Subject: Instance drawing + IDLgrVolume
Posted by s[1] on Wed, 12 Feb 2003 10:06:33 GMT

View Forum Message <> Reply to Message

Hey all,

I recently discovered instance drawing for drawing an interactive 3D
cursor (using /DRAW_INSTANCE) over a volume (rendered with
CREATE_INSTANCE=1), to my big surprise it both works really fast and
shows only parts of the cursor that are not hidden by the volume.

My question is: How does the cursor (just a polyline) know where the
volume begins? How does instance drawing work for IDLgrVolumes? Is there
somewhere a hidden depthmap for the volume or is the cursor drawn first
and then the parts of the volume where the cursor is are re-rendered?
Anyway, is there a way to get to know which parts of the window have been
updated by the cursor?

Just curious,

Sebastian

Subject: Re: Instance drawing + IDLgrVolume
Posted by Karl Schultz on Wed, 12 Feb 2003 16:38:34 GMT

View Forum Message <> Reply to Message

"Sebastian" <s@visita2.die.upm.es> wrote in message
news:Pine.LNX.4.44.0302121059190.1555-100000@visita2.die.upm .es...

> Hey all,
>
> I recently discovered instance drawing for drawing an interactive 3D
> cursor (using /DRAW_INSTANCE) over a volume (rendered with
> CREATE_INSTANCE=1), to my big surprise it both works really fast and
> shows only parts of the cursor that are not hidden by the volume.

Instancing is also useful in a more general sense when you have a
complicated scene that takes a long time to render and you want to draw
small, fast dynamic stuff over it, like cursors.

As you have seen, this works very well for volumes because they take so long
to render.

> My question is: How does the cursor (just a polyline) know where the
> volume begins? How does instance drawing work for IDLgrVolumes? Is there
> somewhere a hidden depthmap for the volume or is the cursor drawn first
> and then the parts of the volume where the cursor is are re-rendered?

When you create the instance, IDL saves the color buffer and the depth buffer. If you drew the volume with the ZBUFFER property set, there is information in the depth buffer that was generated as the volume was rendered.

When you draw the saved instance, IDL writes the saved color and depth buffers to the device, essentially putting you back to the same state you were in after the initial draw used to create the instance.

> Anyway, is there a way to get to know which parts of the window have been
> updated by the cursor?

You could read the depth or color buffer before and after the cursor draw and compare.

Karl