CHAPTER 3 AUDIO MEDIA

3.1 Background

It is important to understand how the various audio software is distributed in order to plan for its use. Today, there are so many audio media formats that sorting them out is no small task. Years ago it was simple—analog on either a vinyl disk or magnetic tape. The two media were roughly equivalent although they both had different (obvious) flaws. I will discuss the analog formats briefly and then the non-compressed digital formats, ending with a detailed discussion of the new field of compressed audio, which is clearly the future.

3.1.a Analog

Analog formats basically record some mechanical or magnetic characteristic which is an equivalent copy or "analog" of the original waveform. For LPs, it was the actual mechanical displacement of the groove while for magnetic it was the strength of the magnetic field placed on a magnetically coated backing. It is interesting that today's hard disks are in fact an evolution of the magnetic tape of the early days. Some early PC's actually used cassette tapes for their mass storage needs (when 64k was a big program). As ridiculous as this sounds, it sure beat paper tape! (A one inch wide very

long strip of paper with holes punched in it.) In principle, analog recording is the highest quality since there are no conversions of the signal required at any point. However, in practice this is never the case. For LPs, the mechanical integrity of the grove is a big issue. Dust or minor scratches degrade the "image" recorded on these surfaces as does wear from the actual playback process. One could, and this has been done, sense the LP groove optically with no resulting wear of the media, but the dust and scratch problems still remain. There is also no reason why magnetic tape could not be made higher quality given the refinements in sensors and magnetic coatings that have resulted in the last few years from the magnetic storage industry, thereby yielding a near perfect analog medium.

Clearly analog storage is not an unreasonable proposition, it is simply that **digital** is now so prolific in our everyday life that analog storage is just no longer viable as a competitive technology. Add to this situation the tremendous peripheral capabilities of digital and it is a hands-down winner. What exactly are these new digital technologies that are making digital so ubiquitous?

3.1.b Digital Signals

As a subject of discussion, digital audio is enormous. By now most people are comfortable with the digital interpretation of audio thanks in many ways to the computer. When one can actually see the digitized signal with a computer program like Cool_Edit, then the understanding of how digital sampling works follows very quickly. If you have not looked at a wave file (an audio file with a *.wav extension that is an exact sampled representation of an actual acoustic waveform, we will discuss wav files a little later) with a program like Cool_Edit, then I highly recommend that you do so. Even experienced practitioners, like myself, find it enlightening to actually see what it is that we are listening to as we listen. Nothing quite compares with a direct experience like this for understanding a technology such as digital audio.

Sample Rate
the number of sound measurements (data samples) made per second.

Bits of resolution
the number of data bits in the data sample.

In the digital world there are two factors that control the quality of the recorded signal. The first is the **sample rate**, or how often a sample of the waveform is measured for storage to the recording medium, and the second is the **data word length** or how many **bits of resolution** are used to encode the value of that sample. The current standard, that found on a CD, is a

sample rate of 44.1 kHz and a word length of 16 bits. An emerging standard (one in which I am dubious about) is a 96 kHz sample rate with a 24 bit word length. Later on in this chapter, I will show some strong evidence for why this higher resolution may be a waste of bandwidth (a word which I will also define in more detail a little later). Cool_Edit can use a number of different sample rates and word lengths and the reader is encouraged to experiment with these variations. Since I will be encouraging the use of a computer as the base platform for the AV sound and image processing, it is a good idea to become somewhat proficient with some of the more common applications and techniques. At the time of this writing, Cool_Edit, for example, is free with limited capabilities and is readily available on the net. The full version is also well worth the purchase price. As a base platform for audio file manipulation it is unsurpassed. (Cool Edit is no longer available having been bought by Adobe and converted into Encore. There are many Cool Edit facsimiles, but none seem as good as Cool Edit.)

Consider now a stereo signal sampled at 44.1 kHz with a 16 bit word. This signal has to pass:

$$44{,}100\ {}^{samples}\!/_{\sec} \times 16\ {}^{bits}\!/_{sample} \times 2\ channels = 1.4\ {}^{megabits}\!/_{\sec}$$

Bit rate
the number of bits sent through a channel per second.

This value is known as the **bit rate** and it means that every seven seconds a megabyte of data has to be processed. Today, this seems like an achievable task, but even a few years ago this was a major feat. From a storage perspective, this is still a huge amount of data. A 3.5" floppy can only store about 10 seconds of audio at this bit rate. And even today a fast ethernet home network could not pass this data rate without some error or compression. Fast internet hookups, like the higher bandwidth ADSL lines, can only carry this amount of data in real time with some difficulty. Clearly, passing audio around computer systems requires a substantial reduction in the data rate or **bandwidth** as it is also known.

Bandwidth
in digital data steams—the width of the frequency band that is required to carry the data.

Frequency Modulation
the technique of changing the instantaneous frequency of a waveform as a means to encode data.

The term bandwidth originated literally from the width of the **Frequency Modulation** (FM) signal that is required to transport a given signal. Generally, about 1.5 Hz of bandwidth is required for each bit in the bit stream. That means the minimum required bandwidth of the above stereo signal is about 2–3 MHz (mega-Hertz) for transmission as a digital signal. In general, virtually all data transmission is done by frequency modulated data transmission techniques—like FM radio. In FM, the carrier frequency is

Carrier frequency
the base frequency about which the FM signal is modulated.

Channel
in digital data streams, the path taken by the data stream.

the center frequency about which the frequency modulates, which is how the data is encoded. The carrier frequency can be placed anywhere, as long as it is greater than the modulation frequency bandwidth. The modulation frequency, or bandwidth, tends to be fixed by the data requirements. The minimum **carrier frequency** required to do this task is about 1.5 MHz with a modulation frequency of about 1.5 MHz. We can see that the important parameter is the data bandwidth since this determines how much data a **channel** can carry. When looked at in this way, transmission of digital data are real bandwidth hogs. After all, the physical bandwidth of the audio signal is only two times 44.1 kHz or about 100 kHz. However, when the carrier frequency is 100 MHz, then 1.5 MHz modulation is not much of a problem and when the carrier is in the Gigahertz range, as in cell phones, etc. then this bandwidth is even less of a burden. Unfortunately, computer networks are not quite this fast—at least not yet. At any rate, it should be obvious that for high quality audio to be useful on a computer, some substantial reductions in bandwidth are required.

The first obvious way of reducing bandwidth is to reduce the sample rate and/or the word length. These reductions have obvious detrimental effects on the sound quality but do represent an often used technique for bandwidth reduction at the extreme. Some webcast radio stations have extremely low sample rates and word length (along with the techniques that I will discuss below) to allow them to be transmitted over the internet. For voice signals these reductions are generally benign, but for music, these techniques certainly lack appeal.

Loss-less Encoding
data encoding which returns an exact duplicate of the original

Coding
the technique of data reduction using a computer algorithm.

The next approach we could take is to consider compressing the actual bit stream by looking for patterns or long strings of 1's and 0's, as would often occur in an audio signal file. This type of compression is called **loss-less** because the original digital bit stream can be reconstructed exactly with no loss in data. There are numerous techniques used in loss-less **coding** (the term given to compression techniques when applied to AV signals), but in essence they work exactly like common software programs that create *.zip files (WinZip, etc.). In fact, loss-less coding can be easily performed by simply taking a *.wav file and creating a zip file out of it. Compressions vary dramatically depending on the *.wav file but, typically, only about a 10% reduction in file size is achieved with this technique. Windows Media Player has a loss-less encoding option which seems quite effective. Our media storage is all encoded in this loss-less format.

The next class of data compression is called **perceptual audio coding**. With these techniques, and there are currently a proliferation of them, any data that cannot be "perceived" by the listener, and hence is superfluous, is removed. Perceptual coding forms the basis for virtually all modern audio **codecs** (the term for an audio coder/decoder) in use today, so I will take some time to explain the background for these techniques.

Codec
the input and output algorithms for encoding and decoding a data stream.

3.2 Perceptual Coding

Now that the reader has a basic understanding of hearing, spectral masking (Chapter 2) and the concept of the frequency domain (Chapter 1), I will put this information to use.

The masking of one signal by another is the fundamental characteristic of human hearing that makes much of the data in an audio signal imperceptible and, hence, superfluous. Consider a signal composed of two sine waves, one at 500 Hz with a level of say 80 dB (SPL) and another at 1 kHz at 40 dB (SPL). In this example the second tone would be completely inaudible (see Figure 2-4 on pg. 37). By actually eliminating the second tone, we can reduce the data requirements by only coding the 500 Hz tone. This data analysis must be done in the frequency domain since in the time domain the frequency masking effect is not evident. There are time domain masking effects, as I discussed in Chapter 2, and these are also used in many codecs to further reduce the data, but these effects are secondary to frequency domain masking.

What about a complex signal? How can I determine what data can be eliminated and what cannot? First, consider the following situation. For a low level signal there is very little masking. Low level signals use only a few low order bits with the upper bits being set to zero. Low level signals will benefit from the loss-less coding portion of the codec. For high level passages there is a significant amount of masking, so these passages benefit most from perceptual coding techniques. When combined, these techniques lead to a fairly consistent data reduction level for the entire signal.

In an actual codec, the incoming audio signal is filtered into a number of bins using a bank of filters which are set to represent the ears' critical bands

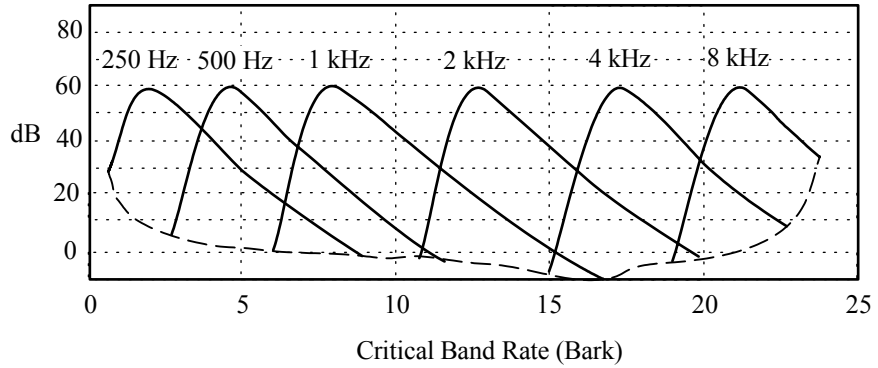

Figure 3-1. Masking effect versus critical bands.

(see Chapter 2 pg. 38). The reason for this choice of filters is that the masking effect is independent of frequency when plotted versus critical band rate as shown in Figure 3-1. The scale of this figure is in **Barks**, which is a sort of frequency warping that makes the critical band filters appear to have similar shapes. This allows for a single masking algorithm for all of the bins. There are typically 32 bandpass filters or bins set to cover the entire audible frequency range in a perceptual coder.

Bark
the unit of proportional frequency used in psychoacoustic data.

If, in our above example, I had set the secondary tone to 60 dB instead of 40 dB, then this secondary tone would now be audible; however it would only have a dynamic range of 20 dB (the level above the masking level) due to the presence of the masking tone at 500 Hz. This dynamic range, which is essentially a signal to noise ratio, is called the **signal to masker ratio**. Returning now to a complex signal passing through the codec I would calculate the masking effect from each adjacent bin to determined a signal to masker level for each of the 32 bins.

Signal to masker ratio
the ratio of the signal level to the masker level in a critical band.

Coefficient
the level of an FFT bin at a particular frequency.

Knowing the signal to masker level in each bin allows me to determine the minimum number of bits that are required for each **coefficient** (the value of the FFT at a particular frequency) in order to represent the signal at that frequency with the required signal to masker ratio. That is, the value of the frequency components in each bin are represented by numbers which have no more dynamic range than the ear is able to detect. It is well known that each bit in a coefficient contributes 6 dB of dynamic range. Rather than using 16 bits to represent the value at each frequency in a particular critical band, only the number of bits required to just exceed the signal to masker ratio are used. The reduced number of bits in a coefficient in a bin will

Quantization error
the noise created by the finite bit depth in the data word.

cause data noise, **quantization error**, to appear in that bin, but this noise will not be perceived since it is masked.

In our two tone example, I would need

$$\frac{80\,dB\,(500\,Hz.)}{6\,dB} = 14\,bits$$

for each coefficient in the critical band containing the 500Hz tone, plus

$$\frac{20\,dB\,(1000\,Hz.)}{6\,dB} = 4\,bits$$

in the critical band containing the 1kHz tone and no bits in any of the other bins. If the FFT of the signal has 1024 points and there are 32 critical bands covering this signal, then each band has 32 coefficients. (This is not exactly correct since the critical bands are not all the same number of FFT points, but you get the picture.) The total data that would be required in this example is:

$$32\,coefficients \times 14\,bits + 32\,coefficients \times 4\,bits = 576\,bits$$

for the compressed signal versus

$$1024\,coefficients \times 16\,bits = 16,384\,bits$$

for an uncompressed signal. This is a remarkable reduction in data for little or no compromise in perceived sound quality.

This example is extreme since most of the frequency domain is empty and real signals would not usually have this advantage. Still, it is quite apparent that this is a dramatic reduction in the data required to represent the compressed signal with the same perceived signal to noise ratio as the original.

By taking an inverse FFT at the transmission reception end, I can reconstruct the data in time blocks and link them together in a continuous chain to recreate a reasonable facsimile of the original signal, from a perceptual point of view. However, It is highly unlikely that the reconstructed time signal would still look like the original.

In effect, we have a technique whereby the masking effect of the human hearing system can be used to reduce the information bandwidth required to model a perceptually equivalent signal with much less data. It is only per-

ceptually equivalent because in the time domain the recreated signal will look very different than the original signal. The signal has in fact been significantly distorted, but in a way that is imperceptible. This brings us back to a point that I made in an earlier chapter. Altering the time domain waveform of a signal visually (and mathematically) distorts the signal, but, because of masking effects, we cannot conclude from this that this alteration causes a perceived degradation in the sound quality. Standard linear system tests for distortion, like Total Harmonic Distortion (THD) or Intermodulation Distortion (IMD), when performed on a perceptual codec can produce extremely high numbers, but I have shown how these distortions could be inaudible. The conclusion that can be drawn from this simple fact is that standard linear systems theory should not be applied to a system where perception is the basis of evaluation. If one wants to evaluate the relationship between the measurement domain and the perceptual one, they have to consider the characteristics of the hearing mechanism, the human ear. This evaluation system is not a linear system and one should not use simple linear system theory in this kind of evaluation. Perception is a highly nonlinear function.

In Chapter 1, I talked about the distortion in a system, more specifically, in an amplifier. It is important to realize that it is not the levels of THD or IMD that indicate a problem with the system, but how these systems act on the actual signals passed through them. This is a complex issue. It turns out that it is the higher harmonics of a distorted signal that pose the greatest audible deficiency and that these defects are more audible at lower signal levels. This is exactly why I stated that the distortion must not rise with lower signal levels in a system-this is a clear indication of poor sound quality.

3.2.a MP3

With the basics under our belt lets take at look at the most common perceptual coder the Motion Pictures Experts Group (MPEG) Layer 3 codec. Dating back to the early 90's the MPEG began to study ways of reducing the transmission data for movies, television, etc. We will see in Chapter 8 how this group has also defined the current standards for video data reduction. In this chapter, I am primarily concerned how they defined the standards for audio reduction.

MPEG defined three alternatives for audio compression, called layers:

Layer 1) uses a single data block of 384 samples in 32 critical bands where only frequency masking is used.

Layer 2) uses three data blocks, total of 1152 samples, which does a cursory time domain masking simulation.

Layer 3) uses an improved auditory filter model (critical bands) which vary with frequency. It also adds in a more sophisticated time domain masking model and does stereo redundancy reduction. The data is finally reduced with a bit reduction scheme.

Layer 1 has about a 4:1 reduction, layer 2 about 6:1 and Layer 3 an impressive 12:1. It should be noted that MP3 is one of the earliest codecs and that many newer algorithms are known to work better. Still MP3 remains as one of the most prevalent and its use is extremely widespread.

3.2.b Windows Media

Microsoft has one of the leading algorithms from a perception standpoint—WMA. It is currently part of the Windows Media Player and has several fixed bit rates, a **variable bit rate** and a loss-less option. One can also create.wav files from a CD with Media Player. A variable bit rate does not have a fixed allocation of bits, it uses more when needed and less when possible. Typically, variable bit rate codecs will sound better as they make better use of the bits that are available. Variable bit rate codecs make a lot of sense when files are not being transmitted over band limited channels. For instance, when a collection of songs are stored on a PC the bit rate is only important from the storage point of view and the extra space of the variable bit rate is not a significant factor. Fixed bit rates work better for transmission since the bandwidth required is a know quantity.

Variable bit rate
a technique were the instantaneous bit rate is allowed to fluctuate.

3.2.c Dolby Digital

Also called AC-3, this algorithm differs slightly from other standard techniques in that it encodes the bin coefficients **mantissa** (the decimal portion) and **exponent** differently. It uses the exponent to track the waveforms envelope and allocates the bits in the mantissa based on the envelope and a psychoacoustics model, much like MP3. AC-3 is a 5.1 channel system

Mantissa
the decimal portion of an exponential number representation.

Exponent
the power of ten that scales the mantissa.

which means that it has five discrete full bandwidth channels and one limited bandwidth channel for a sub woofer.

AC-3 is currently the broadcast standard for HDTV and is the most common format for DVD; This makes AC-3 an important codec since in a few years it is quite possible that more audio will be coded in this format than any other.

3.2.d DTS

DTS is an eclectic mixture of compression schemes, some of which are perceptual based and others are not. DTS is currently a competitor to Dolby Digital in the 5.1 arena. The interested reader is referred to their web site for further information. DTS is popular for music based DVDs.

3.2.e Others

There are many other perceptual coders that have appeared and some have even found implementation in significant areas. Advanced Audio Coding (AAC) is an MPEG-4 audio standard which has had contributions from a number of companies including Fraunhofer, AT&T and Sony. It is optimized for low bit rates as would be used in Digital Satellite systems. It is licensed by Dolby Labs.

3.3 Conclusion

In conclusion, I want to make a few comments about compressed audio based on my own personal experiences. First, compressed audio is, for the most part, comparable to uncompressed audio if a good codec is used and a sufficiently high bit rate. It seems unreasonable to me to use anything but the highest bit rate available if one is storing music for playback in a quality Home Theater. Storage is just too cheap to worry about storage requirements. I think that it would be unfair to rank codecs other than to say that good ones do exist (while some are pretty bad). I am not saying that compressed audio is indistinguishable from uncompressed audio because under carefully controlled tests, with well chosen source material, the two can always be distinguished from each other. But the better codecs at the higher

bit rates take an extremely sophisticated listener to detect and even then only on some program material. Obviously loss-less encoding is inaudible.

Ripping
the act of converting uncompressed audio into compressed audio.

The point is that compressed audio offers so many advantages that the small degradation in performance is, in my opinion, well worth the trade-off. I have **ripped** (used a perceptual coder to encode a song) my entire CD collection into a central computer on a home network. As a result, the access to my music is unprecedented. Gone are the days of searching through stacks of CD's trying to find the song that I am looking for or the tedious organizing that is required in order to better facilitate a search. I can find and play any song in my collection of 700+ CD's from any room in my house in seconds. With loss-less encoding the original CD is only needed once and it can be put away for good. (I don't encourage illegal ripping of music, it's simply not necessary, and clearly unfair to the artist.) The sound of the ripped file is fine for almost any situation.