Liam Gumley wrote:
> Ken Mankoff wrote:
>
>
>
> I'd love to see a simple example of how the MAP_PROJ routines may be
> used with direct graphics. The object graphics example in the IDL
> documentation is beyond me...
>
> For example, how do do you define a Lambert Azimuthal Equal Area
> Projection in direct graphics centered at a given lat/lon with a
> specified resolution, such as 1000 meters per pixel?
>

Hi Liam,

I have tried to take up this challenge with limited success.   There is one
place in the documentation that gives a bit of a hint (see MAP_CONTINENTS near
the bottom.) That, with the shapefile demo in the documents, was just enough
fpr me to pull together part of the solution. The code below shows how to
generate a projected map with the State o' Maine filled.   It ain't pretty, I
agree, but it's close.

Tried to ...
 center map a specified location   (= done)
 use specified projection (= done)
 use specified resolution (= not done)

Note that ...
 Map axes are in !MAP.UV_BOX coords not in LonLat.   I guess a function has to
be written to show the axes in lonlats using MAP_PROJ_FORWARD as they would be
shown using MAP_GRID, /BOX_AXES etc.   I don't think that would be too hard
(fingers crossed.)   I think the hard parts are figuring out the specified
resolution and a nice window shape for ISOTROPIC projections. I wonder if
MAP_SET could be modified to accept a MAP_STRUCTURE input keyword?

Here 'tis...

PRO mapTest

```
; GCTP Polar stereographic projection
myMap = MAP_PROJ_INIT(4, $
 LIMIT = [20, -150, 60, -60],$
```

```
      CENTER_LONGITUDE = -120, $
 CENTER_LATITUDE=45)

; Create a plot window using the UV Cartesian range.
PLOT, myMap.uv_box[[0,2]],myMap.uv_box[[1,3]], $
   /NODATA, /ISOTROPIC, XSTYLE=1, YSTYLE=1

MAP_CONTINENTS, MAP_STRUCTURE=myMap , /USA
MAP_GRID, MAP_STRUCTURE=myMap


;Open the states Shapefile in the examples directory
myshape=OBJ_NEW('IDLffShape', FILEPATH('states.shp', $
   SUBDIR=['examples', 'data']))

;Get the number of entities so we can parse through them
myshape->IDLffShape::GetProperty, N_ENTITIES=num_ent

;Parsing through the entities and only plotting the state of
;Colorado
FOR x=1, (num_ent-1) DO BEGIN
   ;Get the Attributes for entity x
   attr = myshape->IDLffShape::GetAttributes(x)
   ;See if 'Colorado' is in ATTRIBUTE_1 of the attributes for
   ;entity x
   IF attr.ATTRIBUTE_1 EQ 'Maine' THEN BEGIN
     ;Get entity
     ent = myshape->IDLffShape::GetEntity(x)
     xy = MAP_PROJ_FORWARD((*ent.vertices)[0,*], (*ent.vertices)[1,*], $
      map = myMap)
     ;Plot entity
     POLYFILL, xy[0,*], xy[1,*]
     ;Clean-up of pointers
     myshape->IDLffShape::DestroyEntity, ent
   ENDIF
ENDFOR

;Close the Shapefile
OBJ_DESTROY, myshape

End
```