
Subject: Re: Trouble with positioning cgColorbars while using !P.Multi

Posted by [David Fanning](#) on Wed, 19 Dec 2012 22:24:55 GMT

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

Leah Huk writes:

> I am creating some postscript files with 4 contour plots each using !p.multi. I would like each plot to have it's own colorbar positioned with it as they all have unique ranges. I've dug around and found that use of the POSITION keyword is not compatible with !p.multi. Is there another way for me to position each colorbar with it's corresponding plot, instead of all on top of each other and in the middle of the plots? Or am I doomed to use the position keyword for every plot and every colorbar? This seems daunting to implement; I have my cgContour and colorbar command embedded within a set of for loops for ease of plotting many many parameter "slices" of a rather large data grid.

I've added a new program to the Coyote Library that will help with this problem. The program is called cgLayout and it can be found here:

<http://www.idlcoyote.com/programs/cglayout.pro>

You use it in a manner similar to the LAYOUT keyword on Coyote Graphics or Function graphics commands. It allows you to set up a grid in a location in a graphics display window. You set the location of the grid by using "outside margins". Once a grid is established, a grid position can be modified by using inside margins, or by setting the ASPECT keyword to return grid positions that are set to a particular aspect ratio.

The default outside margins are set up to accommodate line plots, and a little extra room is left at the top of the page for a title. Here you can plot four plots in a 2 column by 2 row grid:

```
cgDisplay, WID=0
pos = cgLayout([2,2])
FOR j=0,3 DO BEGIN
  cgPlot, cgDemoData(17), NoErase=j NE 0, Position=pos[* ,j], $
    Title='Plot ' + StrTrim(j+1,2)
ENDFOR
cgText, 0.5, 0.925, /Normal, 'Example Plot Layout', $
  Alignment=0.5, CharSize=cgDefCharSize()*1.25
END
```

If you wanted to draw images with color bars, for example, you might do this:

```

cgDisplay, WID=1
cgLoadCT, 22, /Brewer, /Reverse
pos = cgLayout([2,2], OXMargin=[5,5], OYMargin=[5,12], $
    XGap=3, YGap=10)
FOR j=0,3 DO BEGIN
    p = pos[* ,j]
    cgImage, cgDemoData(18), NoErase=j NE 0, Position=p
    cgColorBar, position=[p[0], p[3]+0.05, p[2], p[3]+0.1]
ENDFOR
cgText, 0.5, 0.925, /Normal, 'Example Image Layout', $
    Alignment=0.5, CharSize=cgDefCharSize()*1.25
END

```

If you wanted to draw two plots, side by side, but you wanted the plots to have a square aspect ratio:

```

cgDisplay, WID=2
pos = cgLayout([2,1], Aspect=1.0)
FOR j=0,1 DO cgPlot, cgDemoData(17), Position=pos[* ,j], NoErase=j NE 0
END

```

If you just want to draw the third plot in a 3x2 grid:

```

cgPlot, cgDemoData(17), Position=cgLayout([3,2,3])

```

I think this is flexible enough to meet everyone's needs.

Thanks to Matthew Argall, who gave me some great ideas and inspiration with his `Plot_Positions` program!

Cheers,

David

--

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