
Subject: IDLgrWindow->Draw,/CREATE_INSTANCE
Posted by [Marc Schellens](#) on Fri, 26 Nov 1999 08:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hello,
As I follow this newsgroup not for so long, sorry if this
topic was already raised, but on my Solaris system, using IDL5.2,
the IDLgrWindow Draw method works not as suggested with the
/CREATE_INSTANCE or /DRAW_INSTANCE keyword.
It seems that the method just ignores them.
Has anybody made the same experience, or even better, found a way
to get them work?
thanks,
:-) marc

Subject: Re: IDLgrWindow->Draw,/CREATE_INSTANCE
Posted by [Ben Tupper](#) on Sun, 28 Nov 1999 08:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

In article <3840FFA7.DBCE1F71@hotmail.com>, Marc says...

I was able to make you routine work by setting the /Transparent keyword
to the dynamic View definition. So...

```
> v2=obj_new('IDLgrView',VIEWPLANE_RECT=[0,0,512,512])
```

becomes...

```
v2=obj_new('IDLgrView',VIEWPLANE_RECT=[0,0,512,512],/Transparent)
```

I admit, that I don't know what rotations David has identified in your code.
But I'm gonna study that code he's posted!

Ben

Ben Tupper
PemaquidRiver@tidewater.net

Subject: Re: IDLgrWindow->Draw,/CREATE_INSTANCE
Posted by [davidf](#) on Sun, 28 Nov 1999 08:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

Marc Schellens (m_schellens@hotmail.com) writes:

```
> They don't work means:
```

> When I draw some background with ->Draw,/CREATE_INSTANCE, nothing is
> drawn
> while ->Draw,/DRAW_INSTANCE behaves like no keyword is set (so it just
> clears the screen and draws)
> I added a little test program, which should draw a picture
> and overdraws this picture shifted a little bit to the upper right
> corner.
> For this program I figured out that it did not work on windows NT also.
> Please try, if this works on your machine.

Well, it doesn't work on my machine either, but this is not because IDL capabilities are lacking. Rather, I think it is because you are just a bit unclear about how this works. The example in the IDL documentation is not too bad as far as pseudo-code goes, although if you perform the rotations in the way they suggest you get some truly interesting--if unintentional--effects. :-)

There are several things wrong with your example code. But since I'm doing this for **my** purposes as well as yours, I decided to create my own example program, rather than fix yours.

I modified a Simple_Surface program I have to illustrate object instancing. There is a new Rotation menu item that gives you the opportunity to rotate the surface about the X, Y, or Z axis. The Simple_Surface_Rotate event handler contains the details. You can find the program, named Rotate_Surface, here:

ftp://ftp.dfanning.com/pub/dfanning/outgoing/misc/rotate_surface.pro

Compile the program and run it like this:

```
IDL> .Compile rotate_surface
IDL> Simple_Surface
```

You will notice that I have two models in the program, one holds the objects that do not change in the rotation and one holds the objects that do. The real key to performing this kind of instancing is to hide the appropriate model at the proper time and to be sure you see everything (by making the view transparent) when you are performing the operations. It's a bit tricky, but the code in the event handler illustrates how straightforward it is once you have the concepts clearly in mind.

Cheers,

David

--

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

Subject: Re: IDLgrWindow->Draw,/CREATE_INSTANCE

Posted by [Marc Schellens](#) on Sun, 28 Nov 1999 08:00:00 GMT

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

Ben Tupper wrote:

>

> In article <383E214F.11846F94@hotmail.com>, Marc says...

>>

>> Hello,

>> As I follow this newsgroup not for so long, sorry if this

>> topic was already raised, but on my Solaris system, using IDL5.2,

>> the IDLgrWindow Draw method works not as suggested with the

>> /CREATE_INSTANCE or /DRAW_INSTANCE keyword.

>> It seems that the method just ignores them.

>> Has anybody made the same experience, or even better, found a way

>> to get them work?

>> thanks,

>> :-) marc

>

> I have used these keywords extensively on a Solaris but I hadn't noticed a

> problem. How can you tell that they don't work as expected? (I know that

> sounds like a silly question... but maybe its not working for me either and I

> simply didn't know better!)

>

> I did bump into an oddity with the Solaris system I was operating... expose

> events were handled by the system regardless of the value set for the RETAIN

> keyword. I don't know if/how RSI resolved this issue... but it was placed in

> the bug reporting queue.

>

> Ben Tupper

> PemaquidRiver@tidewater.net

They don't work means:

When I draw some background with ->Draw,/CREATE_INSTANCE, nothing is drawn

while ->Draw,/DRAW_INSTANCE behaves like no keyword is set (so it just clears the screen and draws)

I added a little test program, which should draw a picture and overdraws this picture shifted a little bit to the upper right corner.
For this program I figured out that it did not work on windows NT also.
Please try, if this works on your machine.
thanks,
:-) marc

```
;;-----  
  
pro test  
  
w=obj_new('IDLgrWindow',DIMENSIONS=[512,512])  
  
v1=obj_new('IDLgrView',VIEWPLANE_RECT=[0,0,512,512])  
v2=obj_new('IDLgrView',VIEWPLANE_RECT=[0,0,512,512])  
  
m1=obj_new('IDLgrModel')  
v1->Add,m1  
m2=obj_new('IDLgrModel')  
v2->Add,m2  
  
;; use the path to your rsi directory here  
read_gif,'/usr/local/rsi/idl/examples/demo/demodata/sunvel.gif',image  
  
i1=obj_new('IDLgrImage',image)  
m1->Add,i1  
i2=obj_new('IDLgrImage',image,LOCATION=[10,10])  
m2->Add,i2  
  
;; also with next line commented in it did not work  
;; then at least it draws the instance, but nevertheless erase it when  
;; drawing with /DRAW_INSTANCE  
;;w->Draw,v1,CREATE_INSTANCE=0  
w->Draw,v1,CREATE_INSTANCE=1 ;; 2 did not work either  
  
for i=1,10 do begin  
  i2->SetProperty,LOCATION=[i*10,i*10]  
  w->Draw,v2,/DRAW_INSTANCE  
endfor  
  
end
```

Subject: Re: IDLgrWindow->Draw,/CREATE_INSTANCE
Posted by [Ben Tupper](#) on Mon, 29 Nov 1999 08:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

In article <81sl0n\$c7f@edrn.newsguy.com>, Ben says...

>

> I admit, that I don't know what rotations David has identified in your code.

> But I'm gonna study that code he's posted!

>

Oohh! Cool! The rotations were in David's code. Very spiffy stuff.

Ben Tupper

PemaquidRiver@tidewater.net

Subject: Re: IDLgrWindow->Draw,/CREATE_INSTANCE

Posted by [davidf](#) on Mon, 29 Nov 1999 08:00:00 GMT

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

Marc Schellens (m_schellens@hotmail.com) writes:

> But David, could you please specify what are the other things
> (except missing /TRANSPARENT) wrong in my example code?

Oh, sorry, Marc. I was a little confused, too. I think
I misunderstood what you were doing with Create_Instance
at first. I think the TRANSPARENT thing was the only thing
really missing.

> I include it at the bottom. Please check it out.
> Obviously the problem is that the IDLgrViews did not cover the whole
> window here. However, under NT it works fine, but under Solaris the old
> picture is not cleared before drawing it a little bit shifted.
> So at least an inconsistency. (IDL 5.2)

I was suspicious of the changing views too. I am not
surprised to find a bit of inconsistency in this part
of the code, I guess. I'm running NT, of course, so
I didn't see any problem.

Cheers,

David

--

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

Subject: Re: IDLgrWindow->Draw,/CREATE_INSTANCE
Posted by [Marc Schellens](#) on Mon, 29 Nov 1999 08:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

Oh, oh, seems that I was a litte bit rash here.
Well, with TRANSPARENT keyword it works fine for me also.

So thank you very much! I think I should come clear now.

But David, could you please specify what are the other things
(except missing /TRANSPARENT) wrong in my example code?
Well, the IDL example code uses the instancing as you in your example
code, but I found nowhere said, that it **must** be done this way (or at
least I haven't found it in the documentation), have I overseen
something?

After feeling ashamed about such an simple mistake I looked for the
original code, which made me believe for a long time, that the
DRAW_INSTANCE/CREATE_INSTANCE did not work.
(BTW: this code comes from an IDL training course held here some time
ago.)

I include it at the bottom. Please check it out.

Obviously the problem is that the IDLgrViews did not cover the whole
window here. However, under NT it works fine, but under Solaris the old
picture is not cleared before drawing it a little bit shifted.
So at least an inconsistency. (IDL 5.2)

thanks, :-) marc

```
pro obj_win
```

```
people_a=bytarr(192,192)
people_b=people_a
openr,lun,filepath('people.dat',subdirectory='examples/data' ),/get_lun
readu,lun,people_a
readu,lun,people_b
free_lun,lun
```

```
oview_a=obj_new('idlgrview',$
    viewplane_rect=[0,0,192,192],dimension=[192,192],$
    zclip=[1,-1],location=[0,0])
oview_b=obj_new('idlgrview',$
    viewplane_rect=[0,0,192,192],dimension=[192,192],$
    zclip=[1,-1],location=[50,50])
```

```
oview_b->SetProperty,Transpar=1
```

```
omodel_a=obj_new('idlgrmodel')
```

```
omodel_b=obj_new('idlgrmodel')
```

```
oimage_a=obj_new('idlgrimage',people_a)
```

```
oimage_b=obj_new('idlgrimage',people_b)
```

```
omodel_a->add,oimage_a
```

```
omodel_b->add,oimage_b
```

```
oview_a->add,omodel_a
```

```
oview_b->add,omodel_b
```

```
owindow=obj_new('idlgrwindow',dimensions=[400,400],retain=2)
```

```
; The Keywords Create_Instance and Draw_Instance in the Draw
```

```
; method of the IDLgrWindow.
```

```
owindow->draw,oview_a,/create_instance ;The unchanging part of a  
drawing.
```

```
for i=50,90,5 do begin
```

```
    oview_b->setproperty,location=[i,i]
```

```
    owindow->draw,oview_b,/draw_instance ;The changing part of a drawing.
```

```
endfor
```

```
end
```
