFFICIENT FOR
8 KHz	Phone system	Speech
11 KHz	Old multimedia standard	Better speech
22.050 KHz	Also old multimedia standard	Recognizable music
32 KHz	Common in broadband video	Minimum for decent music
44.1 KHz	Audio CD	Music
48 KHz	DVD, DAT	Music
96 KHz	High-end audio recording	Music mastering

Typical sample rates for common use.

Bit depth

Bit depth is the number of bits used for each sample in each channel—the higher the bit depth, the better the audio quality.

- **8-bit sampling** Originally, multimedia audio used 8-bit sampling, which means there was a measurement of volume from 0-255 for each sample. This bit depth, which was responsible for that harsh sound in the early CD-ROM titles, produces fairly poor quality audio. There is no reason to distribute 8-bit audio anymore. Modern 16-bit codecs provide better quality at a smaller file size than 8-bit codecs were ever able to.
- **16-bit sampling** This bit depth is the current standard for audio distribution. Most modern codecs and all audio CDs use 16-bit sampling as well.
- **20-bit sampling** This bit depth is used for professional audio recording to provide a better dynamic range than 16 bits. Codecs that support 24-bit audio (described later) can effectively take advantage of 20-bit sources.
- **24-bit sampling** This bit depth is rapidly becoming the standard for professional audio recording because it provides more sonic detail than 16-bit sampling, meaning you can use many more effects and layers without introducing errors. A few delivery codecs now support 24-bit sampling, most prominently Microsoft Windows Media Audio (WMA) Professional. The difference between 16-bit and 24-bit samples is not easy to hear on a typical home computer speaker system, but it could begin to matter as people integrate personal computers into home theaters.

Channels

Most audio from computers is either mono with a single channel or stereo with two channels. Low bit rates generally use mono because it requires fewer bits. The industry is evolving to use multichannel (5.1 and 7.1) audio support as well. Multichannel support is included today with Microsoft WMA9 Pro and RealNetworks RealAudio 10. And of course, DVD has always supported multichannel through Dolby Digital.

Perceptual audio compression

Reducing the sample rate and number of channels can reduce the data rate; however, good compression must go well beyond that. Most modern codecs use perceptual encoding techniques: that is, an internal model of the human auditory system determines the important parts of an audio source and spends the available bits on those parts. At low data rates, the audio can still sound quite altered from the source. The number of bits required for “good enough” quality has been dropping precipitously in recent years.

Compression efficiency

So, how does someone compare codecs? The fundamental measurement is compression efficiency, which is defined by how many bits are required to reach a particular quality level. When improving compression efficiency, you can choose to lower the data rate while keeping the quality the same, increase the quality while keeping the data rate the same, or pick a point in between.

You can’t measure compression efficiency with a single number. Relative performance varies greatly and depends on the source and the data rate. Typically, the higher the data rate, the smaller the quality differences. It’s with aggressive data rates and difficult content that compression efficiency makes a difference.

Authoring video for future compression

One key to high-quality compression is creating content that compresses well. Sometimes it isn’t possible to create this type of content—legacy content is legacy content. But when you create content for compression, there are a variety of techniques that you can apply to produce better results. You can apply these techniques during preproduction, production, or post-production.

Preproduction

In preproduction, you plan the various elements of a project. It’s best to introduce the idea of compression early in a project, so all members of the team can consider the implications.

Production

During production, there are techniques you can use to produce video that compresses well:

- **Interlaced versus progressive scan** For content shot with a video camera, one fundamental decision is whether to use interlaced or progressive scan video. Historically, video was always interlaced. In an interlaced frame, the odd and even lines, called fields, contain images that are captured half the frame duration apart. Each video frame is made up of two fields.

  Why do fields exist in video? The first television sets used a single electron gun to shoot a steady beam of electrons at the inside surface of the picture tube. These electrons would excite the phosphorous material that coated the inside of the picture tube and make it glow, creating an image. The gun would move from one side of the screen to another, illuminating a single line, known as a scan line. It would then drop down a line and move across once more. It would do this for the 486 lines that made up the image. The problem was that by the time the gun would get about half way down the screen, the lines at the top were already beginning to fade out.

  The solution was to split the 486 lines into two fields called odd and even (consisting of odd numbered lines and even numbered lines). Now the electron gun could skip every other line and draw a complete field from top to bottom. It would then start from the top and fill in the missing lines. The result was that the eye would see a complete image because only half would be fading away at any one time. This "interlacing" of the odd and even lines became known as interlace scan or field scan mode.

  NTSC video has a frame rate of 29.97 frames per second (fps), or roughly 30 fps, with two fields per frame (59.94 fields per second). PAL video is 25 fps (50 fields per second). As a result, the second field shows an image that is about one-sixtieth of a second later in time in NTSC and one-fiftieth of a second later in PAL. If motion occurs in the video, then the image in one field will be slightly off from the image in the other. You can see this effect if you pause your VCR. Many VCRs will pause on a frame and continue to alternate between the two fields that compose the frame. This causes an effect called field jitter.

  This difference between fields adds complexity to the compression process. A good video compression tool must read and understand fields, compress them independently, and allow for them to be played back as fields in the compressed format. Compression for the web generally does not show fields because computer monitors are not field-based. Advanced technology for computers allows the entire screen to be drawn in a single pass, usually with more than one electron gun for tube-based displays. Flat panel monitors do not use electron guns, and they generate the entire screen instantly. This process is known as progressive scan.

  Interlaced video does not look good on progressive scan monitors. Compression techniques for the web usually change the video from interlaced scan to progressive scan. The most basic method for this change is to discard one field and duplicate the other to create the full frame. This process reduces the amount of data you started with by 50%, which can adversely affect the final quality. The more complex method is for the encoder to read both fields and interpolate (blend) them into a single frame by changing the pixels to smooth out the timing difference between the fields. This more complex method allows you to still use all the original data from the video and produce higher-quality output for progressive scan monitors.

- **Camera motion** The amount and type of camera motion has a huge effect on the compressibility of content. Camera motion changes every pixel of every frame. Lack of camera motion is the simplest, of course. But modern codecs can encode camera motion, such as that created using dollies, quite well. Tilts and pans are more difficult. Depending on the codec, rotating the camera (a Dutch) can be trivial or extremely costly in compression efficiency. The most difficult motion for any codec is hand-held camera motion with random jostling.

  For compression, it's much easier to handle changes of view by editing rather than camera motion, especially at lower data rates. In post-production, the motion stabilization tools in After Effects can remove unwanted camera motion before compression to reduce the size of the output video file.

- **Shutter speed** This speed controls how long the aperture is open in the camera, and hence, how long light hits the film or the CCD. Slower shutter speeds let through more light, and hence less grain, which helps compression. Slower shutter speeds also cause moving parts of the image to become blurry, which makes them easier to compress.

  The default shutter speed for most cameras is half that of the frame rate: 1/60 of a second for NTSC, 1/50 for PAL, and 1/48 for film. The defaults provide good results for most video meant for compression. The exception to those defaults is action and sports videography, where videographers avoid motion blur to make slow-motion images crisp and clear. Be aware of these exceptions in compression tests.

- **Backgrounds** A difficult motion for codecs is foliage rustling in the breeze. Foliage is green, which makes up the majority of luma and therefore gets allocated many bits in the compression. Leaves have hard, irregularly shaped edges, and details that require many bits to encode well. Additionally, the motion of foliage in the breeze is largely random, with leaves constantly overlapping and revealing, all motions that are very difficult for motion estimation to handle.

- **Depth of field** A useful and subtle technique for controlling the complexity of images is depth of field. By managing the depth of field, you can keep unimportant objects in the background or close foreground out of focus while clearly showing the actual subject of the shot. Depth of field is very common in cinematography and photography, so it looks natural to the audience.

  If there is not enough spatial separation among objects, backing up the camera and using a telephoto lens narrows the region in focus.

Post-production

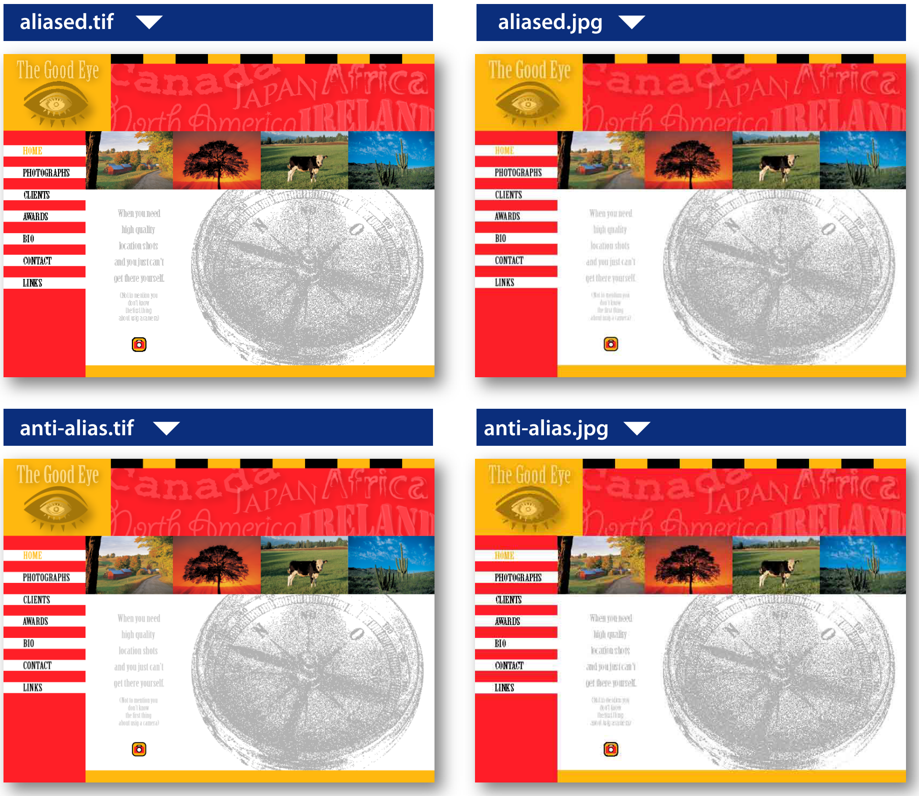
Post-production techniques can have as much effect on compressibility as production techniques. Because post-production techniques occur later in the process, it is often possible to adjust them based on the results of test compressions, making fine-tuning much easier.

- **Static versus moving images** As in production, superfluous moving images aren't appropriate for content that you plan to aggressively compress. They are not that important for DVD, but they are for web and CD-ROM.

- **Motion blur** In the same way that a real-world shutter speed introduces motion blur that helps compression, you can use the virtual shutter in rendering applications. For effects that support motion blur, render them with a shutter speed that matches that of the source. This speed might be expressed as a duration, or, as in After Effects, as the angle of a virtual shutter. The default is 180°, which corresponds to a shutter speed half that of the frame rate.

- **Anti-aliasing** This rendering technique softens the edges of rendered elements, letting a pixel at the edge of an area be an intermediate value. Smoothing out transitions among parts of the image improves compressibility and appearance.

  Anti-aliasing may be its own control, or it may be part of an overall group of rendering settings. For example, in After Effects, anti-aliasing is automatically turned on when encoding in Best mode.



These samples show an Adobe Illustrator® file (top left), the file exported to a Photoshop file (PSD) aliased (top right) and anti-aliased (bottom left), and then converted to JPEG in Adobe ImageReady® (bottom right). The files were converted to the same size, and, the anti-aliased image has significantly fewer artifacts.

- **Progressive rendering** When working with interlaced source video, it may be appropriate to render motion graphics as progressive scan when making the compression master. By rendering the project in progressive scan mode, you can deinterlace the interlaced source, but the graphical elements remain a perfect progressive.

Capturing video: choosing formats and connections

Capturing video is how content gets from an external source like a VTR or camcorder into the computer. The type of video, or format, and to a lesser extent the connection type you select to capture content, can dramatically affect the quality of the video on the computer. It's always desirable to start with the highest possible quality video and audio signal. In almost every case, compression doesn't make video look better than the original source. The higher the quality of the source material used, the higher the quality of the final compression results. The inverse is also true, so it is helpful to build a basic understanding of the video and audio formats available and how they compare in quality.

Today's editing software, such as Adobe Premiere® Pro, and many compression software tools, can capture content in a variety of ways. Video can be captured from analog or digital sources. Both have a variety of different video and audio formats available and various physical connection types that can be used. In most cases, you'll need to install a video capture card with the proper physical connectors in your computer to perform the captures.

To understand the video formats, you must understand the video signal components. In the beginning of video, we dealt only with brightness, or luminance (luma) signals. The variations from very low brightness all the way to complete brightness gave us black-and-white television. Color was added by generating two channels of color, or chrominance (chroma) values for the signal. To accommodate the millions of black-and-white television sets still in use, a method for adding color was invented that allowed a black-and-white set to ignore the colors and still show the luminance, or black-and-white, picture. Since broadcast television is transmitted over antennas, the luminance and two color channels are sent together as a single signal. The various video formats used today were created using this technology.

A list of the common video signal formats follows, along with the popular types of cables and connectors for transmitting them.

- **Analog formats** Analog video uses fluctuating voltage and frequency signals to transmit image data. It ranges greatly in quality but is inherently susceptible to visual interference from electronic noise, like power lines, and suffers attenuation, or the loss of signal power and quality over long cable runs (usually 300 meters or more). Equipment that supports analog signals is generally more affordable, and analog video formats are still the most commonly used because of the long-term use of legacy equipment that is still active today.

  Because computers only use digital data, the analog source must be converted to a digital copy to be captured. When it is sent back to an analog medium like videotape, it is converted back to an analog signal. This conversion can lead to minor errors (quantization errors) being introduced because of the imprecise relationship between analog and digital signals. You can minimize these quantization errors by using the right format, good equipment, cables, and connections.

- **Composite** A composite video signal carries the luminance and the chrominance channels in a single signal. The biggest problem with composite video is that luma information can leak into chroma, and vice versa. This leakage can lead to noise, known as chroma crawl, which can be amplified by the video capture process and reduce the overall quality. Composite video is generally understood to be the lowest quality video signal that you can use and is more appropriate for video delivery (broadcast) than as a source for content creation (video editing, DVD authoring, or encoding for the web). Common sources of composite video are VHS (1/2 inch tape) VCRs and camcorders, broadcast television signals, and older U-Matic 3/4-inch professional VTRs.

  The connectors common to composite video are RCA or Cinch for consumer equipment and BNC (Bayonet Neill-Concelman), a twist-locking connector common to professional and broadcast video equipment. As the quality of the RCA or BNC cable that is used increases, the noise and attenuation decreases.

- **S-Video** The introduction of the S-Video format was a major advance in quality for consumer and professional video editors. S-Video splits the luma and chroma channels into two separate wire pairs, which eliminates generally half of the composite noise artifacts. The two chroma channels are still combined and still interfere with each other, leaving some noise in the signal. Many professional and consumer-grade video equipment manufacturers, including Hi8 camcorders and S-VHS VCRs, provide an S-Video connector. While no longer considered true professional grade, it is a big improvement from composite and can be considered a good video capture minimum standard for for editing and compression.

  S-Video, or Y/C (Y for luma and C for chroma) video signals mostly use a specific four-pin connector on S-Video cables. These cables are intended for short distances (usually 100 feet or less). Some types of professional and broadcast equipment use two BNC cables, one for luma and one for chroma, to carry S-Video signals. This setup provides a connection that can be run farther and that can be locked into place to prevent the cable from falling out and losing the signal.

- **Component analog** Component analog, also known as YUV (Y for luminance, U for one chroma channel, and V for the other chroma channel) was the professional and broadcast standard for many years and it is still widely used today. With component analog, the luminance signal and the two color signals are all transmitted over their own dedicated cables. Because all three components or channels of the video signal are transmitted independently, the quality of the signal is quite high. Noise is very low and the colors in the video are richer and more precise. In broadcast and professional environments, component analog has mostly yielded to digital video formats. Component analog has become very popular in the consumer market, though, as the preferred format for connecting home DVD players to new television sets, owing to the pristine nature of the signal compared to S-Video and Composite formats.

  In professional environments, the component analog signals are carried by three individual BNC cables because of their ability to accommodate long runs and locking connections. In the new home DVD players, it is common to see RCA or Cinch connectors used that are color coded red, green, and blue to make it easy to connect the DVD player correctly.

ANALOG-TO-DV CONVERTERS

Certain devices can perform a live conversion from analog video formats to DV25 for capture. They usually offer composite, S-Video, and unbalanced analog audio conversion to the DV format for storage on the computer. The 25-Mbps data rate of DV is sufficient for content that camcorders produce, but it can be troublesome for converting sources with significant analog noise on lower-end versions of the converters.

- **Unbalanced audio** This type of audio connection consists of a single wire that carries the signal, surrounded by a grounded shield. It is used commonly in consumer audio products because the connection and circuitry are less complex. The downside is that unbalanced connections are more susceptible to interference, so they are not often used in professional applications.

  There are three basic kinds of unbalanced audio connectors. The most familiar to consumers is traditional unbalanced audio that uses RCA or Cinch jacks (normally red for right and white or black for left). To connect a stereo signal, you can use a single RCA cable for each channel, or a stereo cable that has two single cables molded into one.

  There are other single-cable unbalanced connectors as well. The quarter-inch connector is common to high-end headphones and is standard on most home stereo equipment. Similar but smaller is the eighth-inch connector that is standard for portable audio devices and computer sound cards.

- **Balanced audio** This type of connection consists of two wires, which serve to balance the signal, and a shield. A balanced connection is far less susceptible to interference, so it is possible to maintain high quality with long cable runs. Balanced audio normally uses the XLR, or three-pin, locking connector, which provides a much tighter and robust connection than the connectors used for unbalanced audio. Once again, longer cable runs and a locking connector form the basic standard for a professional connection.

Digital formats

DV formats the original video signal on tape using digital data similar to that used by computers. All brightness and color data is stored in this very accurate computer-style method and as such suffers no problem of interference or noise between the channels of the video signal. Digital signals are therefore cleaner and crisper than analog sources. The digital data format allows digital video to be copied from tape to a computer hard disk without the conversion process that analog video requires.

Multiple digital formats are available in the video industry. Many of these formats are related to compression and not acquisition or filming with a digital camera. This section covers only the most popular acquisition formats of digital video.

- **DV25 (DV)** This format is most commonly known by the physical connection standard it uses, IEEE-1394 or FireWire, however it uses only a fraction of the data rate and speed available in the IEEE-1394 connection standard. It revolutionized the video editing industry, becoming the fastest growing format for new editors and the de facto standard for news, owing to the small size and relatively low cost of DV camcorders. Video is compressed 5:1 during acquisition or recording in order to reduce the data rate of the signal to 25 Mbps, and is only slightly higher quality than the S-Video analog format. Higher-end DV camcorders can produce a DV signal that is almost as good as component analog video at the same DV data rate because of better recording technology.

  The DV format includes a series of computer-style 1s and 0s and no analog, which enables the video data to be copied from the DV tape directly to a computer. The data on the computer is exactly the same data as on the tape, so the result is a perfect copy. Many nonlinear video editing applications, like Adobe Premiere Pro, can read and edit this format natively, and finished video is virtually the same as the original source footage. Few to no processing errors are introduced.

  Both the audio and video signals are carried on a single FireWire or IEEE 1394 cable (the Sony variation of which is known as i-Link). The cable also carries device control signals that allow the computer to control the DV camcorder.

  In addition to DV25, there are a number of other popular DV formats, such as DV50 and DV100, which have data rates of 50 Mbps and 100 Mbps, respectively. DV50 is used in Panasonic's DVCPro and Sony's DVCAM; DV100 is used in high-definition digital video formats like DVCProHD.

Adobe Premiere Pro supports DV25 and DV50 over standard FireWire ports. These ports are common to most new computers; inexpensive FireWire add-in cards can also be added to most computers. Four-pin and six-pin DV connectors are available, and they are compatible with each other. DV camcorders usually feature the four-pin connector since it is smaller and only four pins are needed for data transmission. The extra two pins on a six-pin connector are for power, and are commonly used, for example, by FireWire hard disks.

- **SDI (Serial Digital Interface)** SDI is a high-end, professional format for digital signals. The SDI format, technically known as CCIR-601 or 601, is superior in quality to component analog video and is the standard in professional and broadcast television. Because it is digital and uncompressed, it can be edited and copied many times with no detectable loss of quality. SDI was originally meant for standard definition, but a high-definition version, HD-SDI, is growing in popularity as HD becomes more common. (Please see the primer, Understanding and Using High-Definition Video, available on the Adobe web site.) SDI is typically only supported in high-end gear and can be expensive, especially for SDI video monitors.

  The video, and sometimes the audio, is carried in a single BNC cable.

- **Digital audio** There are many different digital audio formats. Consumer-grade digital audio is carried on electronic (such as S/PDIF) or optical (such as square TOSlink or round Miniplug) connection cables. Digital video carries an embedded digital audio signal with the digital video signal. The most common professional-grade digital audio format is Audio Engineering Society/European Broadcasting Union (AES/EBU). A stereo channel of AES/EBU can be transmitted over single BNC or single XLR connection cables. SDI video equipment can sometimes carry embedded AES/EBU digital audio, but it is more common for a separate AES/EBU or balanced audio stereo pair to be used with the SDI video signal.

Choosing the right capture codec

When you capture content, you must choose an appropriate codec. The choice depends on the capture hardware you're using, the format from which you're capturing, and your storage requirements. There are three basic types of codecs to choose from: DV, Motion JPEG, and uncompressed.

DV codecs

The DV25 format is ideal when capturing from a DV25 deck. Over a FireWire connector, you can use DV25 to copy the native bits from the tape onto a hard disk, making a perfect copy. If you use a different format at a higher data rate, the resulting video will be slightly lossy.

Using a DV bridge to convert analog source to DV25 may work, but the quality lessens. The data rate of DV25 isn't quite enough for optimal encoding from source with analog noise. Also, the 4:1:1 color space also entails a loss in quality.

Motion JPEG codecs

Motion JPEG has been the leading format for high-end video-editing systems for some time. The proprietary codecs supported by many vendors of capture cards, for example the Targa and Pinnacle codecs, are variants of Motion JPEG and share its properties.

Motion JPEG is 4:2:2, and most implementations allow you to specify a data rate or quality level. Some implementations can even create a mathematically lossless Motion JPEG. Typically, data rates of 50 Mbps or greater are considered broadcast quality.

Uncompressed codecs

More and more cards support true 4:2:2 uncompressed video. The data rates are large, and the quality is no better than a lossless compression. One significant advantage of some uncompressed codecs is support for 10-bit-per-channel captures, which can perform better for analog and especially SDI sources. High-end video editing applications, such as After Effects Professional, are able to process video in more than 8 bits per channel, improving the quality. Some hardware vendors also include support for 10-bpc rendering in Adobe Premiere Pro. Because uncompressed codecs have such high data rates, you should use a fast drive system, such as a RAID (redundant array of independent disks) system.

***Note:** Don't confuse uncompressed 4:2:2 with uncompressed RGB codecs, often called None.*

Exporting video

Capturing from tape is required for many projects, but for some projects, all content is created electronically. In other projects, a lot of creative work is done during editing and compositing on the computer. In those cases, going back out to tape is an unneeded distraction. Many tools, such as After Effects and Adobe Premiere Pro can export to a compressed file from the timeline, a process that works well for a single file. However, if features in other encoding tools are required, for example, volume encoding to multiple output formats, it is useful to export an intermediate file from the timeline and use the exported file for further processing.

Exporting to file

A file you export for later processing is called an intermediate file. The goal of creating an intermediate file is to provide high quality in the most useful format without introducing any compression artifacts.

Choosing whether to preprocess or to avoid rendering

The first question that you should ask is when to apply preprocessing (described in the next section): during the rendering phase or later. Preprocessing during rendering generally results in a smaller file, but adds more rendering time. It also means you can't use an encoding tool for special operations, like inverse telecine.

Basically, to avoid rendering, you use the same codec and file format as your source video. If you don't preprocess, you don't need to uncompress and then recompress unmodified frames—instead, you copy them from the source into the output. Different tools support this process in different ways—in Adobe Premiere Pro, you turn off the Recompress option in the Video pane of the Export Movie Settings dialog box. For this process to work, the video must truly be unchanged with the same frame rate, frame size, pixel aspect ratio, and so on. The simplest way to ensure that no changes were made to the video is to use the same project preset for export that was used for import, which is the default behavior in Adobe Premiere Pro.

There is one exception to exporting with the source codec: when the source is DV25 and contains a number of motion graphics or compositing. Because these synthetic elements have precise color, exporting to a 4:2:2 codec, like Motion JPEG, instead of NTSC DV25's 4:1:1 provides twice the color detail.

When you preprocess, choose the highest frame rate and largest frame size that you will use. If you are targeting web codecs, use square pixels (pixel aspect ratio of 1).

Choosing a file format for rendering

You can choose from a variety of file formats for rendering. The most common formats are QuickTime and AVI; both offer good-quality intermediate codecs. You can also use MPEG-2 at high data rates. Modern tools like After Effects and Adobe Premiere Pro can export to Windows Media 9 format. It is essential to make sure that the encoding tool supports the file format.

Choosing a codec for exporting

When you export an intermediate file, pick a codec and a data rate that doesn't introduce compression artifacts. When you are not preprocessing, it's typical to use the source codec or Motion JPEG. When you use Motion JPEG, use it in single field when you export as progressive scan, or interlaced when you export as interlaced.

For AVI files on Windows, the Huffyuv codec is quite popular.

Understanding preprocessing

Preprocessing is the middle step of video compression and involves transforming the decoded frame of the source video into the optimized frame given to the codec. The goal of preprocessing is to transform the video into the final format.

Why preprocessing matters

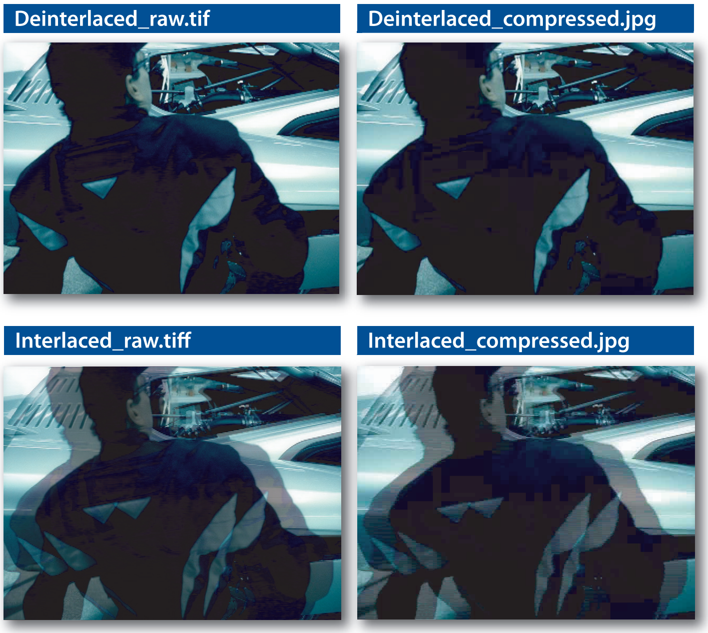
Preprocessing is one of the more subtle steps of compression, and it is often the most time consuming and difficult. It's also very important for quality, and getting it right can dramatically improve the quality of the final video. A well preprocessed video can look better than the same source file poorly preprocessed, but can be encoded at twice the data rate.

Preprocessing is critical for web video, but generally it is not needed when you convert to video formats like MPEG-2 for DVD or SVCD. Because those formats are played back as video, instead of on a computer screen, you can leave them as-is.

Also, you can create rendered graphics natively for computer playback without preprocessing.

Deinterlacing video

The most basic and critical form of preprocessing for the web is deinterlacing. Most video is produced and edited as interlaced (although, an increasing amount is done as progressive scan). But almost all web video is encoded as progressive scan. If the source frame of video is left with interlaced lines intact, the output shows the stacked lines. The lines look bad, but worse, those lines have sharp edges and are difficult to encode. So, at the same bit rate, deinterlaced video shows fewer artifacts than interlaced video.



The same video frame, shown prior to compression (left) and after compression (right), as both progressive (top) and interlaced (bottom). Note that there are more compression artifacts in the interlaced version, even though the file size is the same.

***Note:** MPEG-2 files encoded for DVD or SVCD don't need to be, and shouldn't be, deinterlaced. MPEG-2 can encode frames in a specific interlaced mode.*

Working in progressive mode when you author for progressive output can give you substantially better results when you export from a tool like After Effects and Adobe Premiere Pro, where you can add a fair amount of graphics or other synthetic effects on top of the video. The tools automatically deinterlace interlaced sources while they perform the additional processing in progressive mode, providing higher quality than if you deinterlace everything during export.

Understanding telecine and inverse telecine

A special case of progressive scan is film source and film-like 24P production. Traditional motion picture film runs at exactly 24 frames per second (fps) progressive scan, which doesn't convert to the 59.94 fps of NTSC or the 50 fps of PAL.

Converting from film to PAL is easy. The video is sped up 4% to 25 fps and converted into progressive PAL.

NTSC is a little more complex to perform. The film is first slowed down to 23.976 fps, and then it is converted to the 59.94 fps by using 3:2 pulldown. The first frame of film becomes three fields of video, and the next frame becomes two fields of video, and the next becomes three fields, and so on. This process results in two out of five frames having fields that come from different source frames.

Fortunately, 24P video is evolving into a production standard of its own, so editing tools can increasingly just author in 24P and then convert from that universal master to PAL, NTSC, film, or web formats as appropriate. When possible, it is best to convert to 24P on import and then do all the work on the project in 24P mode.

Cropping

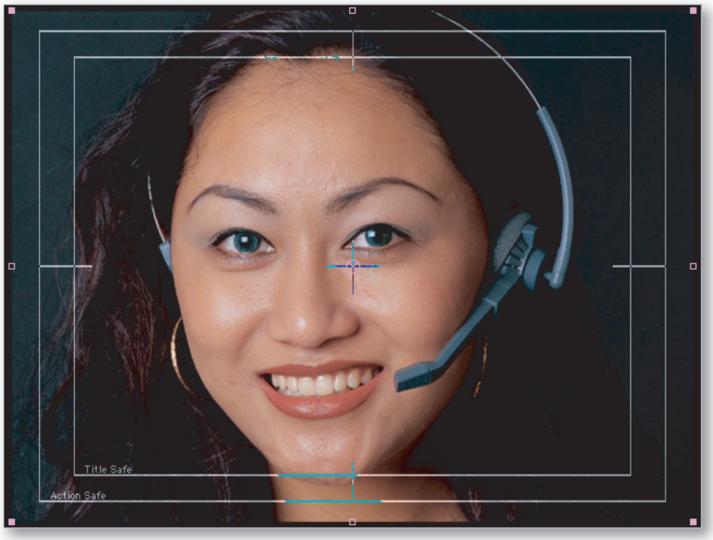
Cropping is excluding areas of the source frame from inclusion in the output frame. Content composed for video normally confines critical elements to what are called safe areas. The safe areas exclude the edges of the screen, which may be cut off or distorted on some televisions (especially older ones).

In some digital video editing applications, you can crop an image using a filter. In others, such as After Effects, the easiest way to crop an image is to scale the video up so the unwanted edges are no longer part of the frame. If you want an asymmetric scale, you can move the anchor point of the video as needed.

An example of scaling is going from 720 pixels wide source to 352 pixels wide output for VideoCD. The specified method for this scaling is to crop eight pixels from the left and right of the source, leaving a 704 pixel wide frame that you can reduce by exactly 50% to 352 pixels wide.

Another special scaling scenario is going between 486 line and 480 line sources in NTSC. Motion JPEG commonly uses 720 x 486 and MPEG-2 commonly uses 720 x 480. The different line count isn't due to being scaled differently but to six more lines of the video signal captured in the 486 line mode. When converting from 486 line to 480 line (for example, in DVD authoring), you'll need to crop six lines. Generally, four lines off of the top and two lines off of the bottom works well. Conversely, if you're going from 480 line to 486 line, you'll need to add six lines (again four lines on the top and two lines on the bottom is a good choice). Doing odd numbers can switch field order, resulting in substantial problems.

Safe Area



The inner boundary is the title-safe area, and the outer boundary is the action-safe area.

Scaling

Scaling is resizing the cropped source rectangle to the proper output frame size, for example, going from a 720 x 480 source frame to a 320 x 240 output frame for web video.

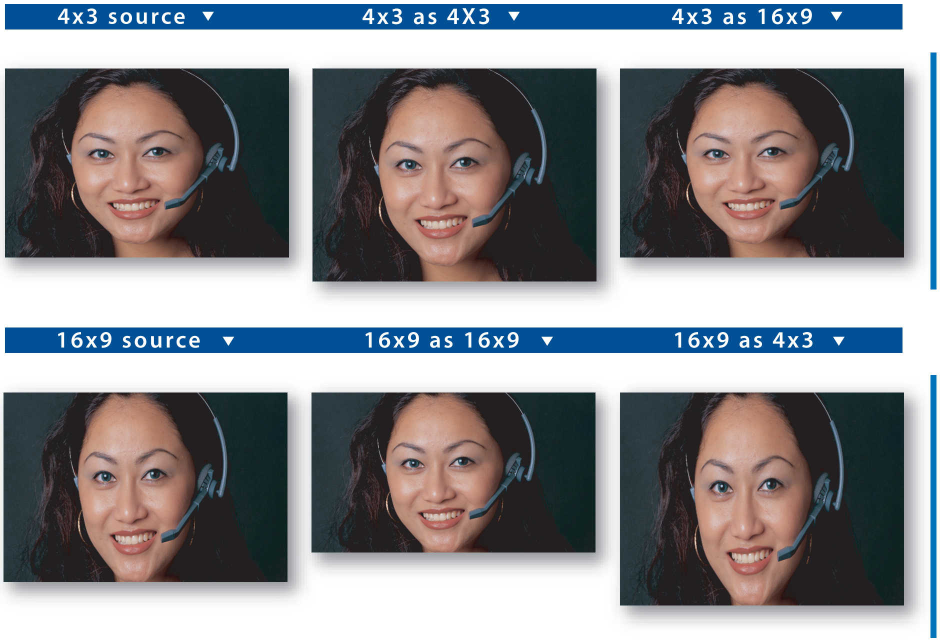
Scaling up on either axis forces a video editing application to make up data that's not in the source. It's always preferable to scale down on both axes if possible.

Aspect ratio correction

An important element of scaling is aspect ratio correction. Most production formats, like DV, use nonsquare pixels. For example, DV in NTSC is 720 x 480, which, by looking at pixels alone, would imply an aspect ratio of 3:2. However, DV isn't ever 3:2; instead, it's either 4:3 or 16:9, depending on how it was produced. In 4:3 mode, the pixels are more narrow than square, and in 16:9 mode, they're more wide than square. Nonsquare pixels are also used for MPEG-2 for DVD and MPEG-1 for SVCD and with different shapes for MPEG-2 for SVCD.

Conversely, most web video formats use square pixels, which makes it easier to do the math. For example, 320 x 240 is 4:3 in square pixels.

The key to getting aspect ratio correction right when you convert to square pixel sources from nonsquare pixel sources is to make sure that the output frame size matches the source aspect ratio. So, if you use a 4:3 source, a 4:3 frame size such as 160 x 120, 320 x 240, 400 x 300, 512 x 384, or 640 x 480 are all good choices, even if the source frame size is 720 x 480, 720 x 486, 640 x 480, or 352 x 480.



This graphic shows a 4:3 (top) and a 16:9 (bottom) 720 x 480 image; each transformed into a 4:3 and a 16:9 output. As you can see, getting the output aspect ratio correct is critically important.

Noise reduction

Noise reduction encompasses a variety of techniques to remove noise, especially grain, from an image, making it easier to compress. Very simple noise reduction algorithms are just blurs—they can hide grain, but can make the image quite soft as well. The commonly used Median filter is a good blur for noise reduction. More advanced algorithms try to blur only the parts of the image that have grain and may take advantage of differences among frames.

Tools such as After Effects Professional perform automatic or manually-assisted scratch removal and other processes to improve or repair degraded source. Normally these tools are stand-alone applications or plug-ins to a video tool that need to be used early in the workflow.

You don't always need to apply noise reduction. Very clean source isn't helped by noise reduction, and noise reduction may hurt the source audio. Typically, you should use only the minimum noise reduction required to suppress poor quality in the source.

Image processing

Image adjustment is the process of using controls, such as those found in image editing applications, to change the overall image. Because different color spaces are used by different codecs, image adjustment used to be required for most projects. Today, for the most part, modern formats and tools automatically take care of image adjustment, so it is mainly done to improve the suitability of the content for compression and isn't required for most projects.

Luma range expansion

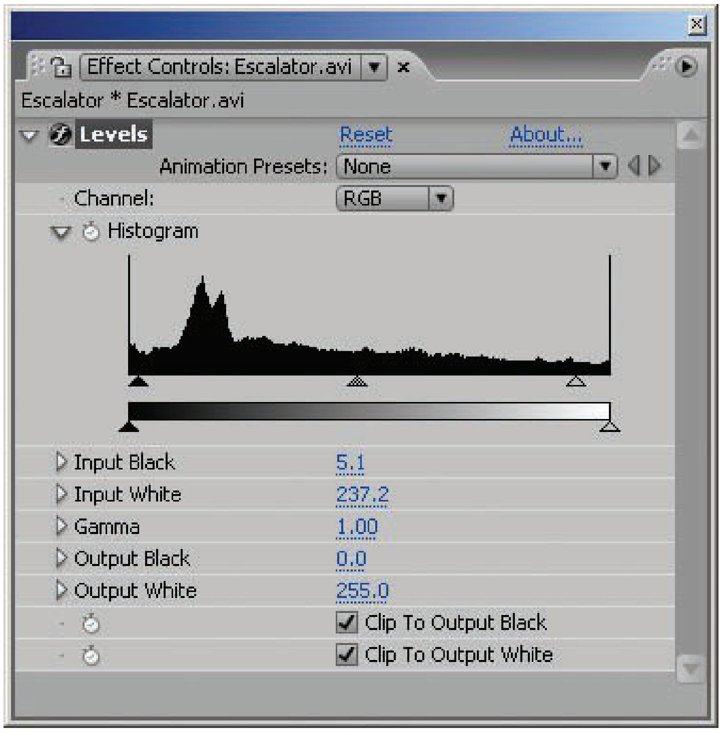
In the past, a major issue in video compression was making sure that black was pure black and white was pure white. Some tools, formats, and codecs considered black to be 0 and white to be 255, but others used the ITU standard of black as 16 and white as 235. Generally, modern tools automatically keep black and white consistent. But if blacks and whites appear either crushed or faded, there might be a luma range mismatch during the authoring process. Normally all RGB codecs use the 0-255 range, Y'CbCr codecs use the 16-235 range, and video tools automatically convert from Y'CbCr or RGB to the other. Beware of codecs that use studio RGB, which also uses the 16-235 range, because you may need to add additional contrast when converting to a delivery codec.

Black levels

Another goal of controlling luma is making black and white in the source mathematically black and white in the output. For elements like digitally rendered title cards, achieving this goal should happen automatically. But video captured from analog tape always has noise, which randomizes the luma values of pixels and leaves some black a higher value than pure black.

While you can get values down to black by either reducing brightness or increasing contrast, using both techniques gives you more precise control than doing only one or the other. Applying noise reduction before adjusting luma range can help you return some of the brighter black pixels closer to black, so you can use a less aggressive luma expansion.

The Levels filter with histogram in After Effects is invaluable for this type of processing. The key is to make sure that the region, which should be black, is being processed as black. If brightness and contrast aren't enough, you can also use a black or white restore filter. These filters set a threshold below which all pixels become black, and those pixels equal to or above it become white.



Animation Presets: None

Channel: RGB

Input Black 5.1

Input White 237.2

Gamma 1.00

Output Black 0.0

Output White 255.0

☑ Clip To Output Black

☑ Clip To Output White

The Levels effect in After Effects with typical settings to set the black and white levels beyond that of the noise floor and ceiling.

Gamma correction

Gamma is a measurement of how the 0-255 luma range turns into brightness when it is displayed. With different gamma values, 0 and 255 appear the same way, but values in the middle of the range look different. Different platforms (such as Mac OS, SGI, and Windows) have different default gammas. The net effect is that some files appear to have brighter midtones on a Macintosh computer than on a computer running Windows. However, most modern video codecs can automatically convert to the proper gammas upon playback.

Unfortunately, many older codecs that don't perform gamma correction are still in use. For content that you need to play back correctly on a platform other than the one for which it is authored, it's important to spot check samples to make sure they appear correctly.

Audio processing

Audio may only get a fraction of the bits that comprise video, but it is half the experience. While audio processing is generally easier than video processing, it is important to correctly process both.

Resampling audio

For low data rates, especially those targeting dial-up modems, you should reduce the audio sample rate to 22.050 KHz or lower. Modern video editing applications use algorithms for rate resampling, which provide much better quality than the algorithms of a few years ago.

Setting volume

The peak volume should be slightly below the maximum volume. Using a compressor/limiter to reduce dynamic range, you can sometimes improve audio quality at very low data rates. In general, television broadcast audio is easy to encode because the audio has already been processed through compressors and limiters.

Adjusting channels and bit rates

Mono content encodes better than stereo at lower data rates. Consequently, it's preferable to convert the audio from stereo to mono when you target lower bit rates (typically anything below 32 Kbps).

Given the choice of setting audio to stereo at a lower sample rate or setting it to mono at a higher rate, you should choose mono because sample rate matters more than stereo for the content sounds.

A number of sound systems also support 5.1 and 7.1 discrete surround-sound audio. The data rates required for surround sound (128 Kbps and above just for audio) generally preclude their use in real-time streaming, but they can be quite usable for disc-based playback.

Dolby Pro Logic surround sound (which stores surround sound in a stereo pair) doesn't survive the encoding process with many audio codecs. If Pro Logic encoding is required for a project, test carefully to make sure that the Pro Logic data survives the encoding process. With some codecs, raising the data rate or turning off options like Joint Stereo might help. Some codecs can't encode Pro Logic.

Reducing noise

There are also noise reduction filters for audio, although these filters are normally part of professional audio tools like Adobe Audition®, not a part of the compression tools. Providing clean, undistorted, hiss- and pop-free audio source results in better compression.

Encoding

If preprocessing is the art of compression, encoding is its science. Encoding means defining the correct parameters, such as data rate, frame size, and frame rate, to get optimum results for the project. Balancing these parameters can be difficult because there are many trade-offs. As always, it's most important to understand the goals that the compressed video is meant to achieve. This section starts with parameters that apply to all codecs and continues with parameters that are specific to video and audio.

Choosing a data rate mode

Different codecs and formats can provide different data rate modes. Some codecs support only one data rate mode, others provide a number of options. In most cases, there is a single optimal mode for a particular use.

Data rate modes include:

- **Average limited** is the most common type of data rate limitation. In this mode, you specify the average data rate, and the codec tries to make sure that the file ends up at the requested average. The codec varies the rate up or down to match the changing difficulty of the video.

  Most CD-ROM and QuickTime codecs work this way. Average-limited encoding is used for most progressive download files where file size and data rate are the limiting factors.

- **Peak limited** encoding is generally meant for hardware devices or high-data-rate content where the speed of the decoder (CPU, memory, or disk) is the limiting factor. Specifying a particular peak-limited data rate doesn't mean the peak of the file will actually be at the requested bit rate. Instead, the peak-limited data rate is the maximum rate supported by the decoder, but the encoder is free to use a lower rate if appropriate for the content.

  Peak-limited encoding also works well for high-resolution and high-data-rate web and CD-ROM files that need to play back on lower-end computers. Some codecs support encoding with both average limiting and peak limiting.

- **Buffer limited** encoding is the method of choice for real-time streaming. In buffer-limited encoding, the average data rate isn't applied to the overall file, but it is applied to any given section of video. So, when video is encoded with a three-second buffer, any arbitrary three-second chunk of the file has a data rate at or below the target data rate. Most streaming encoded files use both average-limited and buffer-limited encoding. Buffer-limited encoding is often called constant bit rate or CBR.

- **Quality limited** encoding is different from the other methods because it doesn't specify a data rate. Instead, it specifies a quality target, and each frame uses as many or as few bits needed to reach that quality target. This mode isn't compatible with real-time streaming. However, it can work very well for archiving content or for downloading mission-critical video where quality is more important than the file size or download time. For example, a stock video company may use quality-limited encoded files for short web samples to make sure that the content looks as good as possible and does not waste bits on unnecessarily high data rates that would make the content easy to encode.

  Most codecs that perform quality-limited encoding specify the quality on a 0-100 range. However, actual visual quality produced by the same value can widely vary among codecs. Some codecs support average-limited and quality-limited encoding together; the quality limit sets a minimum quality for each frame. If a frame falls below the quality limit based on available bit rate, the bit rate for that frame is raised so it meets the minimum. Typically, one or more frames after the low-quality frame are dropped to preserve the average bit rate. This mode sacrifices smooth motion in favor of image quality.

Choosing between encoding speed and quality

Many codecs offer a control that specifies a trade-off between encoding time and quality. The faster modes typically perform much less exhaustive motion estimation. Generally, big speed changes cause much smaller quality gains—an eight-times slower encoding process might only have 20% better compression efficiency. The correct trade-off between speed and quality varies depending on the project. For DVD projects, titles under an hour won't gain much by slower, higher-quality encoding because the data rate is so high. Web-distributed content almost always benefits from the slower, high-quality encoded files, because bandwidth is so often the limiting factor for web video.

Setting frame size

Frame size (also called resolution) is the height and width of the final video frame, measured in pixels. The data rate changes in proportion to the area of the frame (height times width). Thus, converting from 320 x 240 to 640 x 480 would require about four times the data rate, but converting to 192 x 144, you could reduce the data rate down to one-third. The effect isn't quite as strong as the math would suggest; because there are fewer pixels, pixel quality becomes more important.

Most codecs and formats require the height and width of a video to be divisible by two and give best compression efficiency when height and width are divisible by 16.

Specifying aspect ratio

An important facet of frame size is aspect ratio. Most web and CD-ROM formats use square pixels, and many formats only support square pixels. Conversely, the disc-based VCD, SVCD, and DVD formats only use nonsquare pixels. It's important to figure out the target aspect ratio first, and then make sure that the frame size you choose gives the correct aspect ratio given the picture shape in which the content is delivered.

4:3	16:9
160 x 112	208 x 120
192 x 144	256 x 144
256 x 192	336 x 192
320 x 240	432 x 240
384 x 288	512 x 288
512 x 384	688 x 384
640 x 480	848 x 480

Suggested frame sizes for 4:3 and 16:9 aspect ratios when working with square pixels.

Adjusting the frame rate

Frame rate is the number of frames per second at which the video runs. Higher frame rates appear smoother, and lower frame rates appear choppier. Anything below 20 fps doesn't appear smooth, and below 10 fps, the video looks more like a filmstrip than moving video.

Reducing or increasing the frame rate has much less effect than frame size on the data rate because reducing the frame rate requires encoding twice as much motion for each frame, a process that requires more bits for each frame. Also, each frame appears for twice as long, so there's more time to notice each artifact; thus, the image quality per frame must be higher to provide the same perceived quality.

Consequently, you should try to use the source frame rate if possible, and no less than half the source frame rate, unless you target real-time streaming to dial-up modem users. Raising frame rates generally pays off more than raising frame sizes in improving the user experience.

Note that NTSC frame rates are 0.1% lower than an integer number. So, "30 fps" is really 30/1.001, which is conventionally rounded to 29.97. A 29.97 fps output is more accurate than 30 fps when dealing with NTSC source. Also note that interlaced video has two fields per frame.

There is another very important rule when you change the frame rates: The output number of frames per second must be an integer divisor of the number of source frames per second. Therefore, the output can be one-half or one-third the source but not three-fifths. Good frame rates for each type of source appear in the table below.

FILM	PAL	NTSC
59.94	23.976/24	50
29.97	12	25
15	8	12.5
10	6	8.67
7.5	N/A	6.25
6	N/A	5
5	N/A	N/A

It's important to use frame rates that are evenly divisible by the source frame rate.

Negotiating frame dropping

Some codecs optionally or automatically reduce frame rate in order to maintain the quality of difficult frames and keep within the target average data rate. This process generally takes the form of a control that sets the minimum image quality per frame. If the bits that the codec would normally allocate to a given frame aren't sufficient to hit the target quality, it raises the data rate for the frame to the minimum required for quality; then drops one or more frames that follow to keep the average data rate on target.

Whether this process is appropriate depends on the content. For entertainment content, such as movies and especially music videos, having the video smoothly in sync with the audio is important, so frame dropping is not recommended. Conversely, presentation video, especially with a whiteboard in the background, might be better with many frames dropped in order for on-screen text to remain legible.

How keyframe rate comes into play

Keyframe rate controls the number of keyframes (I-frames) that are inserted in the video. Depending on the tool or the format, keyframe rate is measured in a keyframe every x frames or a keyframe every x seconds. MPEG-1 and MPEG-2 require a lot of keyframes; typically two per second. Web formats might have only one keyframe every 10 seconds.

Most formats also insert additional keyframes, called natural keyframes, when the video changes dramatically, for instance at scene cuts, or when there is extremely fast motion, such as an explosion. Natural keyframes reset the "keyframe every" value, so if a video has a lot of rapid cuts, it doesn't matter if the automatic keyframes are set to every 100 or 1000 frames—all of the keyframes are natural. For codecs that also create natural keyframes, the "keyframe every" value simply sets a maximum distance between keyframes.

How keyframes use 1-pass and 2-pass algorithms

Historically, most codecs worked in a single pass—they read the source file and wrote the compressed data frame-by-frame. The process worked, but required the data rate mechanism to anticipate the makeup of the future content. Many good heuristic algorithms have been developed for this process, but algorithms may guess wrong in some cases. The worst situations are when a 1-pass algorithm interprets the video as getting more difficult and raises the data rate to keep quality constant, or when another, more difficult sequence follows a few seconds behind the current frame. Just when the video gets the most complicated, the codec has to reduce the data rate below the average to compensate for the previously raised rate.

In a 2-pass algorithm, the codec first analyzes the video and builds a table of the relative encoding difficulty in each frame. It then calculates the actual bits per frame needed to achieve the highest possible average quality and applies these bits in the second pass. The net effect is the same as a quality-limited encode in which the correct quality value reaches the target data rate.

A benefit of 2-pass encoded files is consistent quality—the parts that are easier and the parts that are more difficult to encode match more closely. This consistency creates a better user experience than a video in which some scenes look great and others look terrible.

You can use 2-pass encoding with bit-rate-, peak-, and buffer-constrained files if they are supported by the format and the codec. With buffer-limited files, larger buffers give the algorithm more flexibility to distribute bits.

A few codecs employ 3-pass or N-pass systems. With these systems, the codec analyzes the results of the second pass, builds a new data rate allocation table, and encodes the file, usually with better results. In N-pass systems, the codec can repeat this process multiple times, although subsequent passes yield diminishing results. (Generally, a good 2-pass implementation is already very close to optimal.)

Modern formats such as Windows Media and RealMedia integrate 2-pass algorithms into the user interface. AVI and QuickTime codecs that support 2-pass have more difficulty making 2-pass encoding work. With AVI, the codec typically needs to create two different encoded files—one dummy file and one full file.

How sample rate affects the quality of audio

Sample rate is the number of audio measurements made per second, measured in kilohertz: 44.1 KHz for CD, 48 KHz for DV, and 8 KHz for analog telephone.

Anything less than 32 KHz sounds noticeably degraded compared to the source. For most broadband and CD-ROM audio you should use at least 32 KHz.

Unless the specifications of the target computers are available, use 44.1 KHz for computer playback. It is a safer choice than 48 KHz. Many sound cards, especially older and cheaper models, only play 44.1 KHz internally, so by using 48 KHz, you force a software sample-rate conversion. This conversion isn't as good as a high-quality filter that you can apply at compression, and so final output quality is less.. 48 KHz is a fine choice with audio source at 48 KHz or higher and an output computer or device that supports 48 KHz . A few codecs, most notably WMA9 Professional, can also encode at higher sampling rates, like 96 KHz.

Channels

For most codecs, the only choice for number of channels is between mono and stereo. Some codecs are mono only, some are stereo only, and some offer multichannel modes such as 5.1 or 7.1.

Older codecs, for example IMA, double the data rate when converting from mono to stereo. Modern codecs are able to reduce redundant data between channels, so the additional bit rate required to provide stereo at the same perceived quality is only 20-50% more than the mono bit rate, depending on the codec and how different the channels are.

Number of channels is less important to perceived quality than sample rate or the presence of audible artifacts. 32-KHz mono sounds better to most than 22.050-KHz stereo. Generally, you should use stereo only if you can use at least at 32 KHz—ideally, at least 44.1 KHz.

Multichannel (5.1 and 7.1) audio codecs also exist. The most common of these codecs are Dolby Digital (AC-3) for DVD and WMA9 Professional. Multichannel is supported in RealAudio 10.

Delivery methods for video

The method you use to compress content depends greatly on how you plan to deliver it. It's useful to understand the different environments in which people use video and the trade-offs each environment entails.

Disc-based delivery

CD-ROM was once the dominant format for digital video delivery. Since that time, however, web video has progressed from tiny slide shows to full-screen and full-motion video. Today, users can view video on mobile devices, like cell phones. The good news is that disc playback keeps getting better even as the focus of the industry moves to network delivery.

Disc-based delivery works well, and the available data rates are radically higher than those for web video. Even an 8x IDE CD-ROM can provide a sustained data rate of 8000 Kbps! The largest CD-ROM discs today are 800MB. DVD provides vastly more room: DVD-5 discs hold 4.7GB and DVD-9 discs hold up to 8.4GB.

When you target personal computers with modern, high-speed processors, disc-based playback can do incredible things. You can play compressed video at 1280 x 720 frame size on most computers purchased in 2002, and you can play 1920 x 1080 on the high-end computers of 2003. You can easily play video at 640 x 480 on computers several years old, especially if you use older, performance-tuned codecs (with a corresponding reduction in compression efficiency).

Downloadable files

The most basic type of video on the web is a simple, downloadable file. The file is downloaded to the user's hard disk like any other file.

The advantage of downloadable files is that users don't expect real-time playback during the download. The browser informs the user of the size of the file and often download time before it downloads. Also, because the file is copied to users' computers, they can pass the file among themselves, thereby reducing your bandwidth cost per viewer. The drawback is that users can't view any portion of the file until the download is complete. In this case, users may spend a lot of time downloading a file they would otherwise have realized in the first 10 seconds they don't want.

Downloadable files are often used for marketing or noncommercial use. They are very common in video game samples where very high data rates are required to encode game play in sufficient quality.

Progressive download files

During progressive download, a file is transmitted through standard web protocols such as FTP (File Transfer Protocol) or HTTP (HyperText Transfer Protocol), but users can view the part that has already been transmitted while the remainder is still downloading.

With most formats, the video starts playing automatically when the amount of time remaining in the download is less than the duration of the clip. This format allows the clip to finish downloading while it is playing. However, in some cases, the video playback catches up with the downloaded data, causing the playback to pause momentarily.

Because progressive download uses protocols like HTTP and FTP to ensure that all data packets arrive, it isn't affected by the video and audio glitches associated with real-time streaming. However, progressive download can't provide reliable real-time performance. A 400-Kbps file playing over a 200-Kbps connection requires nearly an additional minute of buffering per minute of video. Of course, that same 400-Kbps file couldn't stream over a 200-Kbps connection.

Progressive download is the easiest, cheapest, and most forgiving type of web video to author and deploy. It's a great way for users new to web video to start.

Real-time streaming

The next level of web video is real-time streaming. This type of transmission offers the traditional experience of seeing the video after only a few seconds of buffering, but it risks image glitches, poor quality, and limiting the data rate to the bandwidth available to the client at the time of viewing.

So, why put up with these limitations? Because real-time streaming provides real-time performance and random access to different parts of the file. Progressive download works well with movie trailers, but not for movies, especially if the user wants to skip to the last 20 minutes. With real-time streaming, the file starts playing in under a minute, no matter the length, and users can skip to the part they want. Also, most real-time streaming formats provide scalability, allowing streams to dynamically adapt to the user's connection speed.

The user experience of real-time streaming has improved significantly compared to only a few years ago. Codecs are much better and so are servers. Although users seldom think about servers, improvements in server and networking technology have greatly benefitted users. Modern servers are able to aggressively manage the buffer and send frames minutes before they're needed, which provides plenty of time to retransmit dropped packets and adjust bandwidth based on changing network conditions.

A common question is when to use real-time streaming and when to use progressive download. There are cases where the choice is obvious, and other cases where the choice is not so obvious. A good rule to consider is duration—clips under three minutes are often better as progressive, and clips over three minutes are often better as streaming.

One important feature of modern real-time streaming is scalability. Data rates on the Internet aren't reliable. Connectivity through a cable modem can substantially vary throughout the day. Many users share a T1 connection to the workplace, leading to rapid changes in available bandwidth per user.

It's not enough to measure the data rate at the start of a stream. Instead, the player and server need to communicate in real time to take advantage of all the available bandwidth without sending more data than can be received at any given time.

All of today's scalable formats work by putting multiple versions of the video and audio data as different streams inside the same file. These scalable solutions are often generically called multiple bit rate, or MBR. The details vary and can matter as much as the codecs themselves in providing a good streaming experience.

Choosing a compression format

Depending on the project, picking the correct format can be an easy or a difficult decision. Sometimes the customer mandates the format; other times the person performing the compression has to make a recommendation. Sometimes the project requires features that are only available in a particular format. Perhaps the requirements haven't been determined. After you choose a format, you need to determine the versions of players that support the format. This section guides you through some questions to ask when choosing a format.

- **Features** Often the biggest determining factor of a platform is the features that users need. If a format can't meet the project requirements, it's obviously not an option. In every project, different features vary in their degree of importance. Compression efficiency is almost always critical. Real-time streaming scalability is another major feature.

- **User base** Depending on the project and audience, user base may be very important. With most advertising content, users are unlikely to download a new player or codec just to watch a marketing clip. But for compelling content, users may consider making sacrifices. For example, Apple logged several million downloads of QuickTime in the 72 hours after a very popular movie trailer was released.

- **Platforms** Some formats are only available on a single platform or have different features depending on the type, or the version, of the operating system. Most of the modern media players are available for at least Mac OS® and Windows; some are available on Linux® and other operating systems. If you require very broad compatibility, an older standards-based format, like MPEG-1, might be appropriate.

In many cases, especially when you create content for the Internet, it may make the most sense to deploy content in multiple formats so that the user has the choice.

Still, most consumers of digital media have already installed the players for the major formats. While it can be tempting to offer variants in five formats at four data rates each, it is generally better to focus on as few formats as possible and make three optimized files instead of twenty low-quality files.

Exporting to Windows Media format

Windows Media is the Microsoft digital media technology. Originally introduced as NetShow, it has improved significantly, and the Windows Media 9 platform, released early 2003, is a popular proprietary digital media platform.

Using Windows Media for streaming delivery

Windows Media originally focused on real-time streaming, and even though it has become a more diverse technology over time, real-time streaming is still a core focus. Windows Media 9 has an excellent streaming architecture, excelling with MBR technology, servers, and codecs.

The MBR solution in Windows Media format is called Intelligent Streaming. Prior to Windows Media 9, it was limited, handling up to five video data rate bands per file, but all of these data rate bands needed to share the same resolution and audio track.

Intelligent Streaming was significantly improved with Windows Media 9. Now, it can use over a dozen data rate bands for both video and audio. The video bands can vary resolution as well, making it possible to provide a single link that can scale from audio-only for slower modems up to high definition playback for a local area network (LAN). The only significant restriction to Intelligent Streaming is that all bands must share the same codec and encoding mode (for example, Windows Media 9 with a 2-pass CBR).

To use the Intelligent Streaming options, you need to host content on a Windows Media 9 server running on Microsoft Windows Server™ 2003. If you need to host a video file on Windows Media Server 4.1 for Microsoft Windows NT® and Windows 2000, use the same resolution for all video bands and only use a single audio band. Older players (earlier than Windows Media 9) also need to use a subset of the available features.

Using Windows Media for progressive download

Windows Media 7 and earlier aren't suitable for progressive download, but this changed with version 7.1. Now the players reliably play partially downloaded material, and the encoders support download-optimized modes, such as 2-pass bit rate limited VBR.

Using Windows Media for disc-based delivery

Windows Media works with CD-ROM and DVD-ROM, as long as Windows Media Player or another Microsoft DirectShow-based player is used. However, because data rates are much less limited on discs, peak data rates that the computer's processor can decode may limit playback performance. The peak constrained VBR mode can be used in those cases to guarantee performance on the minimum playback platform.

The Macromedia Director CD-ROM authoring environment can't use DirectShow for video playback directly. To use Windows Media files in Director, you need a third-party plug-in.

EXTENSIONS

There are many different file types and extensions for Windows Media format. Here are some of the most common extensions and their meanings:

.asf
The file format used for Windows Media files is called the Advanced Streaming Format, and the original extension was .asf. However, this extension didn't discriminate between audio-only and video files, and so the extensions .wmv and .wma were introduced with Windows Media 7. Today, .asf files are typically legacy content using the Microsoft MPEG-4v3 video codec. However, the file format itself hasn't changed.

.wmv
A Windows Media Video file (.wmv) is an .asf file that contains video.

.wma
A Windows Media Audio file (.wma) is the same as an .asf file, but it only contains audio.

.asx
An .asx file is a metafile for Windows Media Streaming. The .asx file goes on the web server and points to the media on the streaming server. The .asx file format itself hasn't changed, but with the advent of WM7, the .wvx and .wax file extensions have been used more commonly.

.wvx
A .wfx file is a metafile that contains video and, optionally, audio.

.wax
A .wax file is a metafile that contains only audio.

There are a growing number of multimedia titles using high definition Windows Media 9 content on DVD-ROM discs. While discs require a fast computer for playback, Windows Media 9 is proving to be the first viable high-definition disc format.

Windows Media players

There is a wide variety of different Windows Media players available for different platforms, making compatibility a moving target. Recent versions of the player aren't always available for some common operating systems, a condition that requires you to provide multiple versions of the content or exclude some platforms. The following list describes the most commonly deployed players.

Windows Media Player 6.4 (Windows and Windows NT)

Windows Media Player 6.4 was included with Windows 98 and Windows 2000, and is the most recent version available for Windows 95 and Windows NT. Windows 95 doesn't have an appreciable market share for digital video consumers, but Windows NT is still common on corporate desktops and matters as a delivery platform. Windows Media Player 6.4 is very limited in general, both in performance and features. Connecting to the same content over the same connection, Windows Media Player 6.4 has lengthier buffering times and less reliable playback than newer versions.

Windows Media Player 6.4 includes Microsoft MPEG-4 version 3 as its most advanced codec. However, the player can automatically update codecs through Windows Media 9 and later. This update requires administrator privileges on the computer and a firewall that allows the download. There is also an enterprise deployment pack that can be used to provide the updated codecs in a network-wide system image. Windows NT systems typically run on older hardware, and so there may be performance issues playing modern codecs on those systems.

Windows Media Player 6.4 doesn't work with nonsquare pixels or interlaced video like Windows Media Player 9, but those features typically require a faster CPU than Windows NT computers are likely to be running on.

Windows Media Player 9 and later (Windows)

Windows Media Player 9 runs on Windows operating systems as old as Windows 98 Second Edition (SE); Player 10 runs only on Windows XP. These players support the full suite of features for Windows Media Video 9 (WMV9) and are the only versions that support new features, like multichannel audio, interlaced video, and nonsquare pixels.

Windows Media Player 9 for Mac OS X (10.1 and later)

The player for Mac OS supports the WMV9 and Windows Media Audio 9 (WMA9) codecs. Otherwise, the update is similar to older Mac OS players, which lack support for nonsquare pixels and interlaced video. Performance for low resolution content is quite good, but even the fastest G5 computers can't play high definition Windows Media Player 9 content. Note that none of the Mac OS X players support the older, but still common, ACELP.net speech audio codec.

Microsoft has no plans for further updates of the player. However, users can install Windows Media Components for QuickTime, by Flip4Mac™, to play Windows Media content directly in QuickTime Player or a browser.

Windows Media Player Mobile

The mobile players include a number of versions for mobile devices, such as the Pocket PC, Smartphone, and palm-size PC. The current version 10 mobile players support Windows Media Audio and Video codecs through version 9. Some features may not be fully supported, such as multi-channel audio, but then few users own Smartphones capable of surround sound.

Note that most mobile devices have slow processors and video performance, so resolution, frame rate, and data rate need to be much lower than for computer playback if you target a wide variety of devices. Ideally, you should target only a particular model so users can adjust settings for ideal playback. The new Smartphones are capable of streaming video over a mobile phone connection at fairly low data rates; downloadable music files can have a higher data rate. However, keep in mind that a mobile user is paying for airtime and may not want to download for longer than a few minutes.

For high-end Pocket PCs, reasonable settings for WMV9 are 320 x 240 at the native frame rate and 300 Kbps data rate. Any WMA9 audio is fine, up to 48 KHz stereo.

Pocket PC playback is normally local, but the Pocket PC player is capable of real-time streaming through a wireless Internet connection.

Windows Media Player for Pocket PC 2002

Windows Media Player for Pocket PC 2002 is only supported through WMV8. It offers lower performance because it was available on lower-performance hardware.

WMP for Pocket PC 2000

Windows Media Player for Pocket PC 2000 is similar to the player for 2002, but it only supports WMV7, not WMV8. Also, Pocket PC 2000 devices tend to be slower than 2002 devices, so even more conservative compression settings are required.

Windows Media video codecs

Windows Media has a broad set of codecs available, but the latest codec is considered to be clearly the best.

Windows Media Video 9 (WMV9)

WMV9 is a leading video codec that provides excellent results in a variety of uses from modem to high definition. It provides efficient compression and good speed for both the encode and decode processes.

So far, the biggest drawback to WMV9 is that it is only preinstalled with recent versions of Windows Media Player on Windows and Mac OS.

WMV9 is the first video codec in the Windows Media platform that supports interlaced and nonsquare pixel video. While not useful for web applications, this codec makes Windows Media much better for consumer electronics and video authoring workflows.

Windows Media Video 8 (WMV8)

WMV8 doesn't offer the compression efficiency of WMV9 (approximately 20% lower), nor does it have the nonsquare pixel or interlaced modes. However, it does feature the various 2-pass and VBR modes of WMV9. WMV8 is a good choice for most mainstream Windows Media projects if you don't know the player version in advance.

The advantage of WMV8 is that it is supported on almost all major platforms, including Windows 95 and Windows NT, Mac OS 9 and Mac OS X, and Pocket PC 2002. The codec is preinstalled on the Mac OS players, Pocket PC 2002 and later, and Windows Media Player for Windows 7.1 and later. If you don't need to support these older platforms, however, you can take advantage of the higher quality of the WMV9 codec.

Windows Media Video 7 (WMV7)

Most players that support WMV7 also support WMV8. One exception is Pocket PC 2000, which can't be upgraded to WMV8.

Microsoft MPEG-4 version 3

The older Microsoft MPEG-4v3 codec is less advanced than WMV8 and WMV9 in compression efficiency. However, it is the most recent codec available by default in the standard Windows Media Player 6.4. It is generally the oldest codec available in modern encoders.

ISO MPEG-4 version 1

The International Organization for Standardization (ISO) MPEG-4 codec is an implementation of the standard simple visual profile of MPEG-4. However, the Windows Media format doesn't support the MPEG-4 file format, streaming, or audio codecs. WMV7 includes the MPEG-4 codec to provide users with access to MPEG-4.

Windows Media Video 9 Screen

WMV9 Screen codec is optimized for screen recordings, like those used in software tutorials. Unlike screen codecs from other vendors, it has an explicit data rate control, which makes it work well for streaming.

Screen codecs work best for source that is like screen animation, in other words, RGB source where most of each frame is identical to the frame before it. It can also work for some types of flat color motion graphics.

Windows Media Screen version 7

Windows Media Screen version 7 is an older version of WMV9 Screen. It has somewhat lower compression efficiency, but otherwise it works similarly to WMV9 Screen.

Windows Media audio (WMA) codecs

Windows Media also has a variety of audio codecs optimized for different uses.

Windows Media Audio 9

Windows Media Audio 9 codec is the latest version of the WMA encoder, and its output is backward-compatible with the earlier WMA2 decoder. Any version of Windows Media Player still in use should be compatible with WMA9. Now that there are other codecs with the WMA9 prefix, the original WMA is now generally called WMA Standard, although it doesn't appear that way in encoding tools. In this document WMA9 refers to the standard codec.

WMA9 now supports all of the WMA9 bit rate modes. However, you should use CBR mode if you require compatibility with Windows Media Player 6.4. Note that the Bit rate VBR mode enables you to limit the data rate to between 48 and 192 Kbps. With the CBR mode, you choose from 5 to 320 Kbps. The Quality-based VBR mode doesn't provide a full 0-100 range, but offers a number of options that are selectable in a drop-down menu.

There is a 0 Kbps option for WMA9 CBR as well. This option is for making audio-free streaming files compatible with the legacy Windows Media Services 4.1 used prior to Windows Server 2003. Intelligent Streaming for that server requires each file to have one and only one audio track, so the 0 Kbps option provides a way to have an audio-free file.

ACELP.net

ACELP.net is a low-bit-rate speech codec. It sounds better than WMA9 for speech content below 24 Kbps. However, the decoder isn't available on all platforms—notably, Mac OS X and Pocket PC.

Windows Media Audio 9 Voice

WMA9 Voice is a low-bit-rate speech codec and replaces ACELP.net.

WMA9 Voice performs decently with music content at very low bit rates when played in the music or mixed modes. WMA9 Voice is also known as WMA 9 Narrowband. It is generally a better choice than WMA9 for music at 20 Kbps and below.

Note that WMA9 Voice is mono and has a CBR only with a maximum sample rate of 22 KHz and a maximum data rate of 20 Kbps. It's only appropriate for low-bit-rate applications.

Windows Media Audio 9 Professional

WMA9 Professional is a new codec for multichannel and high-resolution audio. It requires data rates of at least 128 Kbps, so it isn't appropriate for most web content. It was the first codec in one of the major architectures to support more than two audio channels, sample rates higher than 48 KHz, and bit depths greater than 16 bits. You should use it only with source that requires one of those features given its high-data-rate requirements. High-bit-rate WMA9 does just fine with stereo 16-bit sources at 48 KHz or less.

Multichannel WMA9 Professional plays back on systems with only stereo speakers, combining channels intelligently. It can also fold down (convert to fewer playback channels) well in other cases, as from 7.1 to 5.1, or to left, right, and center gaming systems. Still, multichannel is best when you anticipate a significant number of listeners who have multichannel audio playback.

WMA9 Professional supports all of the data rate modes of WMA9, including 2-pass CBR and all three types of VBR. As always, you should use 2-pass CBR for streaming and VBR for file-based playback.

Windows Media Audio 9 Lossless

The WMA9 Lossless codec is a mathematically lossless codec for archiving audio. Because a lossless compression can't guarantee any particular data rate, it is only 1-pass VBR. WMA9 Lossless works on the same players as WMA9 Professional. Lossless is overkill for delivery and you should only use it for archiving and mastering. It is also popular for digitally archiving audio CD content.

Exporting to RealMedia format

RealMedia is the oldest real-time streaming web format; the pioneering RealAudio debuted in 1995. RealNetworks has continued to enhance it; the updated Real 10 platform was announced in January 2004.

Unlike Windows Media, most RealMedia tools specify the total data rate from which audio data rate is subtracted to leave video data rate. However, some tools don't use this method—it's important to know the mode a particular tool uses.

Using RealMedia for streaming delivery

RealMedia was the first streaming platform. RealMedia works well in a streaming environment. Its MBR solution is called SureStream. Unlike Windows Media, it can mix and match codecs in different streams. All streams must share the same frame size, which can limit scalability.

Using RealMedia for progressive download

While RealMedia is mainly considered a streaming format, it has supported the excellent .rmvb 2-pass VBR progressive mode for several years. These files actually use a different extension than the standard .rm files, indicating that the files won't work when placed on a streaming server.

Using RealMedia for disc-based delivery

The .rmvb files also work fine from CD-ROM or DVD-ROM. While most software tools, such as Microsoft PowerPoint, don't allow RealMedia files to be directly embedded in them, the files play well from the disc in the RealOne player. RealMedia is also a good high-definition playback solution. Its codec has lighter decode requirements than WMV9, which allows high-resolution playback on lower-end computers. The new Real 10 also includes multichannel audio support.

RealMedia video codecs

RealMedia supports a smaller set of video codecs than the other formats. The most recent codec is always the best choice.

RealVideo 10

The new RealVideo 10 is a backward-compatible enhancement to RealVideo 9. Compared to the older codec, it offers significantly improved compression efficiency and a control to adjust compression quality and encoding time.

RealVideo 9

The RealVideo 9 codec is an excellent general-purpose video codec. It scales well from modem data rate and mobile device playback on the low end up to high-definition playback at the high end.

RealVideo 9 comes standard with RealOne Player (Windows, and Mac OS X) and is an automatic codec upgrade for RealPlayer 8 (the last version on Mac OS 9).

RealVideo 8

RealVideo 8 was the previous version of RealVideo. RealVideo 9 is almost as common as RealVideo 8 and offers better compression efficiency, so it is generally better to use RealVideo 9 instead.

RealVideo G2

RealVideo G2 was the first modern RealVideo codec. It is only required by old RealPlayer installations.

RealMedia audio codecs

Unlike other formats, a RealAudio codec is defined by a combination of a codec, data rate, sample rate, and channels. Instead of picking a codec and other parameters, the user picks an inclusive combination. Another twist with the RealMedia audio support is that many tools require audio codecs to be specified for video and music content, for both video and audio files and audio-only files. The user picks the one based on the mode the file was encoded with. It's important for the user to pick the correct setting for the mode that is actually being used.

RealAudio 10

RealAudio 10 uses the old RealAudio 8 Stereo Music codec for bit rates below 128 Kbps and AAC-LC for higher bit rates (Sony's ATRAC3 was used for higher bit rates in previous versions). The codec is backward-compatible to RealPlayer 8 at lower bit rates, but the player requires an update for higher bit rates. For more information about AAC-LC, see MPEG-4 audio codecs.

RealAudio 10 Lossless

This codec requires RealPlayer 10. Data rates are estimated to 700 Kbps for 44.1 stereo, so this codec isn't a web or streaming solution.

RealAudio 10 Multichannel

This codec is a 5.0 or 5.1 channel codec that requires RealPlayer 10. It requires high data rates, so it is not a streaming solution.

RealAudio 8 Stereo Music

RealAudio 8 Stereo Music is the leading general purpose audio codec in the RealMedia format. Its compression efficiency is better than the mono RealAudio Music codec, so it works well even for mono audio and at data rates as low as 12 Kbps.

At some data rates, the codec offers a "high-frequency" variant, which provides a broader frequency range. However, with complex audio, it can also result in more artifacts. You need to experiment to find the settings that work well for different types of content.

RealAudio 8 Surround

The RealAudio 8 Surround codec is a variant of the RealAudio 8 Stereo Music codec. It isn't actually a true 5.1 or 7.1 system. Instead, it stores a stereo pair. But it can maintain Dolby Pro Logic surround sound encoding, which the base RealAudio 8 Stereo Music codec can't do. While this codec doesn't provide the full multichannel experience, it is compatible with a wide range of equipment.

This codec is appropriate if the source is a Dolby Pro Logic surround sound mix. For normal stereo music, the RealAudio 8 Stereo Music codec provides better quality and more flexible data rate options.

RealAudio Speech

The RealAudio Speech codec is a relatively standard speech codec; it is intelligible at the lowest data rates and can sound excellent at higher data rates. Unlike most speech codecs, it provides 32- and 64- Kbps options with excellent fidelity for content such as audio books.

RealAudio Stereo Music

The RealAudio Stereo Music codec is the original RealAudio stereo codec from the G2 release. It offers lower compression efficiency than the RealAudio 8 Stereo Music codec. This codec provides extensive backward-compatibility.

RealAudio Music

The RealAudio Music codec is the original RealAudio mono codec from the G2 release. Even for mono content, the RealAudio 8 Stereo Music codec provides better quality. However, the RealAudio Music codec is available at lower data rates than the RealAudio 8 Stereo Music codec, so it can be useful in some cases for low bit rates.

Exporting to QuickTime format

QuickTime is the oldest of the mainstream digital media formats; it was introduced in 1991. It predates the modern, widespread popularity of the Internet and even the CD-ROM drive as standard computer equipment, and has matured to support a wide variety of applications (although it is less advanced than RealMedia and Windows Media for real-time streaming). The greatest strength of QuickTime is that it supports rich interactivity and rich media.

By default, QuickTime data rate is set as total kilobytes per second, not the total kilobits per second of RealMedia or the kilobits per stream per second of Windows Media. However, many tools that support multiple formats provide the rate in Kbits. For consistency, data rates are listed in Kbits below.

Note this section is about the QuickTime format (.mov), not about the QuickTime architecture, which can author and play back many other formats, including MPEG-4 and AVI.

MOVIE ALTERNATES

QuickTime takes a very different approach to scalability than Windows Media and RealMedia formats. Instead of making a single file that contains multiple versions of the data at different data rates, QuickTime uses multiple files at different data rates, which are switched when the movie begins playing back.

There are several drawbacks to this approach. First, the switching only takes place when the file begins playing, so there is no switching supported during playback. Second, authoring is more complex—only a few advanced compression products, like Adobe Media Exporter, support authoring the alternate movies while encoding. Third, QuickTime relies solely on the user to correctly configue QuickTime preferences to determine the data rate. So if a user sets up QuickTime on a laptop with a broadband connection at work, and then connects by means of a dial-up modem from a hotel room, QuickTime tries to transmit the broadband version of the file and either fails dramatically if it is real-time streaming or causes a very lengthy download for progressive files.

For these reasons, it is best to offer users direct links to each media file, even if you also offer a general movie alternates link.

There are some unique features of movie alternates. For one, they work with progressive files, not just streaming files. For another, they can filter in more parameters than just data rate. You can provide different versions for Mac OS and Windows users, for computers using different languages, and for computers with different CPU speeds.

Using QuickTime for real-time streaming

QuickTime doesn't have a real-time scalability solution, like SureStream or Intelligent Streaming. QuickTime can be a poor fit for real-time streaming on the Internet because data rates are so variable. However, QuickTime can work well in an intranet environment. One advantage is that the dedicated streaming server is the free, open source Darwin Streaming server, available for a vast variety of platforms.

Streaming support was added to QuickTime relatively late in its development. Unlike the other formats in which files are always streamable, in QuickTime, you have to add hint tracks to files you want to stream. This extra data gives instructions to the server about how to stream the data. Note that the data itself isn't sent, so the file size of a hinted movie is larger than the amount of data actually being sent to the client.

Using QuickTime for progressive download

QuickTime pioneered progressive download, and this format remains the most popular way to distribute web video with QuickTime.

One complexity with progressive QuickTime movies is that they need to be fast-starting movies. Every QuickTime movie has a movie header, which contains information about the file's structure and is required for playback. Whenever you modify a file, QuickTime moves the movie header to the end of the file, which means that it won't start quickly and only the QuickTime logo will play until the entire movie is downloaded. Even editing metadata, like the movie title, can force this change. The simplest way to move the header to the beginning of the file is to perform a "Save As" from within QuickTime Player Pro.

Don't use hint tracks with progressive files because they aren't used, and they increase the file size. Instead, use the Compressed Movie Header mode with progressive files because it makes the header smaller.

Using QuickTime for disc-based delivery

QuickTime is a great solution for CD-ROM and other disc-based architectures. The QuickTime user interface makes it very easy to add QuickTime playback to applications, and QuickTime is the best supported format inside Director. However, PowerPoint on Windows doesn't natively support QuickTime.

QuickTime video codecs

Given the long heritage of QuickTime for flexibility and support for third-party codec vendors, the range of codecs supported in QuickTime is broader than in the other formats. However, the best codecs are often not provided for free.

Sorenson Video 3

The highest quality codec available for QuickTime today is Sorenson Video 3 from Sorenson Media. It is a very flexible codec and can work in both progressive download and streaming environments. The basic version doesn't include a variety of advanced features that improve compression efficiency and streaming performance.

QuickTime comes with the free, basic version of Sorenson Video 3. However, for professional use, the Sorenson Video 3.1 Professional upgrade is available. Sorenson Video 3.3 Pro offers improved compression speeds, especially on dual-processor computers, and improved quality, especially with real-time streaming protocol (RTSP) streaming. The Pro version is necessary for any commercial QuickTime encoding. The Pro encoder makes files fully compatible with the bundled decoder—no user upgrade is required to view the files.

Sorenson Video 3 Pro options

Sorenson Video 3 has been included with QuickTime since version 5.0.2, which was released in the summer of 2001.

- **Quick Compress** This option slightly speeds up the encoder at the expense of slightly reducing quality. Because neither effect is significant, you can almost always leave this option deselected.

- **Bidirectional Prediction** To use B-frame encoding in Sorenson Video 3 and add compression efficiency and scalability, choose this option. With scalability, the B-frames can drop if the processor isn't fast enough to play back all of the frames or if the data rate isn't sufficient to stream all the frames by means of RTSP. This option automatically drops frames and reduces CPU requirements by at least 50% and reduces data rate by about 25%, at the cost of cutting the frame rate in half. This option is especially useful in streaming because without B-frames, the server drops all frames except the keyframes when under bandwidth stress.

  So, it would be ideal to use bidirectional prediction in all cases. However, there is one problem with B-frames—they cause video to be delayed two frames relative to audio. At 29.97 fps, this delay is only mildly distracting, but at lower frame rates, the video can appear dubbed. Also, B-frames can sometimes cause instability in versions previous to QuickTime 5 (this problem appears to have been resolved in QuickTime 6).

  There is an AppleScript for Mac to fix the sync in a file that has already been encoded.

- **Rate Control Mode** The unlabeled rate control box for this option has three additional options. The first option performs a standard CBR encode.

  The Sorenson One-Pass VBR option loosens up the data rate. This mode is optimal for encoding for RTSP and for most non-RTSP encoded files where peak data rate isn't a concern. Note that if the Streaming option is selected, the VBR window is constrained to the appropriate value for streaming.

  The Minimum Quality option sets a minimum quality threshold for each frame on the 0-100 scale. If the Drop Frames To Maintain Data Rate option is selected, the frame rate is reduced when the quality threshold is exceeded, and the data rate is maintained. If this option is not selected, the data rate rises when the minimum quality isn't met, meaning that the final data rate is overshot.

- **Image Smoothing** This option activates a deblocking filter, which reduces the blockiness of the image. (A blocky image contains rectangles, or blocks, of unshaded colors). It doesn't have much effect at higher data rates, and it is automatically deactivated on playback if the computer isn't fast enough to run the operation. Generally, you should activate it.

- **Media Key** The Media Key option assigns a password to the video that the player application needs to provide for the video to play back.

- **Streaming** You should select this option for all files that will be RTSP streamed. The default settings are appropriate for most projects.

  Don't modify the Slice Picture Into Packets Of Bytes option. It needs to be modified in sync with the hinting settings, and except in some private networks, it isn't helped by other values.

  The Force Block Refresh Every option ensures that every 16 x 16 block of the video is refreshed at the specified interval. This refresh helps with error recovery (any glitch from a dropped pocket is erased at least that often). By using this option, you can lower the number of keyframes in the file. This option slightly reduces compression efficiency with static images (which isn't as important), but it doesn't have much of an impact with fast motion (where blocks are naturally refreshed by motion).

- **Masking** Use this option to include a real-time alpha channel in the video. The codec can either use a provided alpha channel, or it can generate one itself based on an internal chroma-keyer. The internal keyer is limited, so using a professional tool like After Effects to generate the alpha channel can offer much better results. To use an alpha channel in the source files, you need to set the color to Millions+.

- **Watermarking SV3.1 Pro** If you want to add a real-time composited watermark to the video, use this option. This watermark can provide much higher quality than adding a watermark to the video source because the fine details of a source watermark are often lost in compression.

The watermark can have opacity, or it can be colored. Note the file format for the watermark is very finicky—carefully read the section in the codec manual describing how to make the watermark. If you use Photoshop to convert an alpha channel to a Sorenson watermark, a good starting point is to apply the existing alpha channel to a 50% gray background and then flatten the image.

Sorenson Video

QuickTime also includes the older Sorenson Video codec (no number is listed, but it's version 2). This codec is completely different from the current Sorenson Video 3 because it has lower compression efficiency and quality. You can use it if you require compatibility with versions of QuickTime prior to version 5.

There was an upgrade of the original Sorenson Video codec called Sorenson Video 2.2 Developer Edition, but it is no longer commercially available.

MPEG-4

QuickTime includes a MPEG-4 codec. This codec is available by default to all Mac OS encoders. However, it is only available on a few Windows encoders whose developers have paid a license fee to enable it.

QuickTime supports the Simple Visual specification for playback, and you can use third-party MPEG-4 encoders to make a file that plays back QuickTime. Third-party encoders generally offer better compression efficiency, with techniques like 2-pass VBR encoding.

Sorenson Video 3.1 Pro offers better compression efficiency than MPEG-4 and better streaming performance. The biggest advantage of MPEG-4 in a QuickTime file is the decoder speed, which is faster than Sorenson Video 3's.

MPEG-4 has been included in QuickTime since QuickTime 6, which was released in the summer of 2002.

H.263

H.263 is a standard video conferencing codec that has been included with QuickTime since QuickTime 4. It streams well and provides good streaming performance. It can (rarely) beat the free version of Sorenson Video 3 for streaming, but it isn't as good as Sorenson Video 3.1 Pro.

One unique feature of H.263 is that it only internally uses the three listed resolutions. If you pick a different resolution, it uses the next smallest resolution internally, and scales the video to the correct resolution in real-time. If desired, you can assign the specific resolution, but this assignment isn't needed in most cases.

The Cycle Intra Macroblocks feature performs a similar function to Sorenson Video 3's Block Refresh and improves streaming performance, especially in networks in which data loss from compression is a problem. It should be active only for RTSP files.

Cinepak

Cinepak was the dominant CD-ROM codec for most of the 1990s. It has been included in QuickTime since version 1.5 over a decade ago. Cinepak's hallmark was a very fast decoder—video at 320 x 240 played back well on 60 MHz computers. Still, the simplicity of the encoder results in compression efficiency well below that of modern codecs. Cinepak is one of the few delivery codecs that supports a native grayscale mode, which gives faster decode performance with black-and-white sources.

The main use of Cinepak these days is for high-resolution playback on old, slow computers, like those often seen in K-12 schools. A video at 640 x 480 and 29.97 fps can play without issues on a five-year-old computer, albeit at data rates of at least 5000 Kbps.

VP3

On2/The Duck Corporation has been making commercial codecs for a decade. Their more recent VP5 and VP6 codecs aren't available in commercial products, but the VP3 codec is open source and is freely available for QuickTime. While it isn't part of the standard install, it's available through component download, which means QuickTime automatically offers to download the codec when it detects content that uses it. To force an install, run the QuickTime Updater utility—the VP3 codec is one of the third-party options.

The VP3 codec doesn't have a native packetizer, so it isn't an appropriate tool for RTSP streaming. Also, the VP3 codec is several years old and has compression efficiency below that of Sorenson Video 3.3 Pro, and also has unpredictable data rates, so it isn't a good fit for web distribution. However, the decoder is faster than Sorenson Video 3, making the VP3 codec a viable option for high-resolution CD-ROM and kiosk work.

Because the VP3 codec is open source, different implementations vary. For example, the DirectShow version of the codec for making AVI files on Windows has a few options that aren't currently available in the Mac OS encoder, although the Mac OS player can play files encoded with those features.

The Allow Dropped Frames option in the VP3 codec instructs the encoder to drop frames to maintain data rate. The VP3 codec has a fairly high minimum quality, so when this option is not selected, the target data rate might be substantially exceeded. Still, dropped frames are distracting, so the VP3 codec isn't the best one to use for high compression efficiency.

The Quick Compress mode substantially speeds up the encode with only a minimum hit in quality. Unlike other codecs, it's safe to run the VP3 codec in Quick Compress mode by default. Still, if the time is available, encoding with this mode inactive guarantees the best quality and most accurate data rates possible.

VP3 has a more complex keyframe insertion mode than most codecs. Not only can you set a threshold, but you can specify the minimum and maximum number of keyframes. The default settings for Threshold and Minimum work well for most projects. If your video includes very fast editing, you might want to set a lower minimum, and if you're working with very high frame rates, you can set a higher maximum. Make sure that the keyframe rate set in the encoding application outside of this dialog box is greater than the maximum number of frames.

Animation

Apple's Animation codec is an interframe Run-Length Encoded (RLE) RGB codec. Hence, for good compression efficiency, it relies on long horizontal lines of exactly the same color for each frame. The Animation codec doesn't have any data rate control; it only has quality sliders. Hence, it's not appropriate for encoding video originated as natural images where two adjoining pixels are rarely identical. It can do a good job with recorded screen captures, although there are some workflow tricks required to do those well, which are outside the scope of this document.

The Animation codec is lossless when the quality slider is at 100%. At lower levels, it simply makes horizontal lines that are almost identical. This process can produce poor quality with many sources, which results in visible banding. Note that despite the name, the Animation codec isn't very good at encoding three-dimensional optimization, where lighting effects prevent regions of identical pixels.

You can also use the Animation codec to store intermediate files. The Animation codec stores intermediate files well when using an RGB workflow with RGB native tools, like After Effects, especially with synthetic images. However, the data rates the Animation codec produces are not much better than uncompressed data rates for natural images. Also, some compression tools don't work well with interframe encoded files, so you should use a keyframe in every frame in those cases. The Portable Network Graphics (PNG) and Microcosm codecs provide better compression with identical quality in those cases.

Graphics

The Graphics codec is a lossless RGB codec, like the Animation codec. It's different because it offers better compression efficiency, but it only works in 8-bit color. It delivers screen recordings well, but only if you convert the file to 8-bit first, which is a tricky process.

PNG

PNG is an RGB-only codec that was originally developed as a patent-free alternative to GIF. However, it evolved to have better compression efficiency and support for more than 8-bit index color. It is very efficient at encoding synthetic content with areas of flat color, like screen shots. It isn't useful as a delivery codec for moving video because it is interframe only, and it doesn't have a data rate control mode. PNG supports the Millions+ mode, so it can transport an alpha channel.

PNG's interlaced mode isn't for interlaced fields; instead, you use it for progressive decoding of the image. When you choose this option, the image initially shows a lower resolution proxy while it downloads. This codec does not work well for archiving because it slightly reduces compression efficiency.

PNG is always lossless, but the Filter mode controls the trade-off between encoding time and final file size. For maximum compression efficiency, use the Best option. For maximum speed while still getting some compression, use the Up option.

Microcosm

The Microcosm codec is another lossless RGB interframe-only codec. Its big advantage over PNG is its support for 16 bits per channel, so this codec can perform lossless archiving of content when you use it with tools that support 16 bits per channel, like After Effects. This 16-bit support really excels when delivering to film or broadcast, especially when working with Cineon or a source with more than 8 bits per channel. The Microcosm codec can also transport an alpha channel.

A free decoder for the Microcosm codec is available for download, but you need to purchase the encoder.

None16

None16 is a free codec from the creators of Microcosm. It offers 16-bit-per-channel RGB and an alpha channel. However, it is fully uncompressed, so the file sizes are very large.

Photo-JPEG

Photo-JPEG is a codec implementation of the standard JPEG still-photography codec. It's a lossy codec that supports 8-bit-per-channel 4:2:0 Y'CbCr, or grayscale; however, it is visually lossless at quality settings above 85 for most content (Photo-JPEG lacks an explicit data rate control).

Photo-JPEG can be a good way to deliver content for slide shows, but Motion JPEG is more suitable for most video archiving and transport uses. Photo-JPEG doesn't recognize fields, so it works best with progressive content.

Motion JPEG

Motion JPEG is a 4:2:2 variety of JPEG with support for fields. It's the native format of many NLE systems; many of which can perform lossless transcoding from their own native format to the generic QuickTime Motion JPEG codecs.

There are two types of Motion JPEG codecs: A and B. These designations originally corresponded to different chipsets supported by early NLE cards. They're functionally equivalent; so it's probably easiest to use A. You need to manually set the correct field mode and field dominance (which is field order mislabeled) in the Options dialog box.

Component

The Component codec is an uncompressed Y'CbCr 8-bit-per-channel 4:2:2 codec. Because it's uncompressed, it is extremely fast, but data rate requirements can be substantial. It is mainly used as an intermediate format for users with a lot of disk space. The Component codec does not support an alpha channel.

None

The None codec is an uncompressed RGB codec. It offers all of the RGB QuickTime color depths, including alpha channels, except for 16-bit-per-channel modes. Unless you need pure speed, you rarely need to use the None codec, given the variety of lossless RGB codecs in QuickTime.

QuickTime audio codecs

QuickTime audio codecs don't have a specific data rate control in their main user interface. Instead, you set the data rate, if controllable, in the Options dialog box.

MPEG-4

The best general purpose audio codec in QuickTime today is MPEG-4 AAC-LC. Encoding is only included by default on Mac OS. Windows users don't have access to this codec in most applications. The MPEG-4 codec lists options for AAC-LC (a general purpose codec) and Code Excited Linear Prediction (CELP), used for low-bit-rate speech. CELP isn't actually available after QuickTime 6.5.

Like MPEG-4 video, MPEG-4 audio is based on the open MPEG-4 standard. You can use a third-party AAC-LC encoder to make a file that you can import into QuickTime and use as a video track. MPEG-4 has a native packetizer and works well for both real-time streaming and file-based delivery.

AAC-LC is a very efficient modern codec. Data rates above 128 Kbps, even for 48-KHz stereo, are overkill for most content; music is good enough to dance to at 48 Kbps and 44.1 KHz mono, and sometimes below those measurements.

Note that the codec automatically converts from the sample rate specified in the Audio dialog box to the rate specified in the Codec dialog box. If you are encoding to AAC-LC at either 44.1 or 48, set the sample rate in the Audio dialog box to match. If you are using a lower sample rate, set the sample rate in the Source dialog box to match that of the source (generally 48 KHz). Avoid doing two sample rate conversions.

The Encoder Quality option sets the quality and speed trade-off for the codec. For a 16-bit source, Better is the appropriate default. Best only helps with sources more than 16-bit and is slower than Better.

QDesign Music 2

QDesign Music 2 (QDM2) was the leading streaming audio codec before QuickTime 6 and MPEG-4. It remains the best codec available by default in QuickTime for Windows. The QDM2 codec has a native packetizer and it is a fine option for streaming.

QDM2 does very well with instrumental music, but it can have objectionable artifacts when encoding speech, even at higher data rates. If it creates artifacts, try reducing channels from stereo to mono, and then try reducing the sample rate. The Pro version and data rates greater than 48 Kbps are required to compress 44.1 stereo with mixed content. The free version of QDM2 is limited to 48 Kbps maximum data rate.

QDesign Music 2 Pro

The Pro version of QDM2 offers better compression efficiency, much faster encoding speeds, and data rates greater than 48 Kbps. The basic version of QDM2 supports 8-48 Kbps. The Pro version can compress up to 128 Kbps.

This codec includes suggested sample rate and channel combinations. These suggestions work well for instrumental music but are overly aggressive for content containing voices. You can use the codec in either Quality or Speed mode. The Pro version is much faster than the Basic version, and so you should use the Quality mode.

For most content, you can use the default Attack and Spread settings of 50/50. If the compressed content contains a lot of artifacts, adjusting these settings on a trial-and-error basis can sometimes yield better results.

Qualcomm PureVoice

Qualcomm PureVoice is the codec that Code Division Multiple Access (CDMA) cell phones use, and it sounds a lot like a cellular phone. This codec is quite intelligible with speech, but it has difficulty with other types of content. The PureVoice codec supports mono audio only.

The codec works best at 8 KHz, especially with real-time streaming. If you have sufficient data rate to run it at 22 KHz or greater, a different codec would likely provide a better overall experience.

PureVoice has a native packetizer, so it is a fine streaming codec. The packetizer works best at 8 KHz, so you should use that sample rate for streaming projects.

The Qualcomm PureVoice codec has two modes. Full Rate mode offers a 9:1 compression over pulse code modulation (PCM), so 8 KHz is around 15 Kbps. Quality is similar to a cell phone with a good connection. Half Rate mode offers a 19:1 compression over PCM, so about 7.5 Kbps at 8 KHz. While speech should be intelligible, quality is noticeably degraded from the Full Rate mode. The quality is similar to a cell phone with a weak connection.

The Optimize Compression For Streaming option instructs the hinter not to transmit blank data in each audio block when streaming. This option can somewhat reduce real-world data rates, but it performs this reduction unpredictably. You should activate it only for RTSP.

IMA

IMA is an old CD-ROM codec created by the Interactive Multimedia Association. It offers a 4:1 compression over the None mode. Quality is good at 44.1 KHz, but it lessens at lower levels. IMA is useful because of its very fast decoder and wide compatibility (ranging back to QuickTime 2.1 and it works in AVI as well). It is a decent option for CD-ROM and kiosk uses where playback performance is more important than compression efficiency.

QuickTime also supports the IMA Adaptive Differential Pulse Code Modulation (ADPCM) variant created by Microsoft. It's effectively the same codec, but it is more compatible with older AVI players.

MP3

While QuickTime doesn't include its own MP3 encoder, it has a very capable MP3 decoder, and it can play MP3 files encoded by a wide variety of tools. Some encoding tools have native support for encoding MP3 as an audio track.

A good MP3 encoder offers better performance for mixed content than any other QuickTime codec, other than MPEG-4. Quality is quite good at 128 Kbps for 44.1 stereo, and compression is transparent for most listeners at 160 Kbps.

QuickTime 4.1 and later can handle VBR MP3 without problems. However, VBR MP3 causes QuickTime 4.0 to shut down, so you should use CBR only if you require compatibility with QuickTime 4.0.

Although there is an official RTSP mode for MP3, QuickTime doesn't support it. You shouldn't use MP3 for RTSP streaming.

None

The None codec is an uncompressed PCM audio track. It is rarely appropriate for a delivery codec, but it is the best option for intermediate and transport use. The None codec supports 8 bit and 16 bit, but you should only use 16 bit.

24- and 32- bit integer

QuickTime also supports PCM in the higher sample depths of 24 and 32 bit. These codecs are for transporting between two tools that support these higher sample rates.

Floating point

The floating point codecs are occasionally used to export audio to some high-end audio or signal processing tools. You wouldn't use these in a normal workflow.

Exporting to MPEG-1 format

MPEG-1 was the first open multimedia standard. The original vision for MPEG-1 for computers was that users would buy add-on MPEG-1 decoder boards for their computers for playback. However, this vision didn't become popular, so MPEG-1 playback didn't happen widely until computers became fast enough to perform all of the decoding in software.

VideoCD, a precursor to DVD, also used MPEG-1. While it didn't get used much elsewhere in the world, VideoCD became enormously popular in Asia.

MP3 is also technologically related to MPEG-1.

Using MPEG-1 for progressive download

Although its compression efficiency may only be half that of today's cutting edge web codecs, MPEG-1 still plays a role in progressive downloads where widespread compatibility is more important than compression efficiency.

MPEG-1 data rates are higher than other formats. A good option is 256 x 192 video at 336 Kbps coupled with 64-Kbps, 32-KHz mono audio. That data rate produces decent quality with lots of content when coupled with a good 2-pass VBR MPEG-1 encoder. MPEG-1 for the web should always be square pixel.

Using MPEG-1 for real-time streaming

MPEG-1 does a reasonable job with real-time streaming, and is supported by Apple's free Darwin Streaming Server and in a variety of products from Cisco. Note the compression efficiency of MPEG-1 is not as high as other modern codecs.

Using MPEG-1 for CD-ROM delivery

MPEG-1 is often used for CD-ROM work, where decode performance and compatibility is more important than compression efficiency, especially for cross-platform CD-ROM projects where it isn't possible to install QuickTime. MPEG-1 is the only good cross-platform file format for PowerPoint, which can't easily use QuickTime files on Windows or Windows Media on Mac OS.

MPEG-1 encoded for CD-ROM should be encoded in square pixel because older versions of Microsoft DirectShow handle nonsquare pixels incorrectly.

Using MPEG-1 for VideoCD delivery

The VideoCD format is mainly popular in Asia, but it's also playable on most personal computers without additional software, and in DVD players.

Note that VideoCDs, like DVDs, come in PAL or NTSC formats. A computer can play both formats, but a set-top DVD or VCD player may only play discs in its own format.

VideoCD has a very tightly constrained set of parameters. Resolution is 352 x 240 in NTSC and 352 x 288 in PAL, and data rate is also locked. PAL is always 25 fps and NTSC can either be 29.97 fps for video or 23.976 fps for film sources. The data rate is locked to a CBR rate exactly that of an audio CD, so for every minute of audio a CD can hold, it can hold one minute of VideoCD video.

Note that you can make VideoCD on CD-R, but CD-R VideoCD doesn't work on some early model DVD players due to an incompatibility with the laser. Manufactured CD-ROM media works fine.

There are many VideoCDs in use, either older discs or discs encoded with a cheap real-time or software encoder. Modern professional grade encoders can be quite good. So while the resolution might be low, the image and sound quality can come out nicely with a variety of content.

MPEG-1 video and audio codec

MPEG-1 only supports one video codec, also called MPEG-1. This codec also includes three audio modes: Layer I, Layer II, and Layer III.

MPEG-1 (for video)

MPEG-1 offered incredible compression efficiency for its day, and modern encoders can perform well at data rates far below the data rates the format's creators thought possible. MPEG-1's biggest barrier is that it lacks postprocessing, so at low data rates, it gets blocky instead of soft, which results in a relatively high minimum bit rate for decent results. 2-pass VBR substantially helps MPEG-1 and is available in most implementations.

MPEG-1 (for audio)

In practice, only the Layer II audio mode is used. Layer I had lower compression efficiency. Layer III never became standard as part of the decoders. However, it continues to use the name of MP3. Layer II offers compression efficiency about 50% less than MP3. VideoCD uses 224 Kbps, and generally sounds quite good. Using lower Kbps values can also work well for web content.

Layer II supports joint stereo, which improves compression efficiency, especially at lower data rates. Most Layer II encoders don't default to having joint stereo activated, so you may need to set this manually. Use joint stereo unless you need to preserve Dolby Pro Logic surround sound encoding.

Exporting to DVD and MPEG-2 formats

Measured in the number of hours of video watched by viewers, MPEG-2 is easily the dominant digital-video format worldwide. It's used in virtually all digital broadcast systems (digital cable, satellite, and broadcast), as well as DVD.

One strength of MPEG-2 is the way it handles progressive and interlaced content. Entire streams can be interlaced or progressive. Interlaced streams can have progressive frames in them, and interlaced frames can have progressive blocks in them. Note that, depending on the player, interlaced MPEG-2 files may or may not be deinterlaced when played back on computers.

The DVD format's MPEG-2 subset doesn't actually support progressive streams. Telecined film on NTSC DVD is stored with the usual 3:2 pulldown pattern. However, by marking the frames that are progressive and those that are interlaced, and by using "field repeat" tags that indicate the 12 fields every second that are duplicates, you can use progressive DVD players to perform a real-time inverse telecine and restore the video to 23.976 fps. When you attach a progressive DVD player to a progressive display, the video can play back at the native frame rate, which improves image quality and smoothness.

Good MPEG-2 encoders automatically take a 24P (or 23.976 fps) source and make a correct "progressive" DVD out of it.

Using MPEG-2 for DVD delivery

DVDs use the MPEG-2 video codec, but they don't use the MPEG-2 file format. Instead, they use a VOB (Video Object) file, which uses other audio codecs and includes other types of information, such as subtitles.

Many compression tools can't directly export a VOB file. To get around this limitation, you can produce an MPEG-2 video file (called an elementary) stream (generally M2V) and a stand-alone audio file (WAV or AC3), and then use an authoring tool to combine them into a VOB file.

Using MPEG-2 for CD-ROM/DVD-ROM delivery

MPEG-2 hasn't become a viable disc-based playback standard because decoders aren't bundled with all personal computers, and you can't legally distribute them without paying a significant licensing fee.

For kiosk applications, MPEG-2 can work well. It has lower processor requirements per pixel than many other formats, so if disc space isn't a concern, you can use MPEG-2 quite nicely to play back high-resolution, high-quality video.

Working with GOPs on DVD

For DVD, the standard group of pictures (GOP) size for PAL and telecined film content is 12, and NTSC uses 15. The standard allows 15 for the maximum PAL and 18 for the maximum NTSC.

GOPs can be either open or closed. In an open GOP, the last B-frames of the GOP can use the next I-frame for reference. The ability to use the next I-frame gives a small, but welcome, improvement in compression efficiency, but it makes random access a little more complex because each GOP is not wholly self-contained.

For DVD authoring, Open GOP is fine except for multiangle titles. Multiangle titles need Closed GOPs and need to have all the I-frames align among the angles to provide smooth switching among angles.

MPEG-2 audio codecs

While there are default MPEG-2 audio codecs, most implementations use other codecs.

Layer II audio

The theoretically standard audio codec used in stand-alone MPEG-2 files is Layer II, the same as MPEG-1. The AAC audio codec that MPEG-4 uses also originated as an MPEG-2 codec, but it is virtually unused in MPEG-2 files. In an MPG or MP2 file used for computer-based playback, Layer II is the only audio codec used.

However, DVDs don't use Layer II; they use other audio codecs that are not part of the official MPEG-2 specification.

PCM

Consumer DVD authoring packages use uncompressed PCM audio. The quality is flawless, but the data rates are high. For a one-hour home video, quality is not an issue, but the Dolby Digital compression efficiency allows you to add more high-quality video to a single disc.

You can use PCM for high sample rates and channels, but it is almost only ever used at 48-KHz 16-bit stereo.

Dolby Digital

Dolby Digital, also called AC-3, is the standard for professional DVD audio. Including AC-3 encoding is the hallmark of a professional DVD authoring tool.

The historical standard AC-3 data rate was 384 Kbps, but the DVD standard allows for 448 Kbps. The DVD standard is best for most multichannel projects unless data rate is at an absolute premium. Stereo encoding can use less—192 Kbps is generally sufficient.

DTS

Some high-end titles also include the Digital Theater System (DTS) audio format. DTS allows for much higher data rates (although these higher rates are partially due to lower compression efficiency—DTS doesn't use psychoacoustics). DTS is generally only used on A-list Hollywood titles, for several reasons. First, the mastering tools are quite expensive and are not bundled with compression tools. Second, DTS decoding isn't mandated as part of the DVD specification, so DTS titles also include Dolby Digital tracks as well. Having both DTS and Dolby Digital tracks cuts into available disc space, and the audio quality gains are slight at best compared to 448 Kbps AC-3.

Exporting to MPEG-4 format

MPEG-4 is the follow-up ISO standard to MPEG-1 and MPEG-2. Unlike those formats, which were created before the popularity of the Internet and were strictly focused on audio and video playback, MPEG-4 is a much broader and extensible format. The format is only now becoming well known.

Profiles and levels

There is no simple thing called "MPEG-4 compatibility." The MPEG-4 standard has a massive scope. To manage the support needs of particular products and industries, it uses the concept of profiles and levels. A profile defines the features you can use, like B-frames. A level defines the maximum parameters that you can use with those features, like maximum frame rate, frame size, or data rate.

The following list contains the profiles that handle video compression that are most commonly seen, or at least discussed, for future implementation. Not listed are the rich media and other, more advanced, profiles that are more a matter of authoring instead of compression.

- **Short Header** This profile is the same as a baseline H.263 and is a subset of the Simple Visual profile. So far, Short Header mode is only required by some cell phones. Even a typical PDA can use more aggressive profiles. Still, Short Header can provide reasonable quality at lower data rates on the small screens these devices have.

- **Simple Visual** This profile is the current baseline for computer playback and is the highest video profile supported by QuickTime. It provides compression efficiency better than H.263, but is less than competitive against the current generation of codecs.

- **Advanced Simple** This profile is an enhanced version of the Simple Visual profile, adding support for interlaced video (at higher levels), B-frames, and other factors that can improve compression efficiency. Most computer players support the Advanced Simple profile, with the exception of QuickTime.

- **Simple Scalable** This profile is the most basic form of scalability for real-time streaming in MPEG-4. Simple scalable has only one enhancement layer, and hence, two possible data rates. It has been adopted in the PacketVideo suite of tools, but not beyond that.

- **Fine Grain Scalability** This profile provides powerful scalability for MPEG-4. It combines multiple enhancement layers plus the ability to use only part of a given layer to provide very flexible scaling that is able to optimize itself very tightly with the available data rate.

  Unfortunately, no products use it yet. Until Fine Grain Scalability or similar Master Boot Record (MBR) technology is implemented, MPEG-4 won't be competitive for real-time streaming over the Internet compared to Windows Media and RealMedia formats.

- **AVC (Advanced Video Coding)** This profile is popular in the MPEG-4 world. Also known as H.264, JVT, H.26L, and MPEG-4 Part 10, it is an entirely new codec that is not based on H.263. Its creators hope AVC will be a codec that lasts a decade, like MPEG-2, so its design emphasizes compression efficiency over performance.

  At the NAB (National Association of Broadcasters) 2003 exhibition, developers first demonstrated the AVC encoders. Currently, AVC encoders are in the early stages and started appearing in products in 2004.

DIVX

The AVI codec receiving the most attention these days is DivX. It doesn't use the .avi extension; instead, it uses .divx. DivX was originally based on an illegal hack of the Microsoft MPEG-4 version 3 codec, but since version 4, it has been based on standards-compliant MPEG-4 for video (Simple in V4, Advanced Simple in V5).

For audio, DivX normally uses MP3 format. To avoid confusion with other types of.avi files, Divx has adopted the .divx extension for files, even though the file format itself is almost a plain AVI.

The reason the format is not considered a plain AVI is that DivX files often contain VBR MP3 audio, where AVI historically didn't support VBR audio. This support has lead to playback glitches on some players, especially QuickTime.

- **Studio** This profile is meant for studio use, like I-frame-only MPEG-2, Motion JPEG, and proprietary capture codecs from many vendors. Using MPEG-4 video technology, the Studio profile offers mastering quality at lower data rates than today's formats. This profile includes 10-bit-per-channel mode, 4:2:2 and 4:4:4, and other features desirable in high-end video authoring.

  There aren't any Studio profile products yet, nor are any expected in the near feature. It remains to be seen if this profile will become a significant use of MPEG-4.

- **N-Bit** This profile is for high-end uses, such as archiving medical images. MPEG-4 offers the N-Bit profiles that allow encoding in 16 bits per channel and other modes. N-Bit profiles are useful in their own domain, but they are unlikely to be used for compressed video delivery or even video authoring.

- **3GPP profile** This profile, which stands for Third Generation Partnership Project, is designed for mobile phones. There are two types, each with different audio codecs. However, most carriers mandate a carrier-specific type of 3GPP. So far, it isn't possible to make a single file that will work on all the MPEG-4-playing mobile phones. As the mobile phones become more powerful, this limitation may change.

- **ISMA profiles** The most important group working on MPEG-4 for web-based computer playback is the Internet Streaming Media Alliance (ISMA). This organization published the ISMA 1.0 specification in late 2001 defining two basic profiles (confusingly, not the same as an MPEG-4 profile, but a combination of MPEG-4 profiles and level).

  - **ISMA Profile 0** This basic profile is designed for very low-power mobile devices, like cell phones. So far, these kinds of devices are either using 3GPP or a carrier-specific format.

  - **ISMA Profile 1** This basic profile was originally designed for computer playback, but modern XScale-based PDAs are also becoming fast enough to support it. Profile 1 is the one most commonly quoted for support. Unfortunately, many players don't support the whole specification. CELP support is only available in a few players.

MPEG-4 licensing issues

So far, the biggest barrier to widespread MPEG-4 adoption has been licensing issues. The license terms aren't prohibitive, but they weren't finalized for a long time, so concern occurred around the ambiguity.

Most formats involve some license fees to build encoders and decoders, and MPEG-4 is no different. Product vendors pay these costs; they aren't an issue for someone performing compression or the distributor. But MPEG-4 includes fees for distributing content in the MPEG-4 format. For disc-based content, replicators have been collecting MPEG-2 fees for making DVDs for years. Fees for distributing content by means of the Internet is new for MPEG-4.

These fees aren't as intimidating as they sound—most distributors won't have to pay anything. First, the fees only apply to remunerated content, not marketing, advertising, and noncommercial content. For remunerated sites (mainly those doing subscription or advertising-funded content), the first 50,000 subscribers per year per legal entity are free. After 50,000, the distributor can pay his or her choice of $0.25 per subscriber per year or $0.02/hour of content streamed. These numbers are a small portion of the budget of any streaming project. Still, the simple issue of tracking use has limited adoption in some cases.

MPEG-4 audio codecs

Audio profiles in MPEG-4 combine sets of possible codecs. Because most implementations support particular codecs instead of whole profiles, it's generally more useful to describe the audio codecs instead of profiles.

AAC-LC

The audio codec used in most MPEG-4 implementations so far is Advanced Audio Coding-Low Complexity (AAC-LC). The "Low Complexity" aspect is a very high-quality codec. While there are other types of AAC, only one has shown significantly better compression efficiency than AAC-LC in listening tests.

The ISMA standards include AAC-LC, and every computer-based MPEG-4 player supports it.

HE-AAC

The new variant of AAC is HE-AAC (the HE stands for High Efficiency). HE-AAC uses a technology called spectral band replication, which is essentially a very efficient enhancement layer.

Currently, HE-AAC is the best audio codec by a fair margin. Listening tests demonstrate that, at 48 Kbps, this codec is perceived as having approximately the same quality as CD audio; at 32 Kbps, it has entertainment quality, similar to that of FM radio. HE-AAC has yet to become mainstream, although its widespread adoption is expected by industry experts.

CELP

CELP is a low-bit-rate speech codec. Due to licensing issues, it hasn't been deployed in enough players to reach critical mass, so it isn't widely used. Like other similar codecs, it provides intelligible speech at low data rates. There are several other speech codecs in MPEG-4, but CELP is the most frequently used codec. A speech codec is not needed as much in MPEG-4 because AAC-LC performs well with low-bit-rate speech.

AMR

The only other speech codec that matters in MPEG-4 is Adaptive MultiRate (AMR). Like CELP, it's a low-bit-rate speech codec. Unlike CELP, AMR supports real-time scalability. AMR is mainly used in the 3GPP profiles for cell phones.

Exporting to AVI format

The Microsoft Audio Video Interleave format was a very early format for CD-ROM, and it is still widely used in a variety of applications.

Using AVI for web delivery

AVI is not a good web format because it doesn't support real-time streaming. Also, the set of codecs that are available by default on all computers is relatively limited. AVI files on the web are almost always distributed as complete files.

Using AVI for disc-based delivery

AVI is still a popular format for disc-based playback. It's well supported in applications, such as Adobe® Director, and because codecs can be delivered with a disc, installations are much less of a worry.

AVI video codecs

There are many codecs available for AVI. Most of them require manual installation. The following list contains a few of the most common and useful codecs.

Cinepak

The classic AVI codec was Cinepak. As previously described in the QuickTime section, Cinepak offers limited quality but broad playback compatibility and very fast decoder performance. Cinepak is the only video codec that you can assume is available to all AVI players.

Indeo

Intel created the Indeo codecs, but Ligos now owns them. Indeo 3, 4, and 5 were very popular during the height of the popularity of CD-ROMs. However, they haven't been meaningfully updated in years. They used to be good default codecs, but Indeo playback isn't preinstalled in Windows XP, nor is it available for Mac OS X. The Indeo 3, 4, and 5 quality was poorer than modern codecs, especially because they used the YUV-9 color space, which only had one chroma sample per 4 x 4 block.

VP3

As previously described in the QuickTime section, VP3 is an open source codec originally created by On2. Free encoders and decoders are available for it in a variety of platforms, although it is not preinstalled on Windows or Mac OS. (AVI playback of VP3 files works by means of QuickTime, which includes VP3 in its automatic component download program.)

The current version of the VP3 codec has a few features not available in the QuickTime version. The Lowest Allowed Quality control sets a minimum threshold for quality. To maintain the data rate, you should use the minimum value of 63. The Sharpness control sets a target for whether a frame should have high detail but contain potential artifacts, be soft but contain no artifacts, or be in the middle. The best choice is specific to the source and how it's used.

VP5/VP6

These new codecs are also from On2 and are built on VP3 with much improved compression efficiency, especially in VP6. VP6 is available as a free download for personal use from *www.on2.com*.

DivX

DivX codecs had an ignominious beginning but have evolved into an important part of the AVI infrastructure. Today's DivX 5 is a good implementation of the MPEG-4 video codec, supporting either a Simple or Advanced Simple profile.

The DivX codec also contains a number of preprocessing features inside the codec itself that can be useful if you are exporting with a tool that lacks those features.

One unique feature for DivX in AVI codecs is that it supports N-pass encoding, during which it successively refines the compression in multiple passes. DivX performs N-pass encoding in a roundabout fashion by building a dummy AVI file with the complexity data stored in it for the first pass, and then it uses that dummy file for the second pass. Some tools automate this process, others require you to perform it manually. It's rare when more than three passes continue to improve the compression; even the difference between two and three passes is quite slight.

DivX has a specific psychovisual mode in which an internal model selects the parts of the image that are of greater interest to the human visual system, and then the codec attempts to distribute bits to those parts. Using the Light setting is quite safe; however, the Extreme setting can produce odd results. Still, at very low bit rates, this mode can provide a meaningful improvement with some content.

DivX lets you manually set the threshold for triggering a new natural keyframe. The default of 50 is a good starting point for most content. DivX also allows five levels of trade-off between quality and speed of encoding. Time permitting, always use the slowest mode.

AVI audio codecs

Audio codecs in AVI don't receive the same attention as the video codecs, but audio codecs are just as important, and there is a vast variety of them. The following list contains a few of the most important ones:

None

The most basic audio codec is None, an uncompressed mode like the QuickTime None codec. It is mainly useful for intermediates, not delivery.

IMA ADPCM

The IMA codec is the same as the one in QuickTime, offering a 4:1 compression over the None codec with decent quality. It's the most universal audio codec for AVI and has a very light decoder.

MS ADPCM

The MS ADPCM is a slight variant of IMA. It doesn't offer any advantages over IMA, and it's slightly less compatible, so you don't need to use it for authoring.

MP3

The MP3 codec is popular with AVI. However, VBR MP3 audio isn't nearly as widely supported as CBR MP3. Windows computers include a very limited MP3 encoder for AVI that maximizes at 56 Kbps. However, a number of good third-party encoders are available.

FOR MORE INFORMATION

For a comprehensive overview of Adobe products, please visit www.adobe.com.

Adobe Systems Incorporated • 345 Park Avenue, San Jose, CA 95110-2704 USA • *www.adobe.com*

Adobe