Subject: UNsharing an IDLgrImage
Posted by kagoldberg on Sat, 27 Jul 2013 23:07:50 GMT
View Forum Message <> Reply to Message

Here's a strange object graphics quirk, I hope someone can explain. An IDLgrImage object can share its data with another IDLgrImage object. But what happens when you want to quit sharing, and unlink the two? Unless I missed something, it seems that you cannot.

Here's an example where we (1) define an image and draw it. (2) share with another image, and draw that. (3) attempt to unshare, set new data to the image, and draw it (it works). (4) modify the original share image, and observe that it is still linked to the first, with the ability to change it. After each 'stop' use '.c' or '.continue' to proceed to the next step.

How does one break the connection and un-share without destroying the second IDLgrImage? Thanks!

EXAMPLE CODE

```
oWindow = IDLgrWindow(dimensions=[500,500])
oView   = IDLgrView(EYE=1000, COLOR=[50,100,50], DIMENSIONS=[500,500],
VIEWPLANE_RECT=[0,0,500,500])
oModel  = IDLgrModel()
oView.Add, oModel

oImage = IDLgrImage(bindgen(16,16), LOCATION=[0,0], DIMENSION=[256,256], /GREYSCALE)
oModel.Add, oImage

oWindow.Draw, oView & print, 'A, here we see the bindgen ramp'

stop

oIshare = IDLgrImage(bytscl(randomn(seed,16,16)))
oImage.SetProperty, SHARE_DATA=oIshare  ;--- see the shared image
oWindow.Draw, oView & print, 'B, now we see the shared randomness'

stop

oImage.SetProperty, SHARE_DATA=!null, DATA=bytscl(dist(16,16)) ;--- see the dist()
oWindow.Draw, oView & print, 'C, we replace the share with dist() data'

stop

oIshare.SetProperty, DATA=255b-bindgen(16,16)
oWindow.Draw, oView & print, 'D, change the oIshare and it STILL affects the oImage'

end
```

Subject: Re: UNsharing an IDLgrImage
Posted by David Fanning on Sat, 27 Jul 2013 23:51:42 GMT
View Forum Message <> Reply to Message

kagoldberg@lbl.gov writes:

> Here's a strange object graphics quirk, I hope someone can explain. An IDLgrImage object can share its data with another IDLgrImage object. But what happens when you want to quit sharing, and unlink the two? Unless I missed something, it seems that you cannot.
>
> Here's an example where we (1) define an image and draw it. (2) share with another image, and draw that. (3) attempt to unshare, set new data to the image, and draw it (it works). (4) modify the original share image, and observe that it is still linked to the first, with the ability to change it. After each 'stop' use '.c' or '.continue' to proceed to the next step.
>
> How does one break the connection and un-share without destroying the second IDLgrImage?

When we wrote the Catalyst Library we had the notion of sharing data, too. This simply meant that the "data" was a pointer in the object and that when the "data" was shared, the new object was set up with it's data pointer being the very same data pointer the first object used.

When you set the DATA=!Null, you are simply making the thing that the data pointer points to a Null. Just like when you get the DATA out of the object, you get the thing the data pointer points to. You *don't* get the pointer itself.

To "unshare" the data, you are going to have to get at the data pointer itself, not the thing it points to. But, getting at the data pointer inside the object goes against all tenets of object oriented programming practice. Perhaps you can see the problem here. :-)

I'm not saying it can't be done. IDL's object policies are easily circumvented, but as a programmer I am loath to do it for all kinds of reasons. I would have to be convinced, and I would be a very, very hard sell. But, you might be able to convince me to implement an UNSHARE keyword. :-)

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.")