

# WDM Audio in the Professional Environment

When Microsoft released Windows<sup>®</sup> 98, it was the first platform to support the Windows Driver Model (WDM). Prior to that, hardware developers had to write completely different drivers for the “Windows<sup>®</sup> 9x” and “Windows NT<sup>®</sup>” platforms. WDM allowed hardware developers to write a single driver that was source-code compatible across all Windows<sup>®</sup> platforms.

Along with WDM came WDM Audio. WDM Audio brought the same “single driver source for all platforms” paradigm to the audio realm, and also made audio driver development easier by requiring hardware developers to write only “miniport” drivers that provided the parts of the driver code that were specific to the particular audio hardware. All the “higher level” code that connected the miniport drivers to the rest of the system was provided by Microsoft. That meant that application software could access the audio hardware using the “waveIn/waveOut” multimedia APIs, the DirectSound APIs and the DirectX Media APIs without any additional driver development.

This new driver model was attractive to many hardware developers. One set of driver code that worked on all platforms from several application APIs made product development and support easier. And the infrastructure that came along with WDM Audio brought new features and opened new markets. A Kernel Mixer (KMixer) allowed software mixing of multiple audio streams. USB and IEEE 1394 audio devices could be used in addition to ISA and PCI audio cards. Microsoft supplied “in the box” drivers for USB and IEEE 1394 audio devices that followed the relevant device specifications. Acoustic echo cancellation (AEC) was added to support telephony applications. Presentation of 3D audio using both stereo and surround sound speaker configurations became easier.

In consumer applications, these features are very helpful. But for the professional audio engineer, use of WDM Audio drivers can have unexpected – and undesired – effects. This paper describes some of the aspects of WDM Audio that should be considered and planned for.

## 1. Sample rate conversion

KMixer has a set of rules that govern when sample rate conversion is used. When first opening an audio device, KMixer tries to use a sample rate of either 44.1 kHz or the highest sample rate supported by the device, whichever is greater. When a new audio stream is mixed to an audio device that is already open, if the sample rate of the new stream is higher than the current sample rate, KMixer pauses the audio stream and tries to change the sample rate on-the-fly to the rate of the new stream. Otherwise, KMixer does sample rate conversion from the rate of the new stream to the current sample rate and then mixes the new stream in with the current audio stream. Sample rate conversion also has implications for multi-channel cards that have clock restrictions. When one channel is open at one sample rate, and an application tries to open a second channel at a different sample rate, the request for a different sample rate might fail due to clock restrictions in the hardware. In that case, the second channel will be successfully opened at the first channel’s sample rate, and KMixer will do sample rate conversion from the second channel’s rate to the first channel’s rate to play the audio. There are also maximum sample rate restrictions that depend on the platform in use. Windows<sup>®</sup> Server 2003 and

## WDM Audio in the Professional Environment

Windows® XP Service Pack 1 systems allow a maximum sample rate of 200 kHz. Earlier systems are limited to a maximum sample rate of 100 kHz.

### 2. Bit depth conversion

When the bit depth of the audio stream being played does not match the bit depth currently in use, K Mixer does bit depth conversion when necessary. This can have unexpected effects, especially when transferring digital audio streams to and from external devices. When an application opens a device at 16 bits, the system may still open the audio hardware at 24 bits, resulting in an unwanted bit depth conversion that may result in a loss of data integrity. And on some platforms, there is a bug in K Mixer that truncates 24-bit audio to 16 bits on playback. Windows® Me systems must install QFE #25334 (see MS Knowledge Base article Q316638) and Windows® 2000 systems must install Service Pack 3 or later (see MS Knowledge base article Q308883) to resolve this issue.

### 3. Mixing multiple audio streams

As described above, when two or more applications try to play audio streams simultaneously, K Mixer mixes the audio together into a single audio stream for playback. This can cause unexpected results. For example, a “new mail arrived” chime or a taskbar “balloon” message sound may unexpectedly occur in the middle of an audio stream being played out of your default sound device.

### 4. Software volume control

The “Wave”, “SW Synth” and “Master Volume” sliders in the Windows Mixer can be used by K Mixer to adjust the volume of the audio. In certain cases, these sliders are not set to unity gain by default, resulting in unexpected attenuation or amplification. These volume changes can be disastrous in the case of AES/EBU or S/PDIF digital data transfers. If the volume is not set to unity gain, a transfer will not be bit-perfect. And if volume changes are applied to AC3 digital data streams, the result is unusable.

### 5. Audio buffering to prevent dropouts

By default, K Mixer uses 3 buffers, each 10 msec long, to hold processed audio that is waiting to be played, resulting in a net latency of 30 msec. In some applications, this latency may be unacceptable.

### 6. Limitations on the number of audio device interfaces

Windows® 2000 has a limit of 10 device interfaces per device type. Windows® XP increases this limit from 10 to 32. Windows® Me and earlier platforms do not have a limit on the number of device interfaces. These limitations may cause problems in multi-channel applications.

These effects can be surprising for the professional audio engineer. In the past, when an engineer configured an application to open an audio device at a given sample rate and bit depth, that’s what happened. If a combination of sample rates or bit depths wasn’t supported, the application presented an error message to the engineer, and the engineer had an opportunity to resolve the conflict. If an audio device was being used by one application, and a second application tried to open the same audio device, the second application would fail. Pro audio hardware vendors could disconnect themselves from the Windows Mixer, ensuring no unexpected volume changes could occur.

## WDM Audio in the Professional Environment

Fortunately, there are solutions for most of these problems.

1. Disable system sounds  
In a professional environment, system sounds should be disabled to eliminate undesired sample rate changes and unexpected audio output.
2. Don't run unnecessary software  
Closing unneeded programs and taskbar applets reduces the possibility of unexpected system activity and audio device contention.
3. Set your Windows Mixer faders to unity gain  
Setting your Windows Mixer faders to unity gain is the best approach when using professional audio applications. This allows the audio application software (and, therefore, the audio engineer) to explicitly control the volume of the audio stream.
4. Upgrade to Windows<sup>®</sup> XP  
With each new operating system release, Microsoft has improved the quality of the WDM Audio architecture. By far, the best WDM Audio support is available in Windows<sup>®</sup> XP. The increased reliability of Windows<sup>®</sup> XP is also a great benefit.
5. When possible, use a common sample rate for your audio material  
This allows the audio engineer to control when and how sample rate conversion occurs.
6. If latency is an issue, use applications that support WDM/KS or ASIO  
Applications that support WDM/KS (WDM Kernel Streaming) or ASIO allow latencies of less than 30 msec to be achieved. Depending on the audio hardware and driver architecture, latencies of less than 10 msec may be possible.

As in all other aspects of audio engineering, following good engineering practices is a prerequisite for success. Being aware of what's going on "under the hood" in the WDM Audio environment – and avoiding the pitfalls – is an essential part of these practices.

### References:

Microsoft Windows Server 2003 DDK Documentation

Copyright © 2003, Digital Audio Labs division of LR Sound, Inc. All rights reserved.  
NO WARRANTY. This document is provided for educational and informational purposes only. Use of the information in this document is solely at the risk of the user. Digital Audio Labs and LR Sound, Inc. shall in no way be held liable for damages of any kind resulting from use of the information in this document, including, without limitation, damages for loss of business profits, business interruption, loss of business information, or other pecuniary loss.\*

---

\* Our lawyers made us say that