Subject: Re: how to keep an image object in the window?
Posted by Karl Schultz on Tue, 26 Sep 2006 23:24:25 GMT
View Forum Message <> Reply to Message

On Tue, 26 Sep 2006 12:08:49 -0700, Rick Towler wrote:

> Karl Schultz wrote:
>> "In your case this doesn't matter since
>> you're rendering IDLgrVolume objects which force IDL to use the software
>> renderer anyways."
>>
>> IDLgrVolume uses raycasting to generate an intermediate
>> and private IDLgrImage and then renders the IDLgrImage to the device,
>> using either hardware or software rendering, depending on how the device
>> was set up.
>>
>> Raycasting is very slow, and is a software rendering process, but the
>> result may actally be displayed with hardware rendering. So, putting an
>> IDLgrVolume into a scene doesn't cause an automatic switch to the software
>> renderer.
>>
>
> Thanks for clarifying. Makes sense too. After I posted that I thought
> that I was probably assuming too much.
>
> <sigh>
>
> Do you texture a "billboard" with the output of the raycaster?
>
> -Rick

The raycaster outputs an RGBA image, where the A channel has opacity information. It also outputs a "depth array". The RGBA image is drawn like any other image via IDLgrImage, which currently uses a texture-mapped polygon. The depth information is also written to the depth buffer so that the volume "depth" is communicated to the device in what would otherwise be a simple 2D image rendering step. This allows you to, for example, create an isosurface of the volume and draw it in the same view a the volume itself. The volume rendering, depending on opacity, is then able to obscure the isosurface where appropriate. iVolume is a good place to play with this, since iVolume has an operation that will generate the isosurface and then display it in the same view.

Karl