Subject: Re: Stupid question regarding BUFFER keyword in NG output.....
Posted by Michael Galloy on Tue, 07 Dec 2010 01:00:08 GMT
View Forum Message <> Reply to Message

On 12/6/10 2:37 PM, Paul van Delst wrote:
> Hello,
>
> I just created a bunch of plots like so:
>
> p = PLOT(f, r, $
> TITLE=sensor_id + channel_string + ' threshold cutoff discrepancy', $
> XTITLE='Frequency (cm!U-1!N)', $
> YTITLE='Relative Response', $
> SYMBOL='diamond', $
> BUFFER=Buffer)
>
> I returned all the created plot references (lots of 'em in a list of lists in a hash) so I can modify and output if
> necessary. And they appear to contain actual plots:
>
> IDL> p=((tref[5])[23])[0]
> IDL> help, p
> P PLOT<323024>
>
> But, and here is the stupid question, how do I now display these buffer-ised plots?
>
> Thanks for any info.
>
> cheers,
>
> paulv

I believe plots made in a buffer are meant to produce some type of
output and then be destroyed, i.e., like

```
IDL> p = plot(/test, /buffer)
IDL> p->save, 'test.png'
IDL> obj_destroy, p
```

Mike
--
www.michaelgalloy.com
Research Mathematician
Tech-X Corporation

Subject: Re: Stupid question regarding BUFFER keyword in NG output.....

Posted by Paul Van Delst[1] on Tue, 07 Dec 2010 14:38:55 GMT
View Forum Message <> Reply to Message

Hi Mike,

Michael Galloy wrote:
>
> I believe plots made in a buffer are meant to produce some type of
> output and then be destroyed, i.e., like
>
> IDL> p = plot(/test, /buffer)
> IDL> p->save, 'test.png'
> IDL> obj_destroy, p

Nutz! I was hoping there was some sort of undocumented method that displayed the plots, e.g.
p.display
(which I tried by the way :o) Sort of like doing DG to a pixmap and then sticking it in a window.

cheers,

paulv

Subject: Re: Stupid question regarding BUFFER keyword in NG output.....
Posted by Michael Galloy on Tue, 07 Dec 2010 17:00:47 GMT
View Forum Message <> Reply to Message

On 12/7/10 7:38 am, Paul van Delst wrote:
> Hi Mike,
>
> Michael Galloy wrote:
>>
>> I believe plots made in a buffer are meant to produce some type of
>> output and then be destroyed, i.e., like
>>
>> IDL> p = plot(/test, /buffer)
>> IDL> p->save, 'test.png'
>> IDL> obj_destroy, p
>
> Nutz! I was hoping there was some sort of undocumented method that displayed the plots, e.g.
> p.display
> (which I tried by the way :o) Sort of like doing DG to a pixmap and then sticking it in a window.

Not that I know of, but that doesn't mean it's not in there somewhere.

Mike
--
www.michaelgalloy.com

Research Mathematician
Tech-X Corporation

Subject: Re: Stupid question regarding BUFFER keyword in NG output.....
Posted by chris_torrence@NOSPAM on Tue, 07 Dec 2010 23:57:23 GMT
View Forum Message <> Reply to Message

Hi Paul,

Well, you have a couple of different options. You could use the
Refresh method to disable drawing to the window. It will still create
the window, but you won't see any changes to the plot until you call
Refresh again.

Another crazy way is to use the CopyWindow method to get a screen
dump, and then throw it into direct graphics:

p = plot(/test,/buffer)
tv,p.copywindow(),/true

Here's another one for fun:
p = plot(/test,/buffer)
for i=0,100 do begin & p.color = !color.(i) & tv,p.copywindow(),/true

Hope this helps.
Cheers,
Chris
ITTVIS

Subject: Re: Stupid question regarding BUFFER keyword in NG output.....
Posted by lecacheux.alain on Wed, 08 Dec 2010 10:43:21 GMT
View Forum Message <> Reply to Message

On 8 déc, 00:57, Chris Torrence <gorth...@gmail.com> wrote:
> Hi Paul,
>
> Well, you have a couple of different options. You could use the
> Refresh method to disable drawing to the window. It will still create
> the window, but you won't see any changes to the plot until you call
> Refresh again.
>
> Another crazy way is to use the CopyWindow method to get a screen
> dump, and then throw it into direct graphics:
>
> p = plot(/test,/buffer)

> tv,p.copywindow(),/true
>
> Here's another one for fun:
> p = plot(/test,/buffer)
> for i=0,100 do begin & p.color = !color.(i) & tv,p.copywindow(),/true
>
> Hope this helps.
> Cheers,
> Chris
> ITTVIS

But if I am doing :

p = plot(/test)
p2 = plot(/test,/buffer)
q = image(p2.copyWindow())

the plot referenced by q is different (looks like undersampled) from
the plot referenced by p.
Why ?
alx.

Subject: Re: Stupid question regarding BUFFER keyword in NG output.....
Posted by Paul Van Delst[1] on Wed, 08 Dec 2010 14:54:40 GMT
View Forum Message <> Reply to Message

Hi Chris,

Chris Torrence wrote:
> Hi Paul,
>
> Well, you have a couple of different options. You could use the
> Refresh method to disable drawing to the window. It will still create
> the window, but you won't see any changes to the plot until you call
> Refresh again.

That's how I did it at first[*], but since I am creating, let's see, 5x32+16x16=416 plots I don't want to create 416
windows on my desktop. I figured I would create them under the hood, return the references in a hash, and then pick and
choose which one to display/modify based on other info from the processing.

> Another crazy way is to use the CopyWindow method to get a screen
> dump, and then throw it into direct graphics:
>
> p = plot(/test,/buffer)
> tv,p.copywindow(),/true

>
> Here's another one for fun:
> p = plot(/test,/buffer)
> for i=0,100 do begin & p.color = !color.(i) & tv,p.copywindow(),/true

While that is a pretty cool technique, I would need to have "regular" access to the plots I choose to display (e.g.
annotate them, add legends, zoom into regions, save them to file etc).

What are the odds for a method to display buffer-ised graphics in a future IDL release?

cheers,

paulv

[*] See a previous post of mine where I was turned onto this technique (using refresh) after I whinged about the
slowness of NG graphics using the regular overplot technique shown in the plot example docs. And the difference in speed
is like night and day. The last two days I've been creating all manner of plots using NG and using them to easily
inspect data, annotate plots, and save them for inclusion in documents. I haven't used DG graphics for a few weeks now.

Oh, and I still think better PS/EPS output support for NG graphics is crucial in future IDL releases -- I have been
playing with the PNG or PDF output options in NG and they just do not cut it quality-wise when the plot has to look
good both on-screen (at, say 250% zoom) and in a printout. For publication-quality (including being scaleable)
graphics, I still think DG PS output trumps NG (any method).
