Subject: Re: trying to convert map to graphic functions
Posted by Phillip Bitzer on Wed, 05 Oct 2016 14:14:39 GMT
View Forum Message <> Reply to Message

On Friday, June 10, 2016 at 8:38:56 AM UTC-5, Brian McNoldy wrote:
> I still don't use the "new" graphics system much, just because the old way is so entrenched in my memory. But I do really like the new system when I do try it (and can get it to do what I want).
> I have a plot that I'd love to switch over, but have had a rather hard time getting the same behavior.
>

So, I'm a little late with this, but this is exactly the sort of thing that is absolutely necessary - examples of how to bridge the gap between pre-FG and post-FG.

I've been using FG quite a bit, and mostly like them. The biggest adjustment?The way you think about how to make plots is different in DG/CG and FG.

Here's the code to get something close to your plot. I've generated some dummy data. I also tend to be quite verbose in my code. I separated a lot of the setting of properties so we can see exactly what each one does.

```
;first, generate the dummy data
lat = INTERPOL([22.1, 22.4], 100)
lon = INTERPOL([-74.25, -74.1], 100)
press = INTERPOL([1000, 100], 100)

;generate the map:
mapLimits = [MIN(lat)-2,MIN(lon)-2,MAX(lat)+2,MAX(lon)+2]

m = MAP('Cylindrical Equal Area', LIMIT=mapLimits)

;set some properties, including the grid
m.mapgrid.LINESTYLE = 'dotted'

m.mapgrid.LABEL_ANGLE = 0
m.mapgrid.LABEL_POSITION = 0

;let's change the default formatting
;Look ma! Lambda function!
m.mapgrid.LABEL_FORMAT=LAMBDA(orientation, location, fractional, defaultLabel :
STRING(location, FORMAT='(F7.2)'))

m.mapgrid.box_axes = 1
m.mapgrid.GRID_LONGITUDE = 0.5 ;similar to londel in direct graphics
m.mapgrid.GRID_LATITUDE = 0.5 ;similar to londel in direct graphics

;add some continents
mCont = MAPCONTINENTS(/HIRES)
```

```
;now, let's add the data, colored by pressure

;first, byte scale the pressure to 0->255
ind = REVERSE(BYTSCL(press)) ;reverse to make the smallest pressures correspond to higher
indices (reds)

p = PLOT(lon, lat, $
  RGB_TABLE=39, VERT_COLORS=ind, $ ;set the colors
  SYMBOL=3, THICK=4, /OVERPLOT)

;plot some wind barbs

;generate some random locations
nStations = 10
wLat = RANDOMU(1L, nStations) * (mapLimits[2]-mapLimits[0]) + mapLimits[0]
wLon = RANDOMU(2L, nStations) * (mapLimits[3]-mapLimits[1]) + mapLimits[1]

;some random speeds and directions, too
wSpd = RANDOMU(3L, nStations) * 40.
wDir = RANDOMU(4l, nStations) * 360.

;get the horizontal/vertical components of the vector
uVec = wSpd * COS(wDir * !DTOR)
vVec = wSpd * SIN(wDir * !DTOR)

;plot the vectors
wb = VECTOR(uVec, vVec, wLon, wLat, $
  VECTOR_STYLE=1, COLOR='blue', /OVERPLOT)
```