Subject: Re: Object Graphics
Posted by Karl Schultz on Thu, 13 May 2004 14:17:09 GMT

"Michael Wallace" <mwallace.no.spam@no.spam.swri.edu.invalid> wrote in
message news:10a2mljqonpogbb@corp.supernews.com...
> I started playing around with Object Graphics today. I've used Java
> objects in IDL plenty of times, but never IDL's own objects. Anyway,
> the first thing I tried to do was create an IDLgrWindow, however IDL said:
>
> IDL> window = obj_new('IDLgrWindow')
> % OBJ_NEW: Creation of backing store pixmap failed.
>   Requesting backing store from the server.
>
> What does this error mean? For what it's worth, this is on Linux,
> specifically Fedora Core 1.

This error is generated when a request to create a pixmap fails. If it had
succeeded, IDL would render the image into this backing store pixmap as well
as the window, and then when an expose event occurs, IDL copies the backing
pixmap into the window.

When IDL can't get a pixmap, it asks the server to perform the backing store
duties, which doesn't always work. As pointed out in other posts, it is a
little more straightforward to just redraw the scene graph on an expose
yourself, especially since you can't completely rely on the backing store.
I suppose that a really robust app could implement both techniques and let
the user select the RETAIN setting. That way, they'd get the benefit of
RETAIN=2 if their system can handle it. (It is pretty easy to do this - just
redraw the scene graph on an expose event. If RETAIN=2 is on and working,
the expose event never comes and the code to redraw the scene graph just
doesn't run.)

I run IDL on FC1 too with a Radeon 9700 Pro. What card do you have?

One interesting thing I noted in the XFree86 start up log is a line that
said that the maximum offscreen pixmap size was 1280x400. I found this a
little disturbing because the card has 128M of memory. So, if you have the
same sort of limitation on your X server and you try to create a reasonably
sized window with RETAIN=2, say 500x500, the request to create that pixmap
will fail. It will fail even if there is enough pixmap memory but the
"shape" isn't right.

This isn't a good thing, and I don't know if this is an XFree86 4.3
characteristic, or something related to the DRI and/or the graphics card
drivers. My GUESS is that they earmark a huge part of the vid ram for
textures, which leaves this "sliver" of free memory after you assign the
rest to the other various buffers (front, back, depth, etc). Further, I

guess that the hardware on the card can only blit blocks of memory that have the same "shape" between the onscreen color buffer and the offscreen pixmap area. In other words, pixmaps can't be stored in offscreen memory in a linear, contiguous portion of memory nor in an arrangement that "fits" the unsed parts of memory.

There may be some Xfree86/DRI/Driver configurations settings that control this sort of thing, but I have not found them yet. It would probably take some digging over in XFree86-land to find them.

You also mentioned that Direct Graphics retain settings worked as expected. I'm a little confused by this since Direct Graphics does the same sort of thing for RETAIN=2. Were your DG windows smaller than the OG windows? Another possibility is that the max pixmap size restriction can be eased when there are no OpenGL contexts on the card.

Karl