
Subject: Re: An IDL cron job, true color plots, Xvfb, Z-buffer, and all sorts of troubles

Posted by [David Fanning](#) on Fri, 08 Feb 2008 16:52:41 GMT

[View Forum Message](#) <> [Reply to Message](#)

Katheryn writes:

> So, I am still a little confused. The reason for using a color table
> in the first place is so that I can do the neat trick of scaling the
> colors in my plots to represent some arbitrary distribution of values
> over an arbitrary range, which is the purpose of the actual version of
> this little plotter. Is there a way to do that either with FSC_COLOR
> or some version of color tables in the Z buffer and get the colors I
> really want out?

You load color tables in the usual way with LOADCT,
although you might want to become acquainted with the
keywords NCOLORS and BOTTOM as you gain more experience
with this.

Your problem is that the color table you want to
use to display your data (33) is about the worst possible
choice to use in drawing a plot, since--as you see--
it typically results in a blue background with red
axes. Yuck!

The code I wrote for you yesterday loads some plot
drawing colors into the color table, draws the plot
with something sensible, then loads color table 33
to draw your data. That is one way to handle the
situation. The other way, if you insist on using
indexed color, and--alas--you have to in the PostScript
device, is to divide your color table up into "plot
drawing colors" and "data display colors".

For example, you could do this:

```
LoadCT, 33, NCOLORS=250  
TVLCT, FSC_COLOR(['white', 'navy', 'indian red'], /TRIPLE), 250
```

This gives me 250 "display colors" and three "plotting colors",
while are loaded in indices 250, 251, and 252. Now, all I have
to do is make sure I scale my data into only 250 values, so I
can get the correct color display, and I am good to go:

```
!P.Multi=[0,2,1]  
data = BytScl(dist(256), Top=249)  
Plot, findgen(11), color=251, background=250, /nodata
```

```
Oplot, findgen(11), color=252
TVImage, data
!P.Multi = 0
```

This assumes you are using indexed color, of course. If you wanted to do this in a color set-up independent way, you would use FSC_COLOR to specify the colors on your plot commands. For example, if I wanted my program to display the correct results on my display (whether I was using decomposed color or not) and in PostScript and in the Z-buffer and ... wherever, I would write the code like this:

```
!P.Multi=[0,2,1]
data = Bytscl(dist(256))
Plot, findgen(11), color=fsc_color('navy'), $
    background=fsc_color('white'), /nodata
Oplot, findgen(11), color=fsc_color('indian red')
LoadCT, 33
TVImage, data
!P.Multi = 0
```

Notice here, I don't have to restrict the number of colors for my data, I can use all 256 I have loaded.

A handy way to check which colors you have loaded in your color table is to use CINDEX, another Coyote Library program you have downloaded by now. :)

Cheers,

David

--

David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: <http://www.dfanning.com/>
Sepore ma de ni thui. ("Perhaps thou speakest truth.")
