Subject: being clever with postscript
Posted by ianpaul.freeley on Thu, 08 May 2008 22:06:55 GMT
View Forum Message <> Reply to Message

OK, here's the situation I'm sick of: I've written a pile of code
that, after performing some amazing array manipulation, generates some
beautiful plots. Now I'd like to include the plots in a scientific
paper by creating encapsulated postscript files. The problem is, for
many of the plots I will want to convert them to black and white to
save printing costs. For those that I want to leave in color, I'll
need both CMYK and RGB versions (CMYK for the printers, and RBG to put
on-line).

Can anyone come up with a cleaver wrapper to put around my plot
commands so that IDL outputs B&W, RGB, and CMYK postscript files? It
would really be handy if I could open multiple devices to plot to
simultaneously, but I don't see any way to do that.

I could just write a loop around all my current "device" calls, but I
was hoping for something more elegant.

cheers,
IP Freeley

Subject: Re: being clever with postscript
Posted by ianpaul.freeley on Mon, 12 May 2008 22:18:35 GMT
View Forum Message <> Reply to Message

Here's the solution I came up with, I just wrote the stupid loops. It
ends up clobbering whatever the original color table was, I'm too lazy
to put in the common blocks to fix that. pyps is just my favorite
postscript opening procedure, then multi_ps sets up the correct
postscript files.

```
pro pyps, _extra=e, font=font, close=close, encapsulated=encapsulated

if not keyword_set(encapsulated) then encapsulated=0
;need to set this because IDL remembers /encap

if keyword_set(close) then begin
 device, /close
 set_plot, 'X'
 !x.thick=0 ;return to normal
 !y.thick=0
       !p.font=-1      ;return to vector fonts
       !p.charsize=0
```

```
        !p.thick=0
 return
endif else begin
        if !d.name eq 'PS' then print, 'WARNING!  Already set to
Postscript.  Did you just forget to close another device?'
 set_plot, 'PS'
        if keyword_set(font) then !p.font=font else !p.font=0 ;set to
                        ;hardware font

 !x.thick=4 ;thick lines are nice
 !y.thick=4
        !p.charsize=1.25  ;bigger font is nice
        !p.thick=2
 ;if not keyword_set(landscape) then landscape=1 ;landscape by default
 device, encapsulated=encapsulated, _extra=e, /times
 return
endelse



end

pro multi_ps, pn, filename=filename, _extra=e, hard_bw=hard_bw, $
            close=close

                        ;need to incorporate a way to restore
                        ;the original color table that gets
                        ;clobbered when /hard_bw set.

  ;generate regular rgb
  if pn eq 0 then pyps, filename=filename+'_rgb.eps', /encapsulated, $
  /color, bits=8, _extra=e

  ;generate cmyk
  if pn eq 1 then pyps, filename=filename+'_cmyk.eps', /encapsulated,
$
  /color, bits=8, /cmyk, _extra=e

  ;generate B&W
  if pn eq 2 then begin
    if keyword_set(hard_bw) then  $
    tvlct, [[fltarr(256)],[fltarr(256)],[fltarr(256)]]

    pyps, filename=filename+'_bw.eps', /encapsulated
  endif
```

```
end

;try to make some good code for making three output files


x=findgen(101)
y=15.+x^2-3.*x


;now to make some plots

loadct, 39

for pn=0,2 do begin ;pn=plot number
  multi_ps, pn, filename='test', /hard_bw
  plot, x,y,/nodata
  oplot, x,y, color=250, psym=2
  pyps, /close
endfor



end
```