
Subject: Re: Contour: difference between IDL 8.0 and IDL 8.4
Posted by chris_torrence@NOSPAM on Tue, 21 Feb 2017 16:25:30 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi Stefano,

Here's what happened - as soon as we released IDL 8.0, users started complaining that the fill behavior was different than the direct graphics contour procedure. So we changed the default behavior to make it easier for people to transition from direct to function graphics. Unfortunately, that broke backwards compatibility for your case.

However, there is a better workaround that avoids having to patch old code (which would break again if you upgrade). You can actually specify a "transparency" value for the c_color values by giving them 4 values ("RGBA" instead of "RGB"). The "A" (or alpha) ranges from 0 (transparent) to 255 (opaque). The documentation doesn't mention this, and I'll fix that for the next version.

Here's a sample program where I do the contour plot, get the color values back out, then set the transparency for one contour. To try out this program you'll need to revert to your unmodified 8.4 code.

I'm sorry about breaking your code, and I hope this workaround helps.

Cheers,
Chris
IDL Project Lead
Harris Geospatial

Sample code:

```
h = hanning(30,30)
x = findgen(30)
c = contour(1-h, x, x, rgb_table=1, /fill, c_value=[0,0.25,0.5,0.75])
colors = c.c_color
dim = size(colors,/dim)
colors = [colors, 255b+bytarr(1,dim[1])] ; change to RGBA
colors[3,3] = 0 ; make transparent
c.c_color = colors
```

```
c1 = contour(1-h, x+10, x+10, rgb_table=2, /fill, $
  c_value=[0,0.25,0.5,0.75], /overplot)
colors = c1.c_color
dim = size(colors,/dim)
colors = [colors, 255b+bytarr(1,dim[1])] ; change to RGBA
colors[3,3] = 0 ; make transparent
c1.c_color = colors
```
