
Subject: Re: MediaPlayer is out
Posted by [dolik.rce](#) on Tue, 04 Jan 2011 18:13:19 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hello guys,

I learned quite a bit about ffmpeg reading Koldos sources and the documentation on the web (although it is lacking many important things - almost like U++ docs).

The result of my work in this field is a conclusion that audio/video playback is really possible using only U++ ffmpeg and portaudio. In the attachment you will find a proof of concept video player. It is very simple compared to the one Koldo published, but it is a learning tool for me and as such it is adequate. Its main characteristics are: oversimplified, single threaded, blocking (no UI interaction while playing).

The video display is done using SetSurface, which proved to be fast enough for this purpose (good work, Mirek!). Scaling and YUV->RGBA conversion is done in single step using sws_scale from libsws that comes with ffmpeg. It should use hardware acceleration where available, but on my system everything was done in software. The rescaling uses bilinear algorithm. I tested with bicubic as well but it became very slow for higher resolutions. The audio output is done in the simplest possible way using blocking SoundStream from my Sound package published in bazaar. The blocking nature of the sound API is also misused to provide audio-video syncing. For mute videos there is a quick hack to make Paint() block until it is the right time to output the image.

If you decide to test, I recommend you to try with shorter movies. Once you start it, it runs until the end (or SIGKILL). No seeking, no pause, no interaction while playing. This should be actually easy to fix using threads and some pieces of code from Koldo.

I tested this on my Intel Atom netbook, i.e. with no graphics card to speak of. I got fluent playback for most of the movies on my hard-drive, but didn't test any HD clips. Actually the slowest part is the resizing, so if the HD resolution matches with your monitor resolution, it should play fine. For me there were some sound buffer underflows when playing better quality movies on fullscreen (1024x600), but playing them in native resolution was fine.

I left few RTIMING macros in the code so you can make your idea about how fast each part of processing is. Generally speaking audio decoding is very fast, SetSurface scales linearly with size but the scaling can do weird thing depending on the size and on original resolution.

So to sum it up: ffmpeg+portaudio+U++ is a combination sufficient to do most of the common video processing and SDL dependency will most definitely be dropped soon. Which is IMHO great news

Best regards,
Honza
