

## Re: WaveIn notification methods

**Source:**

<http://www.tech-archive.net/Archive/Development/microsoft.public.win32.programmer.mmedia/2004-10/0081.html>

---

**From:** Grant Schenck (*schenckg\_at\_optonline.net*)

**Date:** 10/15/04

Date: Fri, 15 Oct 2004 17:00:08 -0400

Chris,

As always, thanks for your advice and knowledge. This is good stuff and matches my understanding.

Earlier I had issues with the wave in events getting stuck somewhere when I moved other windows on my desktop. I decided to revisit this and have a case working with MS on it. I'm not optimistic in terms of getting to the bottom of it.

I agree on polling vs. other mechanisms. Specifically in my wait I tried using a 5 ms timeout and checking the buffers whether the event popped or I fell out of the wait due to the time out. That didn't help and based on that I can say the issue is that the event isn't popping because the driver is not marking the buffers done. Why the driver stops marking buffers done is beyond me.

Thanks, Grant

"Chris P. [MVP]" <msdn@chrisnet.net> wrote in message news:#IP7elrsEHA.3972@TK2MSFTNGP15.phx.gbl...

> *Grant Schenck wrote:*

>

> > "Are you using `waveInOpen( )`? If so, have you tried with different

> > *callback modes, such as `CALLBACK_FUNCTION`, or `CALLBACK_THREAD`?"*

> >

> > *I looked at the docs and `CALLBACK_THREAD` seems to involve a windows*

> > *message. Searching on goggle the opinions I saw seemed to recommend*

> > *against using it. What to you folks think? Is there any point in*

> > *replacing `CALLBACK_EVENT` with `CALLBACK_THREAD` if I'm trying to achieve*

> > *high performance audio?*

> >

> > *What about `CALLBACK_FUNCTION`? I assume that is better then*

> > *CALLBACK\_THREAD* but is there any reason to think it would be better  
> > then *CALLBACK\_EVENT*?  
>  
> *Many many many many years back I was having the same dilemma as you trying to decide what to use. So many choices! I wrote a test program that implemented all the different methods to see what would work best. Here were my discoveries:*  
>  
> *CALLBACK\_WINDOW* – *Seemed ok at first, but fell apart when there were many window messages flying about. Perhaps the simplest method to implement.*  
>  
> *CALLBACK\_FUNCTION* – *Essentially useless because you can't do any audio handing in the callback. You're choices are to signal an event or post a window message, why created this method beats the \*\*\*\* out of me.*  
>  
> *CALLBACK\_THREAD* – *Didn't work much better than *CALLBACK\_WINDOW*. Makes the code a little cleaner and allows you to elevate the thread priority of the worker thread for a slight performance improvement.*  
>  
> *CALLBACK\_EVENT* – *Worked well on the NT systems. Requires a dedicated worker thread to operate at full potential. The worker thread must be using *WaitOnSingleObject()* or *WaitOnMultipleObjects()*, using any function call that generated or processed messages such as *MsgWaitForMultipleObjects()* seemed to have a negative impact on performance.*  
>  
> *Now the method you probably didn't think about – *CALLBACK\_NULL*. Yes you might be suprised to find out that the lowest latency solution in many cases is not to use any callback method at all but rather to poll the buffer flags. Polling at a rate of 100 to 200 times a second can get you latency in the 20ms range at the expense of a little CPU power. It really doesn't use much if you throw a *Sleep(5)* into the loop. Obviously this requires a*

- > *dedicated thread and ideally the thread should be set to*
- > *THREAD\_PRIORITY\_TIME\_CRITICAL. Make sure there is absolutely no*
- code in*
- > *this high priority thread that can block or you will bring the*
- system to*
- > *it's knees. That means no disk I/O, network etc.*
- >
- > *Which one to use for lowest latency? CALLBACK\_EVENT is typically*
- the best,*
- > *and with special audio drivers can achieve latency down in the 5ms*
- range.*
- > *Second best, if CALLBACK\_EVENT doesn't work out is polling. It's*
- not really*
- > *worth trying to get the latency too low with WaveIn/Out unless the*
- > *manufacturer of the audio device has supplied special drivers as*
- your going*
- > *to be hit by the kmixer 30ms delay anyway.*
- >
- > *-Chris*
- >
- >