Subject: Re: cgColor and Widget_Draw Frame problem
Posted by David Fanning on Thu, 01 Nov 2012 14:23:27 GMT
View Forum Message <> Reply to Message

Helder writes:

> I've been using cgColor quite successfully for quite a while with this, but I started to get a strange problem.
> If I create a draw widget with a frame then cgColor throws an error.
> This is the code that generates the error:
>
> PRO TestcgColor
> wBase = widget_base()
> wDrawColor = WIDGET_DRAW(wBase, XSIZE=100, YSIZE=100, FRAME=1)
> WIDGET_CONTROL, wBase, /REALIZE
> print, !D.x_size, !D.y_size
> print, cgcolor('white')
> END
>
> And the error reads: "CGSNAPSHOT--> TVRD: Value of Area is out of allowed range."
>
> The reason is that the draw region that I generated is 100x100 pixel and the size of the window is given as 102x102 (1 pixel per side).
>
> Is there a way to overcome this problem?
> I think one can read the frame size from (widget_info(..., /geometry)).margin and then subtract this from !D.x_size and !D.y_size (actually 2*margin) when calculating the opposite pixel in cgColor().
>
> Does this make any sense or am I on the false track?

Sigh... If there is one thing I hate, it is widget geometry. :-(

Is this some kind of LINUX you are running on? What is the result of printing !D.x_vsize, and !D.y_vsize? What version of IDL is this? All of these things are going to matter.

All cgColor is trying to do is read the pixel "color" in the upper right-hand corner of the graphics window. You might think this would be a simple thing to do...

Cheers,

David

--
David Fanning, Ph.D.

Subject: Re: cgColor and Widget_Draw Frame problem
Posted by Helder Marchetto on Thu, 01 Nov 2012 20:12:45 GMT
View Forum Message <> Reply to Message

On Thursday, November 1, 2012 3:23:26 PM UTC+1, David Fanning wrote:
> Helder writes:
>
>
>
>> I've been using cgColor quite successfully for quite a while with this, but I started to get a strange problem.
>
>> If I create a draw widget with a frame then cgColor throws an error.
>
>> This is the code that generates the error:
>
>>
>
>> PRO TestcgColor
>
>> wBase = widget_base()
>
>> wDrawColor = WIDGET_DRAW(wBase, XSIZE=100, YSIZE=100, FRAME=1)
>
>> WIDGET_CONTROL, wBase, /REALIZE
>
>> print, !D.x_size, !D.y_size
>
>> print, cgcolor('white')
>
>> END
>
>>
>
>> And the error reads: "CGSNAPSHOT--> TVRD: Value of Area is out of allowed range."
>
>>
>
>> The reason is that the draw region that I generated is 100x100 pixel and the size of the window is given as 102x102 (1 pixel per side).
>
>>
>

>> Is there a way to overcome this problem?
>
>> I think one can read the frame size from (widget_info(..., /geometry)).margin and then subtract this from !D.x_size and !D.y_size (actually 2*margin) when calculating the opposite pixel in cgColor().
>
>>
>
>> Does this make any sense or am I on the false track?
>
>
>
> Sigh... If there is one thing I hate, it is widget geometry. :-(
>
>
>
> Is this some kind of LINUX you are running on? What is the
>
> result of printing !D.x_vsize, and !D.y_vsize? What version
>
> of IDL is this? All of these things are going to matter.
>
>
>
> All cgColor is trying to do is read the pixel "color" in the
>
> upper right-hand corner of the graphics window. You might think
>
> this would be a simple thing to do...
>
>
>
> Cheers,
>
>
>
> David
>
>
>
>
>
> --
>
> David Fanning, Ph.D.
>
> Fanning Software Consulting, Inc.
>

> Coyote's Guide to IDL Programming: http://www.dfanning.com/
>
> Sepore ma de ni thue. ("Perhaps thos speakest truth.")

Hi David,
the result of the
print, !D.x_size, !D.y_size
is
        102   102
and
print, !Version
{ x86_64 Win32 Windows Microsoft Windows 8.2 Apr 10 2012   64   64}

After 15 min of debugging, I found out something quite strange... I monitored the values !d.x_size and !d.y_size. Some when in cgSnapshot you use wSet to select the current window. At this point the size of the window is reduced from 102 to 100 pixel... This seems to overcome the error:

```
PRO TestcgColor
wBase       = widget_base()
wDrawColor  = WIDGET_DRAW(wBase, XSIZE=100, YSIZE=100, FRAME=1)
WIDGET_CONTROL, wBase, /REALIZE
print,'Before wSet:',!d.x_size, !d.y_size
ThisWindow = !d.Window
wSet, ThisWindow
print,'After wSet:',!d.x_size, !d.y_size
print, cgcolor('white')
END
```

I don't understand "why" this changes. My guess is that this is a bug, but then it must be quite old...
One way to come around this problem would be to use wSet before you get the values of the size of the image and call cgSnapShot in cgColor (line 410 in my version).

I hope this helps.

Cheers,
Helder

Subject: Re: cgColor and Widget_Draw Frame problem
Posted by DavidF[1] on Thu, 01 Nov 2012 20:45:45 GMT

View Forum Message <> Reply to Message

Helder writes:

> the result of the
>
> print, !D.x_size, !D.y_size

>
> is
>
>   102   102

What I really want to know is what is the result of *this* command:

 print, !D.x_Vsize, !D.y_Vsize

> print, !Version
>
> { x86_64 Win32 Windows Microsoft Windows 8.2 Apr 10 2012   64   64}

Most interesting, since this the version of IDL I ran your example
program in, and I didn't have any problem whatsoever. What version
of Windows are you running?

> After 15 min of debugging, I found out something quite strange... I monitored the values !d.x_size and !d.y_size. Some when in cgSnapshot you use wSet to select the current window. At this point the size of the window is reduced from 102 to 100 pixel... This seems to overcome the error:

> PRO TestcgColor
> wBase   = widget_base()
> wDrawColor = WIDGET_DRAW(wBase, XSIZE=100, YSIZE=100, FRAME=1)
> WIDGET_CONTROL, wBase, /REALIZE
>
> print,'Before wSet:',!d.x_size, !d.y_size
> ThisWindow = !d.Window
>
> wSet, ThisWindow
> print,'After wSet:',!d.x_size, !d.y_size
> print, cgcolor('white')
>
> END

> I don't understand "why" this changes. My guess is that this is a bug, but then it must be quite old...
>
> One way to come around this problem would be to use wSet before you get the values of the size of the image and call cgSnapShot in cgColor (line 410 in my version).

Well, I sorta hope people know which window they are drawing
in when the *call* cgColor (but, I realize this is asking too much
of most widget programmers), but I am still holding out hope for
a more elegant solution.

I'm very curious to know if !D.X_VSIZE is the same as !D.X_SIZE in
your special case.

Cheers,

David

Subject: Re: cgColor and Widget_Draw Frame problem
Posted by Helder Marchetto on Thu, 01 Nov 2012 21:31:43 GMT
View Forum Message <> Reply to Message

On Thursday, November 1, 2012 9:45:45 PM UTC+1, Coyote wrote:
> Helder writes:
>
>
>
>> the result of the
>
>>
>
>> print, !D.x_size, !D.y_size
>
>>
>
>> is
>
>>
>
>>   102   102
>
>
>
> What I really want to know is what is the result of *this* command:
>
>
>
> print, !D.x_Vsize, !D.y_Vsize
>
>
>
>> print, !Version
>
>>
>
>> { x86_64 Win32 Windows Microsoft Windows 8.2 Apr 10 2012   64   64}
>
>

>
> Most interesting, since this the version of IDL I ran your example
>
> program in, and I didn't have any problem whatsoever. What version
>
> of Windows are you running?
>
>
>
>> After 15 min of debugging, I found out something quite strange... I monitored the values !d.x_size and !d.y_size. Some when in cgSnapshot you use wSet to select the current window. At this point the size of the window is reduced from 102 to 100 pixel... This seems to overcome the error:
>
>
>
>> PRO TestcgColor
>
>> wBase = widget_base()
>
>> wDrawColor = WIDGET_DRAW(wBase, XSIZE=100, YSIZE=100, FRAME=1)
>
>> WIDGET_CONTROL, wBase, /REALIZE
>
>>
>
>> print,'Before wSet:',!d.x_size, !d.y_size
>
>> ThisWindow = !d.Window
>
>>
>
>> wSet, ThisWindow
>
>> print,'After wSet:',!d.x_size, !d.y_size
>
>> print, cgcolor('white')
>
>>
>
>> END
>
>
>
>
>
>
>> I don't understand "why" this changes. My guess is that this is a bug, but then it must be quite old...

>
>>
>
>> One way to come around this problem would be to use wSet before you get the values of the size of the image and call cgSnapShot in cgColor (line 410 in my version).
>
>
>
> Well, I sorta hope people know which window they are drawing
>
> in when the *call* cgColor (but, I realize this is asking too much
>
> of most widget programmers), but I am still holding out hope for
>
> a more elegant solution.
>
>
>
> I'm very curious to know if !D.X_VSIZE is the same as !D.X_SIZE in
>
> your special case.
>
>
>
> Cheers,
>
>
>
> David

Hi David,
sorry, I didn't see the "v" before.
I now tried this procedure:

```
PRO TestcgColor
wBase       = widget_base()
wDrawColor  = WIDGET_DRAW(wBase, XSIZE=100, YSIZE=100, FRAME=1)
WIDGET_CONTROL, wBase, /REALIZE
print,'Before wSet (without v):',!d.x_size, !d.y_size
print,'Before wSet (with v):',!d.x_vsize, !d.y_vsize
ThisWindow = !d.Window
wSet, ThisWindow
print,'After wSet (without v):',!d.x_size, !d.y_size
print,'After wSet (with v):',!d.x_vsize, !d.y_vsize
print, cgcolor('white')
END
```

Result:

Before wSet (without v): 102 102
Before wSet (with v): 102 102
After wSet (without v): 100 100
After wSet (with v): 100 100
 16777215

I'm running win 7 pro 64-bit.

If I comment out the wSet command, I get the error.

Ok, so I'm the only one having this error?

I always use some where a wSet in my programs. And this is the first time that this appears. I was writing very very simple widget to run some test I was drawing something right after the creation of the widget (/REALIZE) and then I used cgColor() for figuring out the color. I do agree that this is not the normal way a widget program runs, but as I said it was supposed to be an easy and quick widget tool...
As far as I'm concerned, I'm happy to call my window with wSet and then use cgColor, but I have something inside me wanting to understand where the error is originating (no, not in the way I wrote the widget code, but rather in the effect of wSet on !d.x_size and !d.y_size).

I also checked the geometry of the widget, but that does not change before and after using wSet.

Cheers,
Helder

Subject: Re: cgColor and Widget_Draw Frame problem
Posted by David Fanning on Thu, 01 Nov 2012 22:15:44 GMT
View Forum Message <> Reply to Message

Helder writes:

> I now tried this procedure:
>
> PRO TestcgColor
> wBase = widget_base()
> wDrawColor = WIDGET_DRAW(wBase, XSIZE=100, YSIZE=100, FRAME=1)
> WIDGET_CONTROL, wBase, /REALIZE
> print,'Before wSet (without v):',!d.x_size, !d.y_size
> print,'Before wSet (with v):',!d.x_vsize, !d.y_vsize
> ThisWindow = !d.Window
> wSet, ThisWindow
> print,'After wSet (without v):',!d.x_size, !d.y_size
> print,'After wSet (with v):',!d.x_vsize, !d.y_vsize
> print, cgcolor('white')
> END

>
> Result:
>
> Before wSet (without v): 102 102
> Before wSet (with v): 102 102
> After wSet (without v): 100 100
> After wSet (with v): 100 100
> 16777215
>
> I'm running win 7 pro 64-bit.
>
> If I comment out the wSet command, I get the error.
>
> Ok, so I'm the only one having this error?

No, I see it now, too. Must not have had my coffee yet this morning. :-)

> I always use some where a wSet in my programs. And this is the first time that this appears. I was writing very very simple widget to run some test I was drawing something right after the creation of the widget (/REALIZE) and then I used cgColor() for figuring out the color. I do agree that this is not the normal way a widget program runs, but as I said it was supposed to be an easy and quick widget tool...
> As far as I'm concerned, I'm happy to call my window with wSet and then use cgColor, but I have something inside me wanting to understand where the error is originating (no, not in the way I wrote the widget code, but rather in the effect of wSet on !d.x_size and !d.y_size).
>
> I also checked the geometry of the widget, but that does not change before and after using wSet.

Well, it's weird, but it's not the weirdest thing I've ever seen when working with widgets. Spend some time trying to get the largest possible window on your display in a device-independent way, if you REALLY want to have some fun with window sizes:

http://www.idlcoyote.com/code_tips/goldilocks.html

Cheers,

David
--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.dfanning.com/
Sepore ma de ni thue. ("Perhaps thos speakest truth.")