

# How it works: MP3 compression

You probably listen to MP3 files every day – it's become the universal format for music – but how were its standards established and why has it become so popular?

Ask us to name a universally known file format, and it would probably be a toss-up between MP3 and JPG. Simply put, if you're a fully paid up member of the digital multimedia revolution, you will have thousands of these files on your computer's hard drive – music you listen to and photos you look at – both of which have been compressed to cram as much information as possible into the minimum of space.

## What is an MP3?

The MP3 is a fairly recent invention in digital music and sound. Before that, there was the compact disc, or CD. The audio on a CD is converted from an analog source, such as the master tape (although these days, most audio is recorded directly as digital). The analogue wave can't be recorded digitally as it is, so a digital audio processor is used to sample the analogue audio wave 44,100 times a second.

This means, at every tick, the digital audio processor works out the amplitude of the original very complex audio wave. It records this as a two-byte value, so there are 65,536 possible values for this amplitude: 32,767 values above zero and 32,767 below. It does this sampling for the two channels of stereo as well. For a CD, the values of the amplitudes are

stored directly onto the CD as a series of pits that the laser in your optical drive can read and interpret. No compression is done on the data stream. Since a CD can store up to 74 minutes of music, we can calculate the storage capacity of a CD: 74 minutes  $\times$  60 seconds per minute  $\times$  44,100 samples per second  $\times$  two bytes per sample  $\times$  two channels = 783,216,000 bytes or 747MB. Furthermore, the I/O channel that the

CD uses needs to be able to transfer 176kB of data per second to the digital audio processor (the one that reconstructs the analogue audio wave from the digital data and then feeds it through the amplifier to the speakers).

## Recording soundwaves

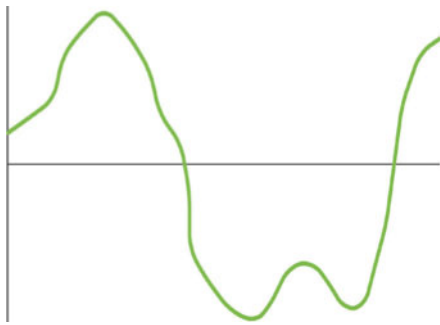
The reason for the sample rate and the amplitude measurement is fairly mundane. Suppose we're sampling the waveform shown in Figure 1. If we sample at too low a rate, we may miss some peaks and troughs in the original audio and so the resulting waveform may sound completely different and muddy. Figure 2 on p76 shows this scenario, where the resulting waveform in red looks quite different from the original. We therefore need to sample much more often. Given that the human ear (in general) only hears a tone up to about 20kHz in frequency, we should therefore sample at least twice that rate in order to properly capture the highs and lows of the audio wave at that frequency. With a fudge factor added just in case, the rate settled on was 44,100Hz.

Figure 3 on p77 shows a different problem: the number of possible values for the amplitude is fairly small. From the original measured amplitude, the processor must choose the closest value it can record. Here we've got a

## 44,100 hertz

When the CD standard was first designed and discussed, it was necessary to establish a standard sampling frequency. It was well known from the Nyquist-Shannon sampling theorem that, in order to sample a wave of frequency  $N$  hertz, you have to take  $2N$  samples per second. Because the human ear can only hear up to 20kHz, that would imply 40,000 samples per second. Yet the rate that was settled on was 44.1kHz. Why?

It turns out that in those early days, digital audio couldn't be recorded onto hard disk efficiently or economically. Instead, digital audio was recorded onto professional quality video tape. So if you take the number of two-byte samples you can get on a line, multiply it by the number of lines in a field, and then multiply that by the number of fields per second, the result will be a rate of 44,100 samples per second. ■



▲ Figure 1: Waveform to be sampled.

► fairly high sample rate, but the measurements of the amplitude are pretty coarse. Again, the resulting waveform looks different from the original – a little more subtle perhaps, but it could still alter the sound pretty badly (highs might be higher than the original, for example, making the result more shrill and meaning that subtle nuances in the music are lost).

Here, a different criterion comes into play: making the sample values fit into a whole number of bytes to help make the output DAC's job easier. One byte would be far too small for this (with only 256 different values for the amplitude), so the original designers decided on two bytes per sample.

### Compressing the data

There things stood until the age of the personal computer and the internet. A three-minute track on a CD (which is the length of a typical pop song) occupies 31,752,000 bytes, or just over 30MB. Downloading a CD track using a 9,600 baud modem would take hours, and would still take well over an hour on a 56K dialup modem (the fastest retail modem before broadband became mainstream). On a typical broadband connection (12mbit/s download, for example), that track would take under three minutes to download, meaning you could just about stream it while listening to it.



▲ Figure 2: Sampling the waveform with a slow sample rate and high accuracy.

## Spotlight on... Patents and copyright

Just because you can rip an album from a CD into MP3 files doesn't mean that you're allowed to. Once you have those files, copying them is easy and deploying them only slightly harder, but again, doing so isn't necessarily legal.

First of all, the MP3 format and encoding process is a patent minefield. Many organisations own (or claim to own) patents on the various algorithms used in the encoding or decoding of MP3s. The complete MPEG-1 standard (parts one, two and three) on which MP3 is based was published in December 1991 as ISO CD 11172. Patents filed before that date are presumably still valid, but those afterwards have prior art to contend with (there's a leeway of one year in the USA). This means that all MP3-related patents filed before December 1992 are still valid until December 2012.

Ripping CDs to produce MP3s is another problem area. In the USA, it's legal to make a copy of a CD for your own use. In the UK, strictly speaking, it's illegal to do so unless you have the permission of the copyright owner. The UK government had made noises about allowing this common practice (55 per cent of Britons think it's legal and 59 per cent admit to doing it, according to one survey).

Once you've made your own MP3s for personal use (or indeed purchased them), it's illegal everywhere to make copies for distribution. In fact, the RIAA in the USA has made a name for itself by going after and suing for thousands of dollars individuals who are shown to have deployed their own MP3s, or to have downloaded large numbers of songs from peer-to-peer networks. ■

The solution would seem to be to compress the digital audio data. As it happens, compressing a typical CD track with something like the Deflate algorithm in Zip doesn't actually give many space savings. The reason is the data stream exhibits randomness: the two-byte accuracy of the sampling means that even similar pieces of music encode slightly differently, negating the benefits of dictionary compression algorithms such as Deflate. Random data doesn't compress, so CD tracks don't compress terribly well at all.

### Trimming the edges

The next solution is to use a lossy compression scheme. Such a scheme essentially throws away unimportant data in order to make the result more compressible. On decompressing the data, the algorithm doesn't produce exactly the same output as the original input, but we don't

notice the difference. This kind of algorithm is therefore only of any real use for things such as images, video, and audio.

For images and photos, the archetypal lossy compression algorithm is the JPG file format. The vast majority of lower-end digital cameras produce JPG images as a matter of course. The reasons are primarily to do with smaller file sizes: more photos can be stored on the camera's internal flash storage, and transferring photos to a computer takes less time. Of course, the fact that the vast majority of digital photos are only viewed on a computer screen (sometimes as thumbnails more often than full size) and never digitally manipulated that much means that JPGs are more than sufficient. High-end DSLRs and professional cameras use a RAW format, which, although it may be compressed, isn't lossy-compressed. We don't usually notice that JPG is a lossy compression format because the algorithm only discards information that the human eye would have difficulty perceiving when viewed alongside other parts of the photo.

With audio compression, we take advantage of the imperfect nature of the human ear to help us identify (and discard) unimportant parts of the music: there are frequencies we can't hear, there are frequencies we distinguish better than others, and when two sounds play at the same time, we hear the louder sound rather than the softer one.

### Why use MP3?

The MP3 algorithm uses these details to remove those sounds we can't hear (or have difficulty in perceiving among the rest of the audio) to simplify the data stream to make it more compressible. The idea is to tweak things so that the removed data does not hurt the quality of the audio for the eventual listener. Nevertheless, to make things plain, MP3 cannot produce CD quality audio since it eliminates information from the data stream;

instead we call the result near-CD quality or even FM quality. However, the compression ratio we obtain is truly remarkable: three-minute MP3 tracks are typically between 3MB and 5MB in size – about an order of magnitude smaller than the original CD track.

The MP3 algorithm has a single tuning knob that enables us to determine how much information is thrown away. Some people will be fine with increasing the lossy part of the compression algorithm because they only listen to MP3s in a noisy environment – in a car or a busy office, for example. Within a noisy environment, you won't hear the most subtle sounds, so it makes sense to optimise for file size rather than audio quality. If you're listening to music in a quieter environment, such as at home in your living room, you may be more aware of the loss of quality and not so bothered by file size. The lossy algorithm tuning knob is known as the bit rate.

Bit rates are measured in bits per second; MP3 varies from 96kbps to 320kbps. At the low end of the scale, 96kbps or 128kbps is equivalent to FM radio. At the high end of the scale – say 256kbps to 320kbps – the sound quality is comparable to that of a CD.

## The speed of sound

Remember that a CD delivers data at a rate of 176KB/s, or 1,400kbps. This means that a song saved at the 96kbps bit rate is roughly 1/14 of the size of a CD

track. At the 256kbps bit rate, files are about a fifth of the size. So, for example, if your

car's CD player can play MP3 CDs (that is, data CDs containing MP3s), you'll be able to put five times as many 256kbps bit rate MP3 tracks on the CD as you could on a standard audio CD.

**“ In general, people can't tell the difference between a 256kbps MP3 and a CD ”**

between an audio track encoded as a 256kbps MP3 and one from a CD. The only significant

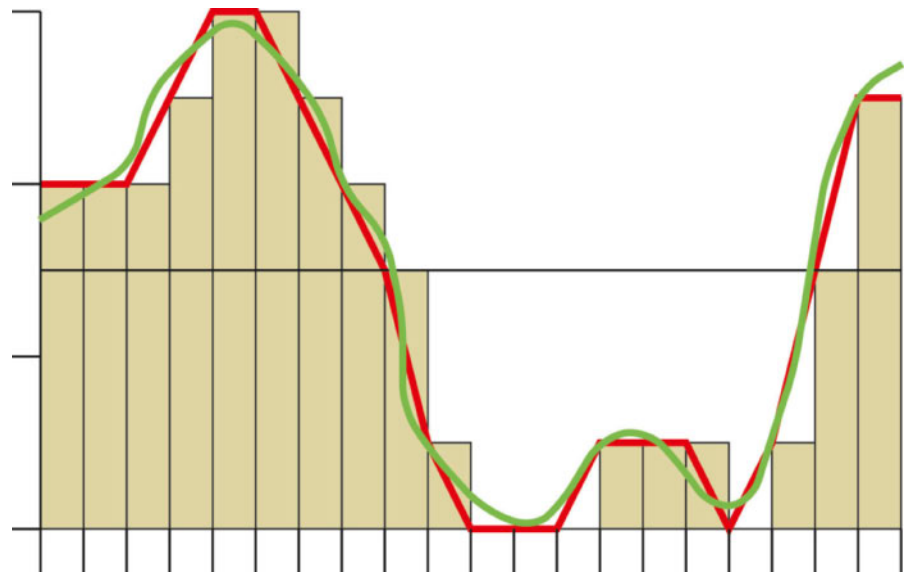
statistic is that if you know a particular track very well from CD, you're more likely to spot an MP3-encoded version of it than if you're listening to a track you've never heard before.

The MP3 file format was designed to contain more than just the lossy-compressed audio data. The file consists of a set of MP3 frames, each comprising a header and corresponding data. A set of frames may be enclosed inside a tag to indicate that the frames are describing something special, such as metadata about the MP3 track (the artist's name, title, album, track number, musical genre, album art and so on).

## Add your own data

Although the MP3 standard doesn't define its own standards for these metadata tags, there are two that have grown into standards through being recognised by several audio players. These are the ID3v1 and ID3v2 tags – although there's also a new one called APEv2, which is gaining familiarity and approval. On playback, the metadata tags are generally read by the audio player, so that relevant information can be displayed for the user.

Although many CD rippers create metadata for your tracks and embed them in the MP3 files (and programs such as iTunes enable you



▲ **Figure 3: Sampling the waveform with a faster sample rate and low accuracy.**

## AAC

MP3 isn't the only lossy compression algorithm out there. The next most familiar one is AAC (advanced audio coding), mainly because it's the default audio format used by Apple's iTunes software for CD ripping, and by the iTunes Store for deployment of audio tracks. It's supported by Apple's audio devices and Android phones, as well as gaming devices from Sony and Nintendo. Many software players also support AAC, including Windows Media Player as of Windows 7.

The benefits of AAC over MP3 are that it adds more sampling frequencies, uses variable bit rate, is more efficient at encoding both stationary and transient signals, and so on. Blind tests show that, using the same bit rate, listeners rate AAC as having better quality than MP3. As the box below left on variable bit rate indicates, this is probably due to VBR more than anything else. ■

to edit the metadata for your tracks), there are MP3 tag editors that allow you to manipulate the metadata at a finer level, or in block mode.

All in all, MP3s have changed the music environment for good. Although you can still buy CDs – or if you're really old-school, vinyl – most people consume their music through MP3 or AAC. Online retailers such as Amazon and iTunes help you buy MP3s for immediate download and gratification. Online radio stations, including Pandora and Spotify, enable you to listen to lossy-compressed streamed tracks without the need for purchase. Programs like iTunes and Windows Media Player enable you to rip your CDs as MP3s onto your hard disk for later listening. Audio players such as the iPod and Zune let you to listen to your MP3-encoded music wherever you want to. In short, MP3s are here to stay. **PCP**

*Julian M Bucknall has worked for companies ranging from TurboPower to Microsoft, and is now CTO for Developer Express. [feedback@peplus.co.uk](mailto:feedback@peplus.co.uk)*

## Variable bit rate

Traditional MP3 files are encoded at a constant bit rate: you decide the level of quality you want at the outset and then rip the track from the CD at that bit rate. Variable bit rate encoding (VBR) does away with that decision and makes the choice for you as it encodes the track instead. Quieter or silent parts of the track may be encoded at some very low bit rate (such as 32kbps, which really muddies a track if you encode the whole thing at that rate), whereas parts with a high dynamic are encoded at a much higher bit rate – up to 320kbps. Overall, the file size for a track may be the same as that of a constant 160kbps MP3, but the audio quality will be much better.

The issue with VBR is that the algorithms are much more complex and take longer to encode a track than a CBR MP3. After all, the algorithm has to continually look ahead to determine the best bit rate to use on the current part of the track, and some VBR encoders even read the input audio two or more times in order to get the best quality.

On the output side, most modern DACs, such as those found in personal audio players, including iPods, support VBR for playback. ■