
Subject: Re: Save 2D conversion matrix

Posted by [Paolo Grigis](#) on Mon, 07 Nov 2005 18:02:21 GMT

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

David Fanning wrote:

> Peter Albert writes:

>

>

>> I am currently working on the display of climate datasets. One window
>> shows a map with e.g. some monthly mean values. My plan now is to let
>> the user pick a location using CURSOR, and then to display a time
>> series at the chosen point in a second window. O.k., I can get the
>> latitude / longitude of the chosen point using CONVERT_COORD, then I
>> can pick the appropriate time series and plot it in the second window.
>> However, if I am now going back to the map window, the correlation
>> between device and data coordinates is of course gone. If it was 3D
>> data, I would use T3D, but this is plain 2D data and don't see the
>> appropriate feature in the documentation. Any help which would save me
>> from re-drawing this map over and over again would be appreciated.

>>

>> N.b. this is all done using direct graphics. Is this finally the reason
>> to go ahead and read the manual about OO graphics?

>

>

> Object graphics is certainly overkill, but I would suggest
> you learn a little widget programming (seriously, the CURSOR
> command!? Sigh...).

>

> Benjamin's suggestion to save and restore the system variables
> is a good one, but it has always struck me as inelegant. Which
> variables, after all, did you really need to restore the
> coordinate system? It surely couldn't be ALL of them!

>

> So I spent a couple of days experimenting until I found out which
> ones I needed. (I was actually trying to find out which ones
> had to be set so I could establish a data coordinate system without
> going to the trouble of actually drawing a plot.) It turns out
> that these four pieces of information are needed to establish
> the data coordinate system: !X.S, !Y.S, !X.Window, and !Y.Window.
> Plus, the current window has to be the one your plot is in. (Not
> a given, always, in widget programming unless you explicitly
> make it so.)

>

> I found out that I could easily establish a data coordinate
> system for *any* window if I knew the X and Y range of the
> coordinate system and the location of the coordinate system
> in the window (i.e., its position). (I simply constructed the
> scaling parameters normally found in !X.S and !Y.S from the

> range and position.)

In order to achieve the same purpose (that is, setting a coordinate system in a fresh window), I was using this kind of statements:

```
plot,[0,0],xrange=[3,5],yrange=[7,9],xstyle=8+4+1,ystyle=8+4+1,xmargin=[0,0],ymargin=[0,0],/nodata
```

I don't know if I should feel ashamed for not using the elegant way with !XY.S & !XY.Window or proud for having found a viable solution to the problem in a few minutes instead of "a couple of days experimenting" ;-)

Ciao,
Paolo

>
> So, even if you still save *everything*, it feels good
> to know what it is in that mess you really need. :-)
>
> Cheers,
>
> David
>
