

---

Subject: Re: Issue using map\_set and !p.multi in combination with polyfill to colour window backgrounds

Posted by [Andy Sayer](#) on Fri, 23 Aug 2013 14:40:08 GMT

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

---

... as often happens, posting to the group led to a flash of inspiration and I think I have figured out what I was doing wrong. Basically, if !p.multi is not zero and you're doing the first plot on the page, in addition to using /noerase you also need to not use /advance. So, this works:

```
nplots = !P.multi(1) * !P.multi(2)
advance = nplots gt 1
noerase=0

if !p.multi[0] eq 0 then begin
    noerase=1
    advance=0
endif

map_set,advance=advance,noerase=noerase
```

Hope this helps someone else (or me in 3 months when I forget)!

Andy

On Friday, August 23, 2013 10:12:56 AM UTC-4, AMS wrote:

> Sorry, it should read 7\*5=35 in the text above, not 25, obviously. :)

>

>

>

> On Friday, August 23, 2013 10:00:03 AM UTC-4, AMS wrote:

>

>> Hi all,

>

>>

>

>>

>

>>

>

>> I'm having a slight issue with a combination of polyfill, map\_set, and !P.multi...

>

>>

>

>>

>

>>

>

>> Basically, I would like to set up multiple maps per page with the option of having a different

colour background for each individual map (which will normally be black or white). I know I can change background colours with Coyote Graphics, but I only want to colour a portion of the window, not the whole thing, if I am drawing multiple plots to the window. So I have been writing some test code to check this out. This is what I have so far:

```
>
>>
>
>>
>
>>
>
>>
>
>>
>
>>
>
>> !P.multi=[0,7,5]
>
>>
>
>> loadct,2
>
>>
>
>>
>
>>
>
>> window,0,xsize=1500,ysize=800
>
>>
>
>> for i=0,!p.multi[1]*!p.multi[2]-1 do begin
>
>>
>
>>
>
>>
>
>> ; Get coordinates of this plot subregion for polyfill.
>
>>
>
>> ; Can't colour the whole window as it won't work with !p.multi.
>
>>
>
>> p_nx=!p.multi[1]
```

```

>
>>
>
>>     if p_nx eq 0 then p_nx=1
>
>>
>
>>     p_ny=!p.multi[2]
>
>>
>
>>     if p_ny eq 0 then p_ny=1
>
>>
>
>>     ; Get the x/y position of the current window
>
>>
>
>>     p_fx=!p.multi[0] mod p_nx
>
>>
>
>>     if p_fx eq 0 then p_fx = !p.multi[1]
>
>>
>
>>
>
>>     p_fy=fix((!p.multi[0]-1) / (1.*p_nx))
>
>>
>
>>     if !p.multi[0] eq 0 then p_fy= p_ny-1; !p.multi[0]=0 means this is the first plot on a new
page
>
>>
>
>>
>
>>
>
>>     print,!p.multi[0],p_fx,p_fy
>
>>
>

```

```

>>
>
>>
>
>> start_x=1.-(1.*p_fx)/!p.multi[1]
>
>>
>
>> end_x=start_x+1./!p.multi[1]
>
>>
>
>>
>
>> start_y=(1.*p_fy)/!p.multi[2]
>
>>
>
>> end_y=start_y+1./!p.multi[2]
>
>>
>
>>
>
>> print,start_x,end_x,start_y,end_y
>
>>
>
>>
>
>>
>
>> polyfill, [start_x,end_x,end_x,start_x,start_x],
[start_y,start_y,end_y,end_y,start_y],/normal,color=i+1
>
>>
>
>>
>
>>
>
>> nplots = !P.multi(1) * !P.multi(2)
>
>>

```

```

>
>>     advance = nplots gt 1
>
>>
>
>>     map_set,advance=advance
>
>>
>
>>
>
>>
>
>> endfor
>
>>
>
>>
>
>>
>
>> end
>
>>
>
>>
>
>>
>
>> The first section of the loop basically sets up the normalised x/y coordinates for this plot
among the set of (here 7*5=25) plots. These coordinates are then printed to the screen using the
print command, and all look correct. So, if I run this I'd expect to get 35 differently-coloured
plots (well, map_set borders), in different shades of 'greenness' according to colour table
2, going across the window in rows from top to bottom.
>
>>
>
>>
>
>>
>
>> However, what actually happens is that the first plot has a black background, rather than the
initial dark green. The other 34 look as expected. If I insert a stop before the map_set command, I
see that the dark green polygon is drawn but then erased by the first map_set call. No problem, I
thought, I'll check if this is the first plot on the page and if so use the noerase keyword:
>
>>
>

```

```

>>
>
>>
>
>>     noerase=0
>
>>
>
>>     if !p.multi[0] eq 0 then noerase=1
>
>>
>
>>     nplots = !P.multi(1) * !P.multi(2)
>
>>
>
>>     advance = nplots gt 1
>
>>
>
>>     map_set,advance=advance,noerase=noerase
>
>>
>
>>
>
>> However, this gives the same behaviour. If I abandon maps for a moment and replace that
section with this, it works fine:
>
>>
>
>>
>
>>
>
>>     noerase=0
>
>>
>
>>     if !p.multi[0] eq 0 then noerase=1
>
>>
>
>>     plot,findgen(4),noerase=noerase
>
>>

```

```
>
>> if !p.multi[0] eq 0 then !p.multi[0] = !p.multi[1]*!p.multi[2]-1
>
>>
>
>>
>
>>
>
>>
>
>> ... but I can't make it work with map_set at all. Can anyone see what I am doing wrong?
This is IDL 8.2.2 on CentOS (not sure which version).
>
>>
>
>>
>
>>
>
>> Thanks,
>
>>
>
>>
>
>>
>
>> Andy
```

---