Subject: Re: IDL 6.1.1 : how to activate opengl in hardware ?
Posted by Karl Schultz on Fri, 28 Jan 2005 18:05:28 GMT

View Forum Message <> Reply to Message

"David Fanning" <davidf@dfanning.com> wrote in message
news:MPG.1c63fce3a2851d259898f3@news.frii.com...
> =?ISO-8859-1?Q?J=E9r=F4me_Boivin?= writes:
>
>> Ok, I think I understand... when you set RETAIN=2 in Widget_Draw(), then
>> it desactivate hardware rendering !!! With RETAIN=0, I must control the
>> drawing update, but hardware rendering is back ! Is it a bug, or I
>> misunderstood what RETAIN keyword means ?

RETAIN = 2 implies that IDL will try to repair damaged windows on its own without expecting the application to redraw them. The beahvior you see is not a bug. Read on for more details.

> I doubt hardware rendering is deactivated, but having IDL
> maintain backing store definitely means that IDL has to
> render each scene twice, once to the window and once to
> the restoring pixmap. No doubt it would be slower. And,
> depending upon how this is implemented, it may even be
> much slower.

What actually happens here on a Windows platform is that IDL passes a Windows bitmap (DIB) to OpenGL and tells OpenGL to render into it (once) instead of a window. OpenGL reverts to its software renderer to accomplish this. Then the DIB is blitted to the window when it needs to be drawn or when repairing window damage. Effectively, you end up with software rendering.

In a way, the RENDERER property isn't described all that accurately. Rather than the 0/1 values meaning hardware/software, it is a tiny bit more accurate to say OpenGL/Mesa. The hardware/software meaning is a bit easier for most people to understand and works most of the time, this situation being one of the exceptions. Another exception can occur if you have a really basic video card and the supplied OpenGL implementation ends up doing everything in software.

I suppose another way to think of it is that RENDERER=0 means use hardware acceleration whereever possible, and RENDERER=1 means always use the standard, stable, and consistent software renderer.

If you have a nice graphics card, you're really shortchanging yourself if you use only software rendering. The later nVidia and ATI cards run IDL very well and the drivers are updated frequently. The frequent driver updates *can* be a problem. For example, I updated my ATI drivers about a month ago and that caused some IDL object graphics text to stop displaying.

I reverted back to the previous driver to recover. I then installed an updated driver that came out soon after that , and everything looked good. So, you do have to be a bit careful with drivers.

> In the early days, setting RETAIN=2 on an object graphics
> window caused an error. These days it is allowed, but still
> seems rather pointless, since you carry around the means
> to restore the window in your view anyway. No sense carrying
> it around twice! As a rule, I always set RETAIN=0 in all
> object graphics windows.

Yep!

Karl