
Subject: Re: !P.MULTI + POSITION keyword problem
Posted by [Craig Markwardt](#) on Tue, 02 Feb 1999 08:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

gpetty@rain.atms.purdue.edu (Grant W. Petty) writes:

>
> I just learned about the !P.MULTI variable and wanted to use it to put
> multiple panels on a single postscript page. The problem (I
> discovered after much head scratching) is that each of my panels use
> 'tv' and 'plot' calls which require the POSITION keyword for alignment
> (like David's 'imagebar' example in his book and for similar reasons).
> Unfortunately, the POSITION keyword causes IDL to disregard the
> contents of !P.MULTI (except for the first element). I had hoped that
> POSITION would be defined in terms of "virtual" Normalized Device
> Coordinates for the current panel indicated by !P.MULTI, but this is
> apparently not the case. Is there a simple way to get the desired
> effect without having to manually calculate new POSITION coordinates for
> each panel?
>

I plot images quite a bit, and have developed a routine called
PLOTIMAGE which is available on my web site
(<http://astrog.physics.wisc.edu/~craigm/idl/idl.html>). If you get it,
be sure to get PLOTIMAGE, TVIMAGE, CMCONGRID, and probably OPLOTIMAGE.
TVIMAGE is taken from David Fanning but slightly modified because
CONGRID does not produce correct output. PLOTIMAGE et al are
documented with extensive comments.

It's pretty easy to use. The following command will do the trick, as
long as the image is already byte-scaled:

```
IDL> PLOTIMAGE, img
```

Often you will want to assign coordinate axes to the image. Do that
with the IMGXRange and IMGYRange which define the extent of the image.
You can display a subrange of the image quite naturally then by
specifying the XRange and YRange like you always would.

PLOTIMAGE will respect the !p.multi settings, but these can be
overridden by the POSITION keyword.

You also mention virtual coordinates. PLOTIMAGE can use virtual
coordinates. It considers a "panel" to be the entire plotting region
including annotations, and the "subpanel" to be the actual plot data
window, specified in coordinates *relative* to the panel. Coordinates
are fed in the same way as for POSITION, but with the PANEL and
SUBPANEL keywords. I do have a program that tiles the page with
smaller panels if you'd like. It helps remove the cavernous spacing

between plots under !P.MULTI too.

Cheers,
Craig

--

Craig B. Markwardt, Ph.D. EMAIL: craigmnet@astrog.physics.wisc.edu
Astrophysics, IDL, Finance, Derivatives | Remove "net" for better response

Subject: Re: !P.MULTI + POSITION keyword problem
Posted by [Liam Gumley](#) on Tue, 02 Feb 1999 08:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

Grant W. Petty wrote:

> I just learned about the !P.MULTI variable and wanted to use it to put
> multiple panels on a single postscript page. The problem (I
> discovered after much head scratching) is that each of my panels use
> 'tv' and 'plot' calls which require the POSITION keyword for alignment
> (like David's 'imagebar' example in his book and for similar reasons).
> Unfortunately, the POSITION keyword causes IDL to disregard the
> contents of !P.MULTI (except for the first element). I had hoped that
> POSITION would be defined in terms of "virtual" Normalized Device
> Coordinates for the current panel indicated by !P.MULTI, but this is
> apparently not the case. Is there a simple way to get the desired
> effect without having to manually calculate new POSITION coordinates for
> each panel?

Try using PLOT to establish your position coordinates.
You'll need <http://www.dfanning.com/programs/tvimage.pro>

;---cut here---
PRO TEST

```
;- Set multiple plot options and make an image
!p.multi = [0,2,1,0,0]
image = bytscl( dist(32), top=!d.table_size-1 )
loadct, 0
```

```
;- Get plot position via PLOT
plot, [0], /nodata, xstyle=4, ystyle=4
position = [!x.window(0),!y.window(0),!x.window(1),!y.window(1)]
```

```
;- Display the image with title and axes
tvimage, image, /keep, position=position
plot, [0], /nodata, /noerase, position=position, title='Left Image'
```

```
;- Get next plot position
plot, [0], /nodata, xstyle=4, ystyle=4
position = [!x.window(0),!y.window(0),!x.window(1),!y.window(1)]
```

```
;- Display next image
tvimage, image, /keep, position=position
plot, [0], /nodata, /noerase, position=position, title='Right Image'
```

END

;---cut here---

Now when you turn on Postscript, everything should be positioned correctly, e.g.

```
set_plot, 'PS'
device, /color, bits=8, /land
test
device, /close ; now look at idl.ps
```

Cheers,
Liam.

Liam E. Gumley
Space Science and Engineering Center, UW-Madison
1225 W. Dayton St., Madison WI 53706, USA
Phone (608) 265-5358, Fax (608) 262-5974
<http://cimss.ssec.wisc.edu/~gumley>

Subject: Re: !P.MULTI + POSITION keyword problem

Posted by [davidf](#) on Tue, 02 Feb 1999 08:00:00 GMT

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

Grant W. Petty (gpetty@rain.atms.purdue.edu) writes:

```
> I just learned about the !P.MULTI variable and wanted to use it to put
> multiple panels on a single postscript page. The problem (I
> discovered after much head scratching) is that each of my panels use
> 'tv' and 'plot' calls which require the POSITION keyword for alignment
> (like David's 'imagebar' example in his book and for similar reasons).
> Unfortunately, the POSITION keyword causes IDL to disregard the
> contents of !P.MULTI (except for the first element). I had hoped that
> POSITION would be defined in terms of "virtual" Normalized Device
> Coordinates for the current panel indicated by !P.MULTI, but this is
> apparently not the case. Is there a simple way to get the desired
> effect without having to manually calculate new POSITION coordinates for
```

> each panel?

A "simple" way!? No, probably not. There was a fairly extensive discussion in this news group about !P.Multi and the TV command started by an article by Cathy (csaute3@alumni.umbc.edu) on 28 May 1998. The basic problem is that the TV command doesn't "advance" the !P.Multi variable, nor can you figure out where to put the image until *after* something has been plotted in the window.

Sometimes this is not a problem because you can write the program in such a way as to get a PLOT command (or something else) to fulfill this essential role.

I offered this simple program as an example that could position an image with axes around it in a window that is set up with !P.Multi.

The program works like this. First, set up !P.Multi.

```
IDL> !P.Multi=[0, 3, 2]
```

Then, pass this information to the program:

```
IDL> MultImages, !P.Multi
```

Works great in this limited capacity. :-)

Cheers,

David

```
***** **
```

```
PRO MultImages, multi
IF N_Params() NE 1 THEN multi = [0, 2, 2]
imageFile = Filepath(SubDir=['examples','data'], 'worldelv.dat')
image = BytArr(360, 360)
OpenR, lun, imageFile, /Get_LUN
ReadU, lun, image
Free_Lun, lun
Window, XSize=500, YSize=400
!P.Multi = multi
FOR j=0, multi[1]*multi[2]-1 DO BEGIN
    Plot, Findgen(11), Color=!P.Background
    x1 = !X.Region[0] + 0.05
    x2 = !X.Region[1] - 0.05
    y1 = !Y.Region[0] + 0.05
    y2 = !Y.Region[1] - 0.05
```

```
TVImage, image, Position=[x1, y1, x2, y2]
Plot, Findgen(11), position=[x1, y1, x2, y2], xticklen=-0.02, $
    yticklen=-0.02, xtitle='latitude', ytitle='longitude', $
    /nodata, /noerase
ENDFOR
END
*****
```

David Fanning, Ph.D.
Fanning Software Consulting
Phone: 970-221-0438 E-Mail: davidf@dfanning.com
Coyote's Guide to IDL Programming: <http://www.dfanning.com/>
Toll-Free IDL Book Orders: 1-888-461-0155

[Note: This follow-up was e-mailed to the cited author.]
