
Subject: How to access properties of multiple plots in a function graphics window?

Posted by [Paul Van Delst\[1\]](#) on Tue, 12 Nov 2013 17:22:02 GMT

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

Hello,

I've got code something like the following in a procedure to plot barplots of some stats in a 2x2 grid. A legend for the datasets is created only for the first plot, i.e. b[0].

```
w = WINDOW()
b = OBJARR(n_plots)

FOR j = 0, n_plots-1 DO BEGIN
  index = 0
  FOR i = 0, n_datasets-1 DO BEGIN
    b[j] = BARPLOT( x, y[*],i,j], $
                  NBARS=n_datasets, $
                  INDEX=index++, $
                  NAME = dataset_name[i], $
                  OVERPLOT = i, $
                  LAYOUT = [2,2,j+1], $
                  /CURRENT )
    IF ( j EQ 0 ) THEN BEGIN
      IF ( i EQ 0 ) THEN $
        l = LEGEND( TARGET=b[j] ) $
      ELSE $
        l.Add, b[j]
    ENDIF
  ENDFOR
ENDFOR
```

Works great. And I return the WINDOW object reference, w, back to the caller.

What I want to do is access w and modify the properties of the various plots (e.g. font size, legend font size) via the "w" object reference.

But I can't find any info in the IDL help that will tell me what the object property references should be.

For instance, if I have access to the object array "b" I can change the font size of the second plot simply by doing:

```
b[1].font_size = 9
```

and colour of the title of the third plot by (from IDL documentation)

```
b[2].title.color = "blue"
```

But how do I do that via the WINDOW reference, "w"?

Each individual plot (the b[j]'s) has been added to the window graphic via the "CURRENT" keyword, but what's the syntax for the accessing those plots via the object reference?

I hope my description of what I want to do makes sense.

Thanks for any info.

cheers,

paulv

Subject: Re: How to access properties of multiple plots in a function graphics window?

Posted by [Matthew Argall](#) on Tue, 12 Nov 2013 18:53:10 GMT

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

You can try using the Select and GetSelect methods to pull the individual object references out of the window...

http://www.exelisvis.com/docs/GetSelect_Method.html

http://www.exelisvis.com/docs/Select_Method.html

Subject: Re: How to access properties of multiple plots in a function graphics window?

Posted by [Matthew Argall](#) on Tue, 12 Nov 2013 19:00:00 GMT

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

Oh, sorry. The "Select" method is only for graphics objects, not the window. As far as I know, you will have to pull the graphics objects out of the window before you can alter them.

You could use

```
myNG = win -> HitTest(0.5, 0.5, DIMENSION=[0.5, 0.5])
```

to select and return the reference of everything in the window.

At any rate, probably not what you were hoping for.

Subject: Re: How to access properties of multiple plots in a function graphics window?

Posted by [Paul Van Delst\[1\]](#) on Tue, 12 Nov 2013 19:19:55 GMT

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

Hi Matthew,

On 11/12/13 13:53, Matthew Argall wrote:

> You can try using the Select and GetSelect methods to pull the individual object references out of the window...

>

> http://www.exelisvis.com/docs/GetSelect_Method.html

> http://www.exelisvis.com/docs/Select_Method.html

I wanted to avoid having to actually select something but, in a pinch, this will work for me. Thanks! Very cool.

BTW, I tried the /ALL keyword to Select on the window but it didn't work:

```
IDL> w.select,/all
```

```
IDL> x=w.getselect()
```

```
IDL> help, x
```

```
X          UNDEFINED = !NULL
```

But, when I actually selected everything in the window with the mouse and then did

```
IDL> x=w.getselect()
```

I got:

```
IDL> help, x
```

```
X          OBJREF    = Array[28]
```

which does contain what I want, I just have to sort through it all:

```
IDL> for i=0,27 do help, x[i]
```

```
<Expression>  AXIS <702414>
```

```
<Expression>  AXIS <702415>
```

```
<Expression>  AXIS <702416>
```

```
<Expression>  AXIS <702417>
```

```
<Expression>  TEXT <702418>
```

```
<Expression>  BARPLOT <698847>
```

```
<Expression>  GRAPHIC <702419>
```

```
<Expression>  AXIS <702420>
```

```
<Expression>  AXIS <702421>
```

```
<Expression>  AXIS <702422>
```

```
<Expression>  AXIS <702423>
```

```
<Expression>  TEXT <702424>
```

<Expression> BARPLOT <696924>
<Expression> GRAPHIC <702425>
<Expression> AXIS <702426>
<Expression> AXIS <702427>
<Expression> AXIS <702428>
<Expression> AXIS <702429>
<Expression> TEXT <702430>
<Expression> BARPLOT <695001>
<Expression> GRAPHIC <702431>
<Expression> AXIS <702432>
<Expression> AXIS <702433>
<Expression> AXIS <702434>
<Expression> AXIS <702435>
<Expression> TEXT <702436>
<Expression> BARPLOT <693078>
<Expression> GRAPHIC <702437>

I do wish the "w.select,/all" worked, though.

Thanks again.

cheers,

paulv

Subject: Re: How to access properties of multiple plots in a function graphics window?

Posted by [Matthew Argall](#) on Tue, 12 Nov 2013 20:31:14 GMT

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

> BTW, I tried the /ALL keyword to Select on the window but it didn't work:

Yes, the online help is a little weird, too. The "Select" method is listed as a method of the "Plot" function, but not of the "Window" function. Despite this, the example clearly shows "plot.window.select()" not "plot.select()".

> When I actually selected everything in the window with the mouse

You can simulate this with the "HitTest" method, if you are going for something non-interactive. You still would have to sort, though...

<http://www.exelisvis.com/docs/HitTestMethod.html>

Subject: Re: How to access properties of multiple plots in a function graphics

window?

Posted by [Paul Van Delst\[1\]](#) on Tue, 12 Nov 2013 22:39:37 GMT

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

On 11/12/13 15:31, Matthew Argall wrote:

```
>> When I actually selected everything in the window with the mouse
>
> You can simulate this with the "HitTest" method, if you are going for
> something non-interactive. You still would have to sort, though...
>
> http://www.exelisvis.com/docs/HitTestMethod.html
```

Oh yeah - I saw that in one of your original posts and I didn't grok (HitTest is a strange name).

Excellent! Thanks again.

cheers,

paulv

Subject: Re: How to access properties of multiple plots in a function graphics window?

Posted by [MP](#) on Wed, 13 Nov 2013 17:03:46 GMT

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

On Tuesday, November 12, 2013 10:22:02 AM UTC-7, Paul van Delst wrote:

```
>
> What I want to do is access w and modify the properties of the various
> plots (e.g. font size, legend font size) via the "w" object reference.
>
```

Hi Paul,

You can access plot properties by NAME. For example, given:

```
w = window()
for i=0, 5 do begin

    !null = plot(/test, color=!color.(i), layout=[3,2,i+1], $
                current=w, name=string(i, format='(i2.2)'))
endfor
```

you can set the color of the fifth plot to red with:

```
w['04'].color = 'red'
```

mp

Subject: Re: How to access properties of multiple plots in a function graphics window?

Posted by [David Fanning](#) on Wed, 13 Nov 2013 17:31:37 GMT

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

Mark Piper writes:

```
>
> On Tuesday, November 12, 2013 10:22:02 AM UTC-7, Paul van Delst wrote:
>>
>> What I want to do is access w and modify the properties of the various
>> plots (e.g. font size, legend font size) via the "w" object reference.
>>
>
> Hi Paul,
>
> You can access plot properties by NAME. For example, given:
>
>   w = window()
>   for i=0, 5 do begin
>
>       !null = plot(/test, color=!color.(i), layout=[3,2,i+1], $
>               current=w, name=string(i, format='(i2.2)'))
>   endfor
>
> you can set the color of the fifth plot to red with:
>
>   w['04'].color = 'red'
```

This is pretty useful. Especially if you know the name of the plot (although unlikely in a widget program). I was curious just what was in a window object (I assumed it was a container), so I wrote a small program to give me back the contents of an object.

```
;-----
FUNCTION cgGetObjects, theObject, ISA=isa, COUNT=count
```

```
    On_Error, 2
```

```
    count = 0
    IF Obj_HasMethod(theObject, 'GET') THEN BEGIN
    contents = theObject -> Get(/ALL,COUNT=childCount)
    ENDIF ELSE RETURN, !NULL
```

```
; If there is nothing in the container, return !NULL.
```

```

IF childCount EQ 0 THEN BEGIN
    RETURN, !NULL
ENDIF

; Make this call recursively.
result = contents

FOR j=0,childCount-1 DO BEGIN
    thisResult = cgGetObjects(contents[j], COUNT=count, ISA=isa)
    IF count GT 0 THEN result = [result, thisResult]
ENDFOR

IF N_Elements(isa) NE 0 THEN BEGIN
    indices = Where(Obj_ISA(result, isa), count)
    IF count GT 0 THEN result = result[indices] ELSE result = !Null
ENDIF

count = N_Elements(result)
RETURN, result

END
;-----

```

Then, I found I could do this without knowing the name of anything.

```

win = window()
p1 = plot(cgdemodata(17), color='red', /current)
p2 = plot(cgdemodata(17), color='green', /overplot, /current)
objs = cgGetObjects(win, ISA='IDLGRPLOT')

```

Oddly, there are four plot objects in the window, not two. Maybe the other two are used for backing store? Not sure about that. Maybe Mark can help us understand this. But, in any case, there appear to be duplicates. Oddly, I can't do this:

```

objs[0].color = 'blue'
objs[2].color = 'orange'

```

But, I can do this:

```

objs[0] -> SetProperty, Color=cgColor('navy', /Triple, /Row)
objs[2] -> SetProperty, Color=cgColor('orange', /Triple, /Row)
win.refresh

```

For what it is worth. :-)

Cheers,

David

--

David Fanning, Ph.D.

Fanning Software Consulting, Inc.

Coyote's Guide to IDL Programming: <http://www.idlcoyote.com/>

Sepore ma de ni thue. ("Perhaps thou speakest truth.")

Subject: Re: How to access properties of multiple plots in a function graphics window?

Posted by [David Fanning](#) on Wed, 13 Nov 2013 21:20:31 GMT

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

David Fanning writes:

```
> Then, I found I could do this without knowing the name of anything.
>
> win = window()
> p1 = plot(cgdemodata(17), color='red', /current)
> p2 = plot(cgdemodata(17), color='green', /overplot, /current)
> objs = cgGetObjects(win, ISA='IDLGRPLOT')
>
> Oddly, there are four plot objects in the window, not two. Maybe the
> other two are used for backing store? Not sure about that. Maybe Mark
> can help us understand this. But, in any case, there appear to be
> duplicates. Oddly, I can't do this:
>
> objs[0].color = 'blue'
> objs[2].color = 'orange'
>
> But, I can do this:
>
> objs[0] -> SetProperty, Color=cgColor('navy', /Triple, /Row)
> objs[2] -> SetProperty, Color=cgColor('orange', /Triple, /Row)
> win.refresh
>
> For what it is worth. :-)
```

I think what I should have been looking for are IDLVisPlots:

```
objs = cgGetObjects(win, ISA='idlvisplot')
```

There are only two of those, and these are apparently the two plots you want to work with:

```
objs[0] -> SetProperty, Color=cgColor('navy', /Triple, /Row)
objs[1] -> SetProperty, Color=cgColor('orange', /Triple, /Row)
```


win.refresh

Cheers,

David

--

David Fanning, Ph.D.

Fanning Software Consulting, Inc.

Coyote's Guide to IDL Programming: <http://www.idlcoyote.com/>

Sepore ma de ni thue. ("Perhaps thou speakest truth.")

Subject: Re: How to access properties of multiple plots in a function graphics window?

Posted by [Paul Van Delst\[1\]](#) on Mon, 18 Nov 2013 14:18:45 GMT

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

Hi Mark,

On 11/13/13 12:03, Mark Piper wrote:

> On Tuesday, November 12, 2013 10:22:02 AM UTC-7, Paul van Delst wrote:

>>

>> What I want to do is access w and modify the properties of the various

>> plots (e.g. font size, legend font size) via the "w" object reference.

>>

>

> Hi Paul,

>

> You can access plot properties by NAME. For example, given:

>

> w = window()

> for i=0, 5 do begin

>

> !null = plot(/test, color=!color.(i), layout=[3,2,i+1], \$

> current=w, name=string(i, format='(i2.2)'))

> endfor

>

> you can set the color of the fifth plot to red with:

>

> w['04'].color = 'red'

Cool.

I only thought NAME was useful for legends.

Thanks.

paulv
