
Subject: Re: Multiple Plots in PostScript

Posted by [Paul Van Delst\[1\]](#) on Mon, 06 Jul 2009 21:34:35 GMT

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

David Fanning wrote:

```
> Eventually, I discovered that I had to set !Y.OMargin before
> *every page* of PostScript output, or it would not take effect.
>
> I have *no* idea why this would be the case. I'm only
> reporting what I have observed this morning. :-)
```

For stuff like this where I do it over and over and over again, I stick all the bits in an object along with the !p, !x, and !y sysvars. My philosophy has become one of: "If in doubt save, and then restore, EVERYTHING" :o)

The only reason I mention this here is that I was similarly surprised when the following worked (for both onscreen and PS output):

I've gotten into the habit of doing the following when I do p.multi plots with objects,

```
PRO myobj__Define
  void = { myobj, $
    ...other obj data....
    ; The following components are PRIVATE to the class
    ; ...Variables for plotting
    xsysvar : PTR_NEW(), $ ; X-axis system variable
    ysysvar : PTR_NEW(), $ ; Y-axis system variable
    psysvar : PTR_NEW() } ; Plotting system variable
END
```

and then have myobj::save_plotvars and myobj::restore_plotvars methods to save and/or restore them in the myobj::plot and myobj::oplot methods. E.g. in the Plot method for a particular object I'm working on now I do:

```
; Set plotting parameters
self->Get_Property, n_Bands=n_Bands, Channel=Channel
!P.MULTI = [0,n_Bands,1]
!X.OMARGIN = [6,0]

; Begin band response plots
FOR i = 0L, n_Bands-1L DO BEGIN
  self->Get_Property, $
  i, $
  Frequency = f, $
  Response = r
  PLOT, f, r, $
  TITLE='Ch.'+STRTRIM(Channel,2)+' ', band #'+STRTRIM(i+1,2), $
```

```

        XTITLE='Frequency', $
        XMARGIN=[2,3], $
        XRANGE=xrange,/XSTYLE, $
        CHARSIZE=charsize, $
        _EXTRA = Extra
    self->Save_PlotVars, i ; <--- SAVE AFTER PLOTTING
ENDFOR

```

and then if I want to overplot a different instance's data (with congruent dimensions of course, I check for that) I just do the following in my oplot method:

```

; Loop over bands
FOR i = 0L, n_Bands-1L DO BEGIN
    osrf->Get_Property, $
    i, $
    Frequency = f, $
    Response = r
; Plot it
    self->Restore_PlotVars, i ; <--- RESTORE BEFORE PLOTTING
    OPLOT, f, r, _EXTRA = Extra
ENDFOR

```

Not terribly scalable, I admit, but it works and no need for using OG that may log me out of my computer..... :o)

cheers,

paulv
