Subject: Re: Preserving coordinate transformation
Posted by David Fanning on Tue, 17 Oct 2006 14:14:20 GMT
View Forum Message <> Reply to Message

Sverre Solberg writes:

> And thanks for the help, I wasn't aware that wset reestablished !D.

I have the advantage of having recently spent several glorious days in the thrall of coordinate conversion mechanisms. :-)

I did learn something else of interest in the three excruciating days I spent considering all of this. I was working with a window that had several images in it. Each image has associated with it its own data coordinate system. My draw widget is able to "select" which image I clicked in, and tells that image to establish its data coordinate system.

Now, I usually need to convert the window location (event.x and event.y) into the data coordinate system of the image. No problem with CONVERT_COORD, in the manor I outlined previously:

```
d = Convert_Coord(event.x, event.y, /Device, /To_Data)
```

BUT, on occasion I need to convert a *distance* in device coordinates into a distance in data coordinates. For example, if I want to draw a station plot at the location where I clicked, but I want to add a station name just *above* the location, I may want to move the text 10 pixels toward the top of the image.

Naively, I was doing the conversion like this:

```
d = Convert_Coord(10, 10, /Device, /To_Data)
```

Then adding the "length" to the y data coordinate:

```
y_data = y_data + d[1]
```

This worked great with some images, and not at all with others! It turns out it worked great when I had one image in the window and the image coordinate system overlapped more or less with the device coordinate system. (Or, when the image I selected was near the origin of the device coordinate system.) But if an

image was far away from the origin of the device coordinate system my text could be WAY off the mark.

I realized, after much futzing around and thinking through an unhappy dinner with my wife ("Are you paying attention to what I'm talking about!?") that this was the wrong way to calculate a distance or length in my data coordinate system.

What I needed to know, was how many data coordinate "units" were in 10 "units" of device coordinates. This depended on my data coordinate range and is something CONVERT_COORD cannot compute for me. In fact, I realized I had to do the conversion myself, like this:

```
d = Abs((yrange[1] - yrange[0])) / !D.Y_Size * 10
y_data = y_data + d
```

Clearing up this little misunderstanding solved a LOT of problems for me. :-)

Cheers,

David
--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.dfanning.com/
Sepore ma de ni thui. ("Perhaps thou speakest truth.")