Subject: Re: MAP_SET vs MAP_PROJ_*
Posted by Liam Gumley on Wed, 10 Nov 2004 16:37:15 GMT
View Forum Message <> Reply to Message

Ken Mankoff wrote:
> In a different thread, I mentioned MAP_SET, and David Fanning wrote:
>
>> Well, the MAP_PROJ_*** routines are quite good and can be used with
>> either direct or object graphics. And of course you get a much more
>> complete set of map projections and options when you use them.
>
>
> I hadn't know about the MAP_PROJ_* routines, as I haven't used IDL since
> 5.6 when they were introduced.
>
> I don't need the extra projections, nor the fine-grained detail of the
> major/minor planet axes, etc.
>
> Can anyone recommend one over the other? Is there a speed difference?
>
> Also, this is an educational product, and it would be great for the kids
> to be able to print & make globes out of their maps. Here are some
> possibilities:
> http://www.progonos.com/furuti/MapProj/Normal/ProjPoly/Foldo ut/foldout.html
>
> Has anyone added map projections to the IDL routines and can offer
> advice? I am comfortable modifying the routines, and in the past have
> added magnetically-aligned continents as an option to MAP_CONTINENTS.
>
> Thanks,
>
> -k.

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?

Cheers,
Liam.
Practical IDL Programming
http://www.gumley.com/

Subject: Re: MAP_SET vs MAP_PROJ_*
Posted by James Kuyper on Wed, 10 Nov 2004 18:05:53 GMT
View Forum Message <> Reply to Message

Liam Gumley wrote:
...
> 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?

Since the pixels per centimeter can be different for the x and y axis,
you'll have to decide which axis it is that you want to be at 1000
meters per pixel. Assuming that it's the Y-axis, this should do it:

lat = 22
lon = 90
resolution = 1000.0
MAP_SET,lat,lon,/LAMBERT,scale=resolution*100*!D.Y_PX_CM,/hi res,/grid,/label

You didn't ask for the coastlines, the grid, or the labels, but I
thought it would be good to have something to tell whether the MAP_SET
was correct.

Subject: Re: MAP_SET vs MAP_PROJ_*
Posted by Liam Gumley on Wed, 10 Nov 2004 21:20:18 GMT
View Forum Message <> Reply to Message

James Kuyper wrote:
> Liam Gumley wrote:
> ...
>
>> 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?
>
>
> Since the pixels per centimeter can be different for the x and y axis,
> you'll have to decide which axis it is that you want to be at 1000
> meters per pixel. Assuming that it's the Y-axis, this should do it:
>
> lat = 22
> lon = 90
> resolution = 1000.0
> MAP_SET,lat,lon,/LAMBERT,scale=resolution*100*!D.Y_PX_CM,/hi res,/grid,/label
>
>
> You didn't ask for the coastlines, the grid, or the labels, but I

> thought it would be good to have something to tell whether the MAP_SET
> was correct.

Thanks James, I am quite familiar with using MAP_SET in this fashion. I
am interested in learning how to achieve the same result with the
MAP_PROJ commands.

Cheers,
Liam.
Practical IDL Programming
http://www.gumley.com/

Subject: Re: MAP_SET vs MAP_PROJ_*
Posted by James Kuyper on Thu, 11 Nov 2004 04:04:11 GMT
View Forum Message <> Reply to Message

Liam Gumley wrote:
> James Kuyper wrote:
>
>> Liam Gumley wrote:
>> ...
>>
>>> 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?
>>
>>
>>
>> Since the pixels per centimeter can be different for the x and y axis,
>> you'll have to decide which axis it is that you want to be at 1000
>> meters per pixel. Assuming that it's the Y-axis, this should do it:
>>
>> lat = 22
>> lon = 90
>> resolution = 1000.0
>> MAP_SET,lat,lon,/LAMBERT,scale=resolution*100*!D.Y_PX_CM,/hi res,/grid,/label
>>
>>
>> You didn't ask for the coastlines, the grid, or the labels, but I
>> thought it would be good to have something to tell whether the MAP_SET
>> was correct.
>
>
> Thanks James, I am quite familiar with using MAP_SET in this fashion. I
> am interested in learning how to achieve the same result with the
> MAP_PROJ commands.

Sorry - I'm not familiar with MAP_PROJ, since it's not available in the version of IDL installed on our machines. What does MAP_PROJ do that makes it unacceptable to use MAP_SET instead?

Subject: Re: MAP_SET vs MAP_PROJ_*
Posted by btt on Mon, 15 Nov 2004 17:11:27 GMT
View Forum Message <> Reply to Message

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
```

Subject: Re: MAP_SET vs MAP_PROJ_*
Posted by JD Smith on Mon, 15 Nov 2004 19:01:41 GMT
View Forum Message <> Reply to Message

On Wed, 10 Nov 2004 23:04:11 -0500, James Kuyper wrote:

> Liam Gumley wrote:
>> James Kuyper wrote:
>>
>>> Liam Gumley wrote:
>>> ...
>>>
>>>> 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?
>>>
>>>
>>>
>>> Since the pixels per centimeter can be different for the x and y axis,
>>> you'll have to decide which axis it is that you want to be at 1000
>>> meters per pixel. Assuming that it's the Y-axis, this should do it:
>>>
>>> lat = 22
>>> lon = 90
>>> resolution = 1000.0
>>> MAP_SET,lat,lon,/LAMBERT,scale=resolution*100*!D.Y_PX_CM,/hi res,/grid,/label
>>>
>>>
>>> You didn't ask for the coastlines, the grid, or the labels, but I
>>> thought it would be good to have something to tell whether the MAP_SET
>>> was correct.
>>
>>
>> Thanks James, I am quite familiar with using MAP_SET in this fashion. I
>> am interested in learning how to achieve the same result with the
>> MAP_PROJ commands.
>
> Sorry - I'm not familiar with MAP_PROJ, since it's not available in the
> version of IDL installed on our machines. What does MAP_PROJ do that
> makes it unacceptable to use MAP_SET instead?

I think MAP_PROJ just gives you access to the MAP_SET projection stuff at a deeper level. Useful for creating your own map transformations for interpolating images of arbitrary size, etc. Basically, if you ever have a generic need for the algorithms (as opposed to the output) embodied in forward and reverse map projections, then the MAP_PROJ_* routines are for you. See an explanation of the technique in this post:

http://groups.google.com/groups?selm=pan.2004.04.09.18.32.37 .106707%40as.arizona.edu

The key here is that you have to understand the relation between your MAP structure and the input/output coordinate units (whereas with MAP_SET much of this is done for you).

Good luck,

JD

Subject: Re: MAP_SET vs MAP_PROJ_*
Posted by Gadhavi on Tue, 08 Feb 2005 16:29:32 GMT
View Forum Message <> Reply to Message

Dear Friends,
I tried to run the program given by Ben that is mapTest. I found that map_continent and map_grid subroutines are not accepting the keyword map_structure. I am using IDL 5.6. The subroutine shown by Ben is very useful for my purpose. Any suggestion to overcome this problem
Thanking you
HG
Ben Tupper wrote:
> 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
```