Subject: Re: iTools colorbar
Posted by sdettrick@gmail.com on Mon, 05 May 2008 08:53:15 GMT
View Forum Message <> Reply to Message

Kenneth P. Bowman wrote:
> I am making an iTools plot and need to add a colorbar to an existing iTool.
>
> I am doing it this way:
>
> IPLOT, [0.0, 1.0], [0.0, 1.0], [0.0, 1.0], OVERPLOT = itool_id, /HIDE, $
>   /INSERT_COLORBAR, RGB_TABLE = rgb_table
> colorbar_id = itool_obj -> FindIdentifiers('*COLORBAR', /ANNOTATION)
> colorbar_obj = itool_obj -> GetByIdentifier(colorbar_id[0])
> colorbar_obj -> SetProperty, ORIENTATION = 1, $
>   BORDER   = 1, $
>   TITLE   = 'Ozone (ppbv)', $
>   MAJOR   = 5, $
>   MINOR   = 1
> itool_obj -> RefreshCurrentWindow
>
>
> The color bar appears, and I can successfully change some
> properties (orientation, number of ticks, add a border), but
> am unable to change others (title, tick labels, size and
> position of colorbar).
>
> If I use the DIMENSIONS keyword, it changes the size of the
> colorbar image, but not the border and annotation.
>
> I could do it interactively, but that means that no two plots would
> ever have the same color bar in the same place. Plus it gets old
> after a few hundred plots. :-(
>
> I have looked at the IDLgrColorbar properties, and the properties
> that can be edited interactively in the iTool, but I am stumped.
>
> Can anyone tell me how to change a colorbar's title, labels,
> position, and size?
>
> Ken Bowman

I also needed to do a few hundred plots with the same colorbar (for a
movie) but found that it was enough to arrange the first frame of
plots using the mouse, and then create each subsequent plot by simply
using the SetData methods of the dataset for each iplot. This worked
all in one IDL session, but I suspect it is also possible to do it
across multiple IDL sessions using save/restore of the iTool

objects...

If that doesn't pan out, then to do things fully programmatically, I would suggest that you create your plot, then set a breakpoint in every iTools colorbar method, then modify the colorbar properties/position via the mouse. Navigating up and down the stack when the breakpoints are triggered should give you the info you need to do the same thing programmatically. This MO has so far worked for me.

By the way there is a bug in vertical colorbar plots which can be fixed by hand - in idlitviscolorbar__define.pro, change the line:

    data = currentorientation ? BINDGEN(2,256) : BINDGEN(256,2)

to:

    data = currentorientation ? TRANSPOSE(BINDGEN(256,2)) :
BINDGEN(256,2)

(Thanks to James Jones, ITT engineer, for that fix).

Cheers,
Sean