Subject: Re: Plot Clipping Problems
Posted by Chris Lee on Fri, 29 Apr 2005 09:09:34 GMT
View Forum Message <> Reply to Message

In article <d4rt21mkb1@peabody.colorado.edu>, "Sean Davis"
<seand@colorado.edu> wrote:

> Hello Folks,
> I recently had a problem with clipping and the OPLOT command in IDL.
> The problem is basically this... Normally, IDL clips data to the current
> plotting window, for example
> plot, findgen(20)
> oplot, findgen(20)-5, findgen(20)
> The freak occurence seemed to be associated
> with a previous call to write_png using the keyword /true in the TVRD
> function, (ie write_png, filename, tvrd(/true)) but I am not able to
> reproduce the error. I have searched D. Fanning's site and this
> newsgroup and can't find any mention of similar problems... Anyone seen
> stuff like this before?!?

!p.noclip is set to 1 maybe? the CLIP keyword is an array representing
the clipping rectangle, not a logical switch to enable clipping, if you
want to enable clipping, you set the keyword NOCLIP=0

> On other notes, *as usual* I find the IDL
> documentation sorely lacking, and also in this case, DEAD WRONG!!!
> Here's a quote from the IDL documentation, talking about the CLIP
> Graphics Keyword: "Coordinates are specified in data units unless an
> overriding coordinate unit specification keyword is present (i.e.,
> NORMAL or DEVICE). If the clipping is provided in data or normalized
> units, the actual clipping rectangle is computed by converting those
> values to device units. The clipping itself always occurs in device
> space. " This is plain wrong! !P.CLIP is defined *device* units, not
> *data* units, as they state.
>

IDL does everything plot related in device coordinates. The documentation
is telling *you* what to provide the IDL commands with. By default,
everything is in DATA coordinates, unless the NORMAL or DEVICE keywords
are present. In DATA or NORMAL space, the coordinates *you* give IDL are
converted to DEVICE before being used, or stored in !p.CLIP et al.

If IDL is doing anything wrong here, it's exposing a bit too much of the
internal API to the user.

Chris.

> Cheers,
> Sean
>