Subject: Re: PNGs without X?
Posted by Dr. Sven Geier on Fri, 14 Feb 2003 19:21:33 GMT
View Forum Message <> Reply to Message

Boy, I hope top-posting is alright...

A while ago I posted the thing appended below -- and got a few responses. I dug through the verious methods and none of them really work. I had noted before that postscript isn't really an option and the same reasons really apply to plotting into the Z-buffer: neat trick that, and I may well use that in the future; but the output I get is nowhere near the output that I get when plotting to an X window. The fonts are entirely different, the colors are differnt, pretty much everything is different.

The color part, I suppose, is an outcropping of another problem, that I never managed to find a solution to: I find it very hard (to say the least) to use a certain given color in IDL. It seems like the simplest thing in the world to have a graph with four lines and I want one of them to be red and one green and one yellow and one blue. However there doesn't seem to be a mechanism in IDL(?) that accomplishes such a thing. Before I knew of 24-bit displays, I had a self-made colortable (with tvlct()) in which I knew index 1 is red and index 5 is blue and such, but the first time I tried this on a true-color visual, I learned something.

The problem is that I need to be certain what the colors are before the plot, since there are some comments added to the data (via xyouts) that refer to "red line = such-n-such" and such. For the same reason I must be able to rely on a certain plot geometry (which I am consistently not getting when plotting in the Z-buffer) because the positions of things are carefully computed. The entire thing has been evolved over years to work just right and if you suddenly run it with a larger character size (like the Z-buffer seems to do) it'll all blow up.

As an example, here's some random data and label:

```
data = tan(findgen(5))
plot,data,/nodata
oplot,data,color='ff'x
oplot,data,psym=5
xyouts, 1,1.53,' <- triangle'
```

On my 24-bit display this draws a few red lines, white triangles and labels one of the triangles with "triangle". If I do this on a "window,xsize=350,ysize=350" and then I do the same thing into the Z-buffer following David's website (great site, by the way. Or terrible site, as you may look at it: Many idl programmers lose much productive time browsing through it... :) I get something that is "kinda like" the original, except

that the plotting area has a different size, the fonts are different and the color is gone - pretty much as different as a plot can be while still showing the same thing.

!P.font = -1 or 0 give the same result in the Z-buffer, !p.font=1 gives a slightly prettyier (but less legible) one, but all three are different from their counterparts that are plotted into the X-buffer and not one of them is as clear and legible as the p.font=0 in X-windows.

If I plot into the PS device and turn the result into an image like this:

```
gs -q -sDEVICE=ppm -r100 -sOutputFile=- idl.ps | pnmflip -r90 | pnmscale \
-xsize=350 -ysize=350 | pnmtojpeg > test.jpg
```

I do not only get no colored line (because all colors 255 and up are just "white" to postscript, apparently), but the result is transparent (i.e. white background) and the fonts are unreadable. It's easy to fix the color in this simple example by searching for 'setrgbcolor' in the postscript file, but once you have a bunch of colored lines and text, that gets pretty tricky to say the least.

I am writing this all down in a last attempt to see if someone has the magic trick "how to get consistent colors and something like an X-buffer without X" - but if there is none, I'd like to say thanks to those who posted before anyways; there were a few good ideas...

-- S

-
Catch a couple Z's before responding...

Reimar Bauer wrote:

> Dr. Sven Geier wrote:
>
>
> Dear Sven,
>
> did you have thougt about the trick printing into a ps file and then using
> convert or pstoimg (which I prefer) to get the ps file converted in
> whatever you want. (You have an unix system so you can use these routines
> with a spawn too)
>
> This method if it's suitable for you has another effort too. The
> postcriptfonts are much better readable as the image fonts.
>

>
> regards
>
> Reimar
>
>
>
>>
>> Heya all
>>
>> I have a routine that gathers data from various places, then plots it,
>> then uses tvrd(/true) to read the plot into an array and then writes it
>> out as a .png graphics file. This all works fine so far.
>>
>> Since I rarely ever look at the graphs as they're plotted (the pngs go to
>> a web-directory) I switched the plotting to a pixmap, i.e. the
>> idl-process now opens its own window with the /pixmap parameter (and also
>> xsize and ysize and such) and thus the whole shebang works quite nicely
>> without ever showing anything on the screen.
>>
>> Now in the process of automation I'd like to move the whole operation to
>> a server box that does not have X running. Unfortunately it turns out
>> that these "pixmaps" are *X*-pixmaps and that I can't open them without
>> X.
>>
>> So now I'm looking for some way to plot data into/onto a PNG (or GIF or
>> JPG or whatever) without having an actual graphic server running. I
>> played around with plotting into a PS device and using ghostscript to
>> convert that into a JPG, but not only do the colors not come out right
>> (the usual PS stuff) but also the fonts are all off (the !P.font=0 for
>> the 'x' devive is designed to be readable on a 400x400 window, the
>> ps-fonts become illegible when I reduce a PS-plot to that resolution) and
>> similar problems.
>>
>> Anybody know how to do a standard run-of-the mill X-type plot into a file
>> or into an array or into RAM or such?
>>
>> Thanks in advance...
>>
>> -- SG
>>
>>
>

--
Catch a couple Z's before responding...