

About HDV

High Definition is Not Just For the Future. It's Here Right Now.

The new HDV format is poised to revolutionize high definition production in the same way DV revolutionized standard definition. But innovations also raise questions. This Web site aims to answer those questions.

What is HDV?

It is a common misconception that HDV and HD formats are the same. Simply put, HDV is a video format that uses the HD line resolution (1080i or 720p) in a highly compressed format: MPEG-2 Transport Stream. This creates a stream that is small enough (roughly 25 Mbps @1080i, 19Mbps@720p) to fit on a standard DV tape. In addition to the data compression of the MPEG-2 format, HDV does not store all of the data that full-resolution HD video has. For example, one form of uncompressed HD is a 1920 x 1080 interlaced frame, but the similar HDV spec stores a 1440 x 1080 interlaced video frame. This combination of MPEG compression with a reduced frame size keeps HDV more manageable while also keeping the quality of the video very high. This technology also makes HD video resolution much more affordable than ever before.

A common analogy used to compare HDV and HD is to compare uncompressed Standard Definition (SD) to an SD MPEG format such as that used for DVDs. An uncompressed SD file is very large file compared to the highly compressed MPEG format, but the can both be the same resolution (720 x 480 is a common example). This is a very similar relationship full HD video and HDV video has with each other.

Background on Digital Images & Imaging

When the DV format first appeared, about eight years ago, it was a massive improvement over VHS. It was a digital video format, which meant an end to tape noise, and the images were so sharp and clear that it was easy to mistake them for those produced on hitherto more expensive 'professional' video cameras. Also in the last eight years, we've seen the birth, and blossoming into maturity, of digital still photography.

In some ways, digital still photography is technically less challenging than video, in that you only have to capture a single 'frame' at a time. With video, it's a bit like having to capture thirty still photographs per second, for maybe an hour. In this respect, video is very much harder than still photography. But in another way, digital still photography is much, much more difficult. And that's because a video frame has a fixed size (in the US it's 720 by 480 pixels), and in still photography the aim is to capture as many pixels as possible.

In the early days of digital still photography, the number of pixels wasn't an issue. In fact, Sony's first digital still camera, the MAVICA, was actually a single-frame digital video camcorder, which made complete sense at the time because the idea with the MAVICA was that you would display your pictures on a television. There was very little wrong with this idea, as long as digital still photographers accepted that television quality is a very, very long way short of film resolution.

Fast forward to 2004, and you can buy cameras from any high-street consumer electronics shop that can rival and even exceed the quality of film.

The resolution of digital still cameras nowadays is simply astonishing, as is the technical skill needed to produce CCDs with as many as sixty four megapixels, which you'll find in some very high-end cameras. For under a thousand dollars, you can buy digital still cameras with resolutions of around eight megapixels - arguably as good as any non-professional would ever need.

So how does DV's resolution compare with this? Well the short answer is, very badly indeed. We can do the calculations quite easily:

A DV frame is 720 by 480 pixels. Multiply them together and you get 345,600 pixels. That's less than half a megapixel! Half a megapixel is a much lower resolution than even the cheapest digital still camera you can buy today. So how can it be that DV originated video, which looks so good in comparison with VHS, can be based on such a low resolution?

There are several answers to this, all of which help to explain the important differences between HDV and DV.

When we see an image, our impression of how clear it is depends, in very broad terms, on the amount of information the image presents to us. With still images, the easiest way to describe the information content of an image (short of actually describing what's in the picture) is to talk about the number of pixels. That's where DV loses out in comparison with digital still cameras.

But video has a huge advantage compared to still photography, because with a moving picture the amount of information presented to the viewer is given by the resolution of a single frame multiplied by the number of frames per second! Another way to explain this is to say that video has both spatial resolution and temporal resolution (resolution over time, in other words). Video updates what you're seeing thirty times a second and our impression of the quality of video material is hugely enhanced by this. If it weren't for this effect, VHS quality would be simply unacceptable.

There's another factor as well, which is that we simply don't expect better quality from video. We're used to watching televisions and we're comfortable with the quality they can give. Most people have never even seen how good standard definition television can get. Pictures direct from a studio camera shown on an expensive broadcast monitor look absolutely stunning, and do so despite being constrained to a resolution of less than half a megapixel. So, standard definition television can look great, and will still be around for a very long time. But there are also very good reasons why we now need something better than SD. Here's the main one: big screens.

Big screens present a challenge to standard definition video. The closer you sit to a big screen, the worse it looks, and it's easy to understand why. Quite simply, with a large screen, you need more pixels, not just bigger ones. And that's the

problem with standard definition. It doesn't matter how big the screen is, it's still going to be showing the same old 720 by 480 pixels. Bigger screens don't mean more detail, but they might mean pixels the size of dustbin lids.

Despite the inadequacy of our current TV standards, big screens are flying out of the shops. If you ask big screen users what they like about their displays, they'll tell you how flat they are, how little space they take up, and how impressively big they are - but only if they sit a very long way from them will they talk about how good the pictures look.

So big screens are a problem for standard definition TV, although it's not only size that matters. Now that people are used to looking at digital photographs on computer screens, they're starting to ask "Why can't my video look as good as that?" And it's a very good question. Why shouldn't video be available in higher resolutions?

Well the good news is that it is.

Data & Compression

All About Data

What's good about HD is that, with five times SD's number of pixels, the pictures are fantastic. What's not so good is that the amount of data you need to store & move around goes up by a factor of five as well. This, to put it mildly, is an issue.

To put it in perspective, standard definition video generates around fifteen floppy disks worth of data per second! In plain text, you can store *War and Peace* on a single floppy. Multiply the fifteen floppy disks you need for a second of SD by five and you reach the staggering conclusion that HD generates the equivalent of seventy five *War and Peace*'s per second. If you've ever read *War and Peace*, you'll know that's an awful lot of data.

Incredibly, computers can actually deal with uncompressed HD, which generates an almost unimaginable one Gigabit per second, but you need to have a computer that is fit for the job, with huge quantities of extremely fast storage. You'll know if your storage is extremely fast, because it will have been extremely expensive. But for the rest of us, who don't want to spend a fortune on enough storage to house the entire US Library of Congress every six minutes, there has to be a better way. There is, and it's called compression.

As you might expect, compression is a complex subject. The programs that perform compression - called **codecs** - are mostly designed by mathematicians. Luckily, these programs work so well that it's perfectly OK for most of us to remain blissfully ignorant of the way they work. If all you ever want to do is simple video editing, using pre-configured settings, then you can safely skip the next bit. But if you're ever going to go beyond that and make videos for distribution using a variety of media (DVDs, the web, etc), then it's worth knowing a bit more.

As we already know, the most popular video format used by camcorders today is DV, along with its close relatives (Panasonic) DVC Pro and (Sony) DVCam. We often refer to these three formats as DV25; where the "25" is the number of megabits per second. DV compresses standard definition video by a very useful factor of five. It brings the data rate down to a point where you can get an hour or so of DV on a videotape, and several hours on the average desktop computer's hard disk. Over the last five years or so, computers have speeded up and hard disks have got bigger, to the extent that you can work quite happily with DV on almost any modern computer.

But HD produces five times as much data as SD. If we were to only compress HD by the same amount as we squeeze SD, then we'd only manage to get about twelve minutes of video onto a DV tape, and we'd only be able to do that if we could move the tape at five times the speed. With a rotating head recording system this is not a trivial engineering task. So, HD has to be compressed a lot more than SD if we're going to be able to work with it. And this presents us with a dilemma: compression reduces quality, and yet the very reason we're using HD in the first place is to improve quality.

Until HDV appeared on the scene, there was no way out of this particular conundrum. And it solves it in a very clever way - by using time.

Squeezing The Most Out of Compression

First, understand that we're not talking about the type of compression used to "zip" files on a PC. This type of compression can reduce data files by analyzing the statistics of character use, and assigning "tokens" to the more frequently used ones. The more common the character, the shorter the token used to describe it, and vice versa. This, together with a few other techniques, means that when you "unzip" the file, you'll get a perfect copy of the original. This is called "lossless" compression and it works very well.

But it doesn't work so well with audio and video because, to a lossless compressor, digital audio and video look like random data. So there are no patterns to recognize, and so they can't be compressed (there are some lossless compression codecs that work with audio and video, but while they give very good quality, they don't work at the high compression ratios that are essential for fitting HD onto a small-format videotape).

Video compression works differently. With such huge compression ratios needed, there's simply no way to reconstruct the original data file. There's no need, either, because if the result looks the same (even if the data file is different) then, to all intents and purposes, it is the same. As we've already mentioned, video compression is a complicated business. But it's very easy to understand the basics. It's a good idea to learn a little bit about how this stuff works because compression does affect the way your video will look. Knowing where the problems are will help you work round them, or avoid them in the first place.

Video compression normally works by looking at the content of a frame, analyzing it, and looking for ways to describe it that don't involve giving a value for every individual pixel. There are several ways to do this. In a simple case like this, all the compressor has to do is say "every pixel in this frame is the same shade of white". That's a lot less data than writing "256, 256, 256" four hundred and fourteen thousand, seven hundred and twenty times. Another way that video compression works is to look how sharp the borders between light and dark shades are and find ways to describe them more efficiently. It does this by dividing the scene into blocks of pixels, called macroblocks, and representing them with numbers that can recreate the patterns within them (all so-called Discrete Cosine compressors, including DV and MPEG work like this).

Despite the complexity of this process, it's a well established technology and works very well. But it doesn't give a good enough compression ratio for high definition. This is where time travel comes in handy. We've already seen that video compression works by looking for easily describable features within a video frame. If these features are repeated, then it's only necessary to describe them once. And exactly the same applies to nearby frames, as well as within the frames themselves.

Again, imagine our white wall. There's nothing at all in the frame, and nothing changes over time, either. So all the compressor has to do is count the number of frames in the shot, and say "all these frames are the same". If every frame is the same, you only need to record the details once.

Things get a bit more complicated when there's movement in the video. If there is movement in only part of the frame, then only the moving parts need to be updated as time passes. The pixels describing the motionless parts still only need to be sent once. And even where there's movement, it's still possible to reduce the data by "tracking" the path of the moving objects. Suppose there's a car driving from right to left in the frame, while the camera viewpoint is fixed. The block of pixels that describe the car effectively doesn't change at all, but their position in the frame does. So all the compressor has to do is figure out where the motion of the car begins and ends, and move the same block of data along that path.

HDV uses MPEG-2 compression. It's exactly the same type of compression that DVDs use – so it's well tried and tested. The only difference is that the pixel count is scaled up to cope with HD resolutions. MPEG-2 is very good at using similarities between frames, so it divides video into bunches of frames called a Group Of Pictures or **GOP**. A GOP contains several different types of compressed frame. There's no need to go into too much detail here, but these are the basic types:

- I frames are compressed frames that do not depend on any frames around them.
- P and B frames are predicted from the content of adjacent frames. You can't decompress an isolated P or B frame because of their dependency on other frames.

There is a version of MPEG-2 used by broadcasters that doesn't use GOPs, it only has I frames. It doesn't compress the video as much as long GOP formats. DV compression is like this too, with good reason. When you're editing video, you need to have equal access to every single frame. Editors want precise control over their footage so they can make cuts in exactly the right place. If you were only able to make cuts every five or ten frames, it would be difficult, not to say impossible, to edit – especially where dialogue is involved.

Working with HDV

We've already seen that compressing HD tightly enough to fit it onto a DV tape presents us with a fundamental difficulty: greater compression leads to lower quality. And we now know that the compromise that makes it all work is **MPEG-2 Long-GOP compression** (a bit of a misnomer because Short GOP means I frame only, which isn't a group of pictures at all).

The truth is that Long-GOP compression wasn't designed for editing video. It was devised as a way of delivering video to end users. MPEG-2 Long-GOP is how digital TV gets to most digital TV Viewers in the world. It's used for satellite TV, cable TV, digital terrestrial TV and DVD. It works extremely well. Most people think that DVD video is the best they've ever seen. So Long-GOP can deliver outstanding pictures.

Long-GOP is good for delivery because it offers very high compression, and good quality, and because end users typically don't edit incoming television programs. But with HDV you have to edit a long-GOP format. We're going to look at how this works, and how to get the best from this clever compromise.

First of all, let's dispose of the idea that the non-I frames in HDV (i.e. the P and B frames) aren't actually there. Even though they are completely derived from the frames around them, P and B frames do actually deliver a picture. They have to, or none of this would work at all! When HDV is decompressed, all the frames are there on your screen. When it's all working properly, you can't see any difference between I, B and P frames.

There are several schools of thought about the best way to edit HDV. Canopus gives you all the options so that you can choose whichever is best for you. But remember, the quality of your finished video is only as good as the weakest link in the chain, which is why Canopus has concentrated on these potential problem points and has given you the best possible solution.

Compression Using Codecs

You can't modify compressed video without decompressing it. You can't view it in its compressed form. In fact, no-one has ever 'seen' MPEG-2 or any other type of compressed video, because you can't 'see' the digits that make up the staggeringly complex body of mathematical data that MPEG-2 is in reality. All you can ever see is the decompressed results of the compressed video.

So, there is no such thing as editing 'native' compressed video, except in the case of working with a short-GOP format like DV, and even then only if you're making cuts only, with no special effects. Not even a dissolve. Strictly speaking, the only way to work natively with HDV would be to make cuts on I-frames only – which will be several frames apart. Making a cut in the wrong place could cut off some B and P frames from their 'parent' I frames, and the video would disappear until the start of the next GOP. You can sometimes see this effect on digital television when a glitch in the transmission either freezes the picture or breaks it up into a multi-colored chessboard, until the next complete GOP arrives.

Editing natively has come to mean something slightly different from this. A 'native' HDV editor stores video in the HDV format, decompresses it when it needs to be seen or processed, and recompresses the footage to HDV to give the final result. This works quite well, but isn't necessarily the best way to work with HDV. Here's why.

Video compression is not lossless - it's lossy. This isn't quite as bad as it sounds. Whilst it's true that when you compress video by twenty to one you are throwing away ninety-five percent of the data, what you're actually disposing of is data

that you probably wouldn't miss anyway. So even after throwing away ninety-five percent of the picture, it still looks nearly a hundred percent as good. This is the miracle of modern compression technology. But, unfortunately, we can't rely on miracles. When you recompress video that's already been compressed, you're starting with less than the full picture, which means that whatever result you get will be based on an approximation based on guesswork based on.... Well, if you do it too often, you'll end up with something that's unwatchable.

There are no hard and fast rules about all this. It depends a lot on the content of the video. Simple, plain colors with little movement will recompress better than lots of detail (think of a side-of-the-freeway mural) and lots of movement (think of a side-of-the-freeway mural filmed from the back of a motorcycle). It's fair to say, though, that you need to avoid too many compressions and re-compressions of HDV. Complex multi-layered work will lead to problems.

There's another reason why you might not want to work natively in HDV. It takes a lot of processing power to compress and decompress HDV footage. This is effort that could be more usefully spent on creative effects and playing the multiple video streams that you need to do complex effects in real time. The burden of working with native HDV is very significant indeed. It can slow down the whole process, and severely limit the extent of what is possible in real time. Canopus has the answer to all these issues.

The Canopus Advantage

First of all, Canopus has hardware solutions. Even though modern PCs have incredibly powerful processors, they still have finite capabilities. Working with DV is comfortably within their abilities. Working with HDV is not.

When working 'natively' in HDV, without dedicated HDV hardware, editing can be a slow and unrewarding business. There are questions about quality too. Canopus hardware solutions are able to filter the video material as it is re-sized in real-time. Other, software-only, editing solutions either take much more CPU power, which can make the process non-real-time, or take resources from other components, for example a graphics card. In the latter case it's very likely that the quality will suffer, as graphics cards are optimized for RGB graphics (unsurprisingly), not YUV video.

Secondly, Canopus has the best codec technology for editors available anywhere, and this is the key to the quality and performance of Canopus systems. We've already seen how editing natively in HDV is slow and can lead to poor quality. The Canopus solution to this is elegant and effective. Canopus has its own high definition codec. It uses more gentle compression than HDV, and is very much more suitable for editing. It needs less processing power as well, so you can play more simultaneous video streams and have more real-time effects.

Here's how it works.

When HDV is brought into a Canopus editing system, it is transcoded into Canopus's HQ format. You work in HQ for the duration of your project. What this means in practice is that you will be able to play more video streams in real time (because it's easier to uncompress and compress in HQ), and you'll be able to work with more layers without significant quality loss.

With Canopus's HQ codec, and with hardware resolution conversion, editing is as easy and straightforward as working with DV. You use the same tools, the same software, and the same workflow.

But how do you get to see the results of your work, given that most people don't have high definition televisions in their homes?

Delivering HDV

There's good news here too, on several fronts; starting with people who've only got SD televisions and DVD players.

When you shoot a video in HDV, the process starts with the light-sensitive element in the camera. Sony's HDV camcorders actually have three of them and, together, they gather four or five times the amount of visual data that a conventional DV camera would. Which means that if you down-convert the image to SD, it will probably look better than if it started life in SD. There may be exceptions to this: if your down-conversion technology isn't very good then the final image will suffer. Canopus's down-conversion tools give you the very best quality, so this isn't a worry. It's also fair to say that if the original footage was shot using a very high-end SD camera (especially one with an expensive professional lens) then down-converted HDV might not look quite as good, although it will look different.

But if you compare the output from a high-end consumer DV camcorder with that from an HDV camcorder, down-converted using Canopus technology, then the HDV-sourced footage will almost certainly look better. This means that everyone, whether they've got HD display equipment or not, can benefit from HDV acquisition (i.e. shooting) and editing.

To see HDV in its full resolution, there are several easy options as well – starting with flat-screen televisions. Plasma and LCD screens often come with a resolution that is higher than standard NTSC. Some don't, and this is something to watch out for carefully if you're thinking of buying one. If you see a plasma screen that's significantly cheaper than everything else, it probably won't be suitable.

To appreciate HDV, you will need a resolution of at least 1280 by 1080. Lower resolutions like 1080 by 768 will show an improvement over NTSC but it's better to go for more pixels if possible. The ideal resolution is 1920 by 1080, but although these screens do exist, they are expensive. There's no real reason why HDV video shouldn't be distributed as HDV. It uses a compression format (MPEG-2, long-GOP) that has all the characteristics of a distributable format. But the files will still be quite big - they're the same size as DV files. With DV, or any SD format, for that matter, it's normal to convert them to Standard Definition MPEG-2 long-GOP, which is the format used for DVDs.

You might think it would be asking too much to compress HDV even more, so that you could fit reasonable amounts of it onto a DVD sized disk, but, remarkably, that's exactly what is possible. There are compression formats available that are even more efficient than MPEG-2. Some of them are part of the MPEG-4 specification. But the most easily available, and easiest to distribute, is Windows Media Video. Using WMV, it's quite possible to fit a whole HD feature film on a DVD-R. Incredibly, data rates as low as five megabits/second can give good results. Of course this depends on content,

and if there's a lot of movement then you'll have to use a higher data rate. Canopus gives you all the tools you need to create WMV files that anyone running Windows will be able to view on their computer (as long as it's fast enough).

For the future: it seems likely that WMV will be a 'required' format for high definition DVDs, which means that every DVD player capable of playing High Definition DVDs will be able to play the WMV files that you create. Unfortunately, until the new generation of DVD players arrives, then you're restricted to playing Windows Media 9 files on a computer.

Until then, there's an alternative, which is probably never going to be a worldwide standard, but is so cheap to buy that you might want to think about it for the short term. It's called HVD, which stands for "High clarity Video Disk". Luckily the pictures it produces have a much better quality than the name of the format. HVD is a Chinese technology that comes in the form of a very low cost player capable of playing back high-definition material recorded onto a DVD-R disk. The compression used is MPEG-2, just like HDV, and the results are surprisingly good for such a low-cost system. Nobody expects the HVD format to be around for decades – but it's so cheap and effective that it's worth thinking about, if you want to get your HDV movies seen, now.

Video Connections with HDV

There are five types of connection that can work with HD: Component, VGA, DVI, HDMI and HDSDI. The normal video connections like composite and Y/C don't work as they're tied-in to NTSC and PAL, which are SD standards.

Component is the connection you're most likely to find at the moment. It is analog, so there is some quality loss between the source and the screen, but with decent (which normally means expensive) cables, it is capable of very good results. What's especially good about it is that most equipment that's even remotely capable of showing HD has a component connection. Component connections don't carry sound, so you'll need a separate arrangement for audio.

VGA is a computer standard for connecting computers to analog display screens. It does work well and in some circumstances is a good, and possibly the only, way to get HD video from a computer to the screen. (we'll be looking at how to prepare HDV for viewing on computers in the next section). VGA doesn't carry sound.

DVI is a digital connection. This is good for video, because there is no quality loss at all. This is absolutely the best way to show video from a computer. Some DVD players have a DVI connection, but this is likely to be superseded by HDMI (see below). DVI has a maximum horizontal resolution of 1,600 pixels, which means that in theory it can't handle the full HD specification – but in practice, this is only going to be an issue with very high resolution screens. DVI also doesn't carry sound.

HDMI stands for **High Definition Multimedia Interface**. It is the ideal way to connect HD sources to displays because it has the capability to handle uncompressed digital HD, as well as several channels of sound. It's a fairly new standard but it will be around for many years and it makes sense to pay a little extra for equipment with HDMI, as it will become the standard home HD interface. HDSDI is the HD version of SDI, which is the professional standard for moving uncompressed SD video around a studio. Since it is digital, there is no loss of quality. It's unusual to find displays with HDSDI outside of professional video environments, and where it does exist it carries a professional price tag. HDMI is just as good and will invariably be cheaper. If you are working with professional HD equipment, then you will probably need an editing system equipped with HDSDI. HDSDI can carry embedded digital sound.

Progressive vs. Interlaced

Interlaced video has been around as long as there have been televisions with Cathode Ray Tubes (CRTs). All analog television standards are based on interlaced video, and we're so used to it that it's not normally something that we think about. High Definition video can be either interlaced or progressively scanned. It's important to understand these terms, and it's really not too difficult.

Progressive scan video is scanned from right to left, top to bottom: line 1, line 2, line 3 etc, up to the end of the frame. It's as simple as that. In fact, it's exactly how you'd expect video to be scanned if you didn't know any better! Interlaced video is scanned from right to left, top to bottom, in the same way as progressive scan video. The difference, though, is that every sixtieth of a second, every other line making up the complete frame is scanned. Then, a sixtieth of a second later, the lines in-between the lines already scanned are captured. Effectively, half the picture's vertical resolution is sent in the first sixtieth of a second, and the second half is sent in the second sixtieth of a second. When the video is played back, the whole thing happens in reverse, giving the appearance of a complete frame. Each of these "halves" of a frame is called a "field".

The effect to the viewer is quite distinct. First, the image doesn't flicker as much as it would if it was a simple 30 FPS progressive scan. This is because, to the viewer, it looks like they are seeing sixty frames per second. Of course, what they are actually seeing is sixty fields per second; but for flicker reduction, the effect is the same as seeing sixty frames.

If you were to look at an interlaced picture on a screen for a sixtieth of a second, you'd only see half the vertical resolution. But, because our eyes have a "persistence" effect, when you look at the screen "normally" what you actually see is something approaching the full resolution, because we're able to accumulate visual data from the two distinct fields, making them seem like one, complete frame.

Progressively scanned high definition video tends to have a resolution of 1280 by 720 pixels, normally referred to as "720p", where the "p" stands for progressive. Likewise, 1080i is interlaced video with a frame size of 1920 by 1080 pixels. Interlacing is actually a form of compression. Offsetting the two fields making up a frame by half a frame's duration, halves the total amount of information needed to transmit or store the video. Uncompressed interlaced high definition video generates around a Gigabit per second. Without interlacing, the rate would be twice that.

There is a special case of progressive scanning, which is known as 24p. Twenty-four frames per second sounds very slow, but it's used by some video systems to mimic the frame rate of film. HDV doesn't support 24p directly, but some

cameras can mimic it using a technique called "pulldown" where fields from adjacent frames are combined and repeated to create the effect of the lower frame rate.

Interlaced video does have some disadvantages in comparison with progressive. It's prone to causing "artifacts", which look like a kind of "comb" effect, especially in slow motion or in still grabs from moving video. They are caused by the relative movement between two fields that make up a single frame. It's a fact of life, though, that 1080p, (1920 by 1080 progressive), which might appear to be the perfect format for high definition video, would generate too much data for current consumer technology, so we might have to wait a while for that. Rest assured that your Canopus technology, which is completely resolution independent, will be able to handle it if it ever does appear.

HDV in Practice

Imagine you're a wedding videographer. You've decided to go with HDV and have a 1080i HDV camcorder and a Canopus HDV editing system. You shoot your video and edit just as if it were standard definition DV. The only difference for you is that, when you were shooting, you probably spent more time looking at details, because HDV shows up everything. Now you have to figure out how to prepare the video for distribution to the wedding guests.

The first decision is easy. You've got to make a DVD. That's because standard definition video isn't going to disappear overnight, and it's what most people will be expecting, anyway. Don't think that just because you're delivering a standard definition DVD, you're throwing away all the benefits of shooting in HDV. The chances are that your DVD will look better than the ones you used to make from footage sourced in DV. That's because Canopus can convert directly from HDV to DVD format, and because the additional visual information in HDV footage actually helps the DVD type MPEG compressor make a better picture. So, standard definition customers will benefit from your choice to use HDV.

You'll also want to produce high definition versions. Don't forget that people often look at wedding videos five, ten or more years after they were made. So you need to make a version that's going to be viewable for the foreseeable future. That's difficult because we don't know what's going to happen in the future.

So what you can do for the time being is:

1. make a Windows Media file for viewing on computers and – probably – future high definition video players, and
2. store the material as HDV video on tape – or store the files themselves on removable storage media.

Remember that Canopus will be able to deal with virtually any new HD distribution formats – so you're always going to be able to convert your material at a later date.

Now let's suppose that you're a corporate videomaker. Again, for the time being, you're probably going to have to produce DVDs from your HDV material. Your clients will appreciate the different "look" to your material. They'll also like the way your graphics and charts look clearer because all graphics processing is done by Canopus in high definition resolution – even if you're only working with standard definition. And if you have to incorporate archive footage, or use material in standard definition, Canopus will let you mix multiple formats on the timeline, in real-time; upscaling video to HD if necessary.

If your work is to be shown in the company's atrium, they've probably got an MPEG-2 HD server, which will want an MPEG-2 transport stream. This is something you can easily output with Canopus. They might even want a version for their web site. You've already got the tools to do this, and the quality will still look better than the DV-sourced equivalent.

Finally...

You don't have to upgrade to HDV. Standard Definition is going to be with us for a long time yet. But HDV editing systems are here, now, and so are the cameras. So, when you upgrade your editing system, buy one that can do HDV. A Canopus editing system will work superbly with your old footage, and it'll be ready for you when you want to make High Definition masterpieces, that will still look fantastic in ten years and beyond.