Subject: Re: clipboard objects and postscript (unix)
Posted by steinhh on Wed, 04 Nov 1998 08:00:00 GMT
View Forum Message <> Reply to Message

In article <MPG.10a92a5f99bbf1609896fd@news.frii.com> davidf@dfanning.com
(David Fanning) writes:
> Sean O'Sea (seanosea@my-dejanews.com) writes:
>
>> I'm afraid the problem seems to be an unhappy xclipboard. RSI recommended
>> using the buffer object, instead, to avoid an intervening clipboard utility
>> entirely:
>>
>>> The best alternative for creating a PostScript file from Object Graphics
>>> output is to render the graphics to an IDLgrBuffer object, which will,
>>> of course contain raster data. Then, using the IMAGE_DATA keyword to the
>>> GetProperty method of the IDLgrBuffer object, you can retrieve the
>>> graphics data into an image array. Finally, you can switch to the
>>> PostScript device (SET_PLOT, 'PS') and use the TV procedure to render
>>> the extracted image array to a PostScript file.
>
> Sigh...It's come to this, has it? Well, with True-type fonts
> it just might work.

But this would mean that all lines etc are rastered, no?
And if you yank up the resolution, you get *huge* files, right?
This means (IMHO) that Object Graphics is still a nice toy,
but in practice it's useless as a tool for producing publication
quality figures on paper.

I've just decided *not* to use object graphics for some 3D
visualization I need for my thesis, since what I want is
to plot lines etc. in a 3D geometry. Now, this is fully
possible to do in direct graphics with "T3D et al", and
it comes out crisp and clear in PS files.

And I cannot really see why it should be any problem for RSI
to create an Object Graphics -> Direct Graphics "translator".

I'm having a hard time figuring out exactly *which* parts of
the OG implementation that's impossible to recreate in direct
graphics. Sure, shaded surfaces, polyfilled stuff, etc. has
to be pixelized, but those are already pixelized in todays
direct graphics implementation - but without pixelizing
the axes/labels/lines etc.

BTW, the trackball object is just as handy when dealing with
direct graphics as well as object graphics!

Stein Vidar