Clap and the filter graph claps with you

April 7, 2009

Raymond Chen

One of my colleagues was a fount of ideas, some of them crazy, some of them clever, and some of them both. I think this one counts as both.

To render multimedia content with DirectShow, you build a so-called filter graph. A filter graph represents a series of transformations that are applied to data as it travels through the graph. For example, bytes from a video file may go through a splitter filter which separates the audio from the video data, then the two data streams each go through a respective audio and video decoder, which converts the compressed data into uncompressed sound or video data, and then to a render device which actually plays the sound samples out the speaker or puts the video image on the screen.

One of the components of a filter graph is the reference clock. The purpose of the reference clock is to keep track of time, so that all the parts of the filter graph produce their output in a synchronized manner. If there weren't something keeping track of this, the audio would fall out of sync with the video.

My colleague decided to create an unusual reference clock. Whereas most reference clocks are based off of real-world time, my colleague's reference clock started by taking input from the microphone. It did some analysis of the volume of the incoming sound stream and detected when there were loud impulses of sound. It then generated a stream of clock ticks based on the rate at which those impulses were detected.

In other words, the clock was driven by clapping.

I was lucky enough to be one of the first people to see this crazy reference timer in action. First, you started up the application and picked a video clip. The video clip just sat there. As you started clapping, the video clip started playing. If you clapped at about 80 beats per minute, the video clip played at its normal speed. If you clapped faster, the video clip ran faster. If you clapped slower, the video clip ran slower. If you stopped clapping, the video clip stopped.

It was freaky cool.

Totally useless, but freaky cool.

Update: Commenter hexatron <u>pointed out</u> that the <u>Metronome</u> sample does the same thing as my colleague's crazy reference clock. I don't know if it's literally the same code, but it's functionally equivalent.

<u>Raymond Chen</u>

Follow