Subject: Coyote Graphics Update, Feb 4
Posted by David Fanning on Fri, 04 Feb 2011 22:21:11 GMT
View Forum Message <> Reply to Message

Folks,

I can't remember if I announced this or not, probably not, but because of the way resizeable graphics windows work it becomes even more important than usual that graphics display programs be able to "use" their own colors, independent of the hardware color table.

To this end, I have added a PALETTE keyword to cgImage, cgContour, cgSurf, and cgColorbar (and their recently retired predecessors, for that matter) that allows you to pass a color table palette into the program. A palette is more convenient than a color table index number, because you are not then limited to IDL's color tables, which I seldom use anymore for real work. :-)

The easiest way to get a palette is from cgLoadCT and the RGB_Table keyword. Obtained in this way, no colors are actually loaded in the hardware color table:

```
IDL> cgLoadCT, 5, RGB_TABLE=stdgammaPal
IDL> cgLoadCT, 4,/Brewer, /Reverse, RGB=brewerPal
```

These palettes can then be passed to any of these four programs and they will do the right thing with them. At the moment cgContour and cgColorbar will sample colors from these color palette vectors to do their thing. I understand this makes it difficult to specify things like out-of-bounds and missing colors, but I am working on a solution to that problem and will announce it shortly. One step at a time. :-)

Naturally, you don't have to use a palette. And if you don't, these programs will continue to work as they always have.

Also, there was some mention about "loading" commands into the cgWindow without immediately executing all the other commands in the window. Rather than add a third keyword to all the Coyote Graphics commands, (as soon as I did, someone would recognize

the need for a fourth and fifth keyword, sure enough!),
I decided to add a window control that that can easily
be turned off with cgControl. The sequence of commands
you use to "load" commands without seeing them flash in the
window now looks like this:

```
cgWindow
cgControl, Execute=0
.... Load all your commands here.
cgControl, Execute=1
```

Now your window works normally.

Cheers,

David

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