
Subject: Re: Reverse engineering the new graphics PLOT() margin property?

Posted by [Sean\[1\]](#) on Wed, 18 Apr 2012 21:15:40 GMT

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

On Monday, 16 April 2012 19:54:42 UTC-6, Mark Piper wrote:

> On Thursday, April 12, 2012 11:37:45 AM UTC-6, Sean Davis wrote:

>> I was wondering if anyone has figured out how the default margins are determined in new graphics?

>>

>> On a related note, it is infuriating that there is no way to return the values of INIT properties like margin, position, etc. (unless I'm missing something!)

>>

>> What I'd really like to be able to do is the following --

>>

>> p = plot(findgen(20))

>>

>> print, p.margin

>

> Hi Sean,

>

> I agree, we should be able to get these plot properties that we can set on init. I'll enter a bug report.

>

> In the meantime, I think this works (though I haven't tested it thoroughly):

>

> IDL> p = plot(findgen(10)^2)

> IDL> r = transpose([[p.xrange], [p.yrange]])

> IDL> m = p.convertcoord(r, /data, /to_normal)

> IDL> print, m

> 0.13000000 0.13000000 0.00000000

> 0.92000000 0.89000000 0.00000000

>

> Check:

>

> IDL> print, p.convertcoord(m, /normal, /to_data)

> 8.8817842e-016 0.00000000 0.00000000

> 10.000000 100.00000 0.00000000

>

> mp

Mark,

Yup, that's the solution I came up with, but there are two complications that are currently undocumented (and will hopefully be fixed in 8.2!!!)

1. For log, axes, data coordinates are returned as the log of the value

;make a plot with a logarithmic y axis

```
IDL> p = plot(findgen(10), yrange=[1e3,10.], ylog=1)
% Loaded DLM: XML.
;try to return the normal coordinates of the lower left corner
IDL> print, p.convertcoord(0,1e3, /data,/to_normal)
    0.13000000    -378.73000    0.00000000
;yikes, the y-value of the lower corner doesn't look right -- try feeding in the log10 of the lower
corner
IDL> print, p.convertcoord(0,alog10(1e3), /data,/to_normal)
    0.13000000    0.13000000    0.00000000
;that looks better!
```

2. For multiplots using the LAYOUT keyword, the margin values are not normal relative to the whole window, but just to the fraction of the window that is taken up by the given plot. This is kind of hard to explain, but here's an example:

```
IDL> p = plot(findgen(10), yrange=[1e3,10.], ylog=1, layout=[2,2,1],title='Margins not specified')
```

;Now get the lower left and upper right corners of the plot box

```
IDL> print, p.convertcoord(0,3, /data,/to_normal)
    0.085000000    0.57000000    0.00000000
IDL> print, p.convertcoord(10,1, /data,/to_normal)
    0.46000000    0.94500000    0.00000000
```

;Try to reproduce the first plot by feeding in the margins that are implied (assuming the layout keyword allots a box going from 0.-0.5 in x and 0.5-1.0 in y (normal coordinates)

```
IDL> p2 = plot(findgen(10), yrange=[1e3,10.], ylog=1, layout=[2,2,1],title='Margin =
[.085,.07,.04,.055]',margin=[.085,.07,.04,.055])
```

;This doesn't work! Why? Because apparently the margins are interpreted as being normal coordinates relative to the plot region allotted by the layout keyword!

;Instead, since we know that the 2 x 2 layout gives a plot region that is half the size of the full window in both x- and y-directions, we multiply the margin by 2

```
IDL> p3 = plot(findgen(10), yrange=[1e3,10.], ylog=1, layout=[2,2,1],title='Margin =
[.085,.07,.04,.055]*2',margin=[.085,.07,.04,.055]*2)
```

;Now we've created a plot that is the same as the original plot!

Sean
