Subject: plot and contour inconsistency
Posted by Matthew Argall on Sun, 10 Mar 2013 06:10:20 GMT
View Forum Message <> Reply to Message

I have a program that uses the system variable !values.f_nan as the default MIN_VALUE and MAX_VALUE for a plotting routine and it works fine.

However, I tried to do the same thing with contour and it does not work -- none of the contours are drawn.

```
x = findgen(256)
y = findgen(256)
c = dist(256)

plot, x, y, MIN_VALUE=!values.f_nan, MAX_VALUE=!value.f_nan

contour, c, MIN_VALUE=!values.f_nan, MAX_VALUE=!value.f_nan
```

Should they behave differently?

Subject: Re: plot and contour inconsistency
Posted by David Fanning on Sat, 20 Apr 2013 03:29:58 GMT
View Forum Message <> Reply to Message

Matthew Argall writes:

> This is quite great! I had been doing something similar, but the /ALLOCATE_HEAP keyword eluded me...
>
> I was wondering, in the cgZPlot::DrawPlot method, is there a reason you call cgGraphicsKeywords::GetProperty? Why can't you get all of the properties by doing simply, e.g., *self.max_value, since cgGraphicsKeywords is inherited?

Mostly because I already had this code written for the GetProperty method and I didn't want to write it again. Laziness, I guess I'm saying. :-)

> Also, why do you always draw to the pixmap then copy to the display?

Copying from a pixmap is 100x faster than drawing the graphic, so this is a standard buffering technique to prevent "flicker", as you quickly draw and erase things like the rubberband box in the graphics window. Flicker can be especially apparent as the graphics window gets larger. It is used here mostly for the panning operations, which would flicker a lot if the pixmap wasn't used.

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.")
