
Subject: OBJ_NEW: Infinite or invalid (NaN) operands not allowed. IDL 7 vs 8 discrepancy?

Posted by [mikrin](#) on Thu, 07 Jun 2012 23:06:59 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hi,

I've developed some IDL object graphics code to draw the continents onto an orthographic projection of the earth. I'm using the file IDL_Default/resource/maps/continents.shp

Specifically I've run into issues where the code I use (see below) works fine on IDL 8.0 running on a Mac OS X Lion system.

On IDL 7.1 running on 32bit Linux (kernel 3.0.0-19) it produces the error,

% OBJ_NEW: Infinite or invalid (NaN) operands not allowed.

The error happens on the line (see below for code snippet)

```
contPoly = obj_new('IDLgrPolyline', xy, polylines=conn[1:*],  
_extra=e)
```

if the array xy contains 1 or more NaN values then I get the error.

xy is calculated by map_proj_forward (which converts lat/lon to xy for a given map projection). In an orthographic projection parts of some continents are behind the visible part of the globe and thus have undefined xy. IDL 8.0 running on my Mac seems to NOT have a problem with this but IDL 7.1 on Linux chokes on it.

Is this an IDL 7.1 issue that was fixed on IDL 8.0 or is this a Mac/Linux issue?

(BTW, the code also fails on Windows XP running IDL 7.1)

Mike

The specific code is:

```
pro drawContinentLines, MAP=map, oModel
```

```
  ; Use the low res continents data for now
```

```
  contFileNames = filepath('continents.shp', subdir=['resource','maps/  
shape'])
```

```
  ; Create a shape object to hold the continent lines
```

```
  conts = obj_new('IDLffShape', contFileNames)
```

```
  conts->getProperty, n_entities=nEntities
```

```

; Loop through all continents
for s=0L, nEntities-1L do begin
  cont = conts->getEntity(s)
  conn = [0]
  for p=0,cont.n_parts-1L do begin
    startInd = (*cont.parts)[p]
    endInd = p eq cont.n_parts-1 ? cont.n_vertices : (*cont.parts)[p
+1]
    conn = [conn, endInd-startInd, lindgen(endInd-startInd) +
startInd]
  endfor
  xy = map_proj_forward(*cont.vertices, map_structure=map)
  ; Create continent boundary object
  contPoly = obj_new('IDLgrPolyline', xy, polylines=conn[1:*],
_extra=e)
  oModel->add, contPoly
  conts->destroyEntity, cont
endfor

obj_destroy, conts

end

```
