
Subject: VERT_COLORS Problem

Posted by [Udo Grabowski](#) on Mon, 25 Nov 2013 10:03:45 GMT

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

Hi,

I struggle with an animated tool that uses the "new style" surface function, and I'm trying to update the vert_colors option (which is tagged as updateable in the docs), but it does not work at all. What I'm doing wrong here ?

```
; boring flat grey semi-transparent surface
; (poor man's transparency, as always in IDL....)
v = dist(10)
rgba = intarr(4,100)
rgba[0:*,0:*] = 110
S = surface(v,vert_colors=rgba)
```

```
; update to light grey nearly transparent
rgba[0:*,0:*] = 210
S.vert_colors=rgba
; still the same boring grey !
```

I tried several variants, setting vert_colors to 0, refresh, different sequences in doing that, using S->SetProperty,vert_colors=..., etc.etc., no help, it seems that this is just an immutable variable and the documentation is wrong. Used to work with 8.1, tried that also with 8.2.3, but the 8.2.x version is so utterly borked in 3D that even reporting all bugs is far beyond my time constraints....

IDL is using the OpenGL hardware driver of my NVIDIA Quadro 2000 (running the solaris 64-bit version on openindiana 151a7). Any ideas how to get that working ?

Subject: Re: VERT_COLORS Problem

Posted by [David Fanning](#) on Mon, 25 Nov 2013 14:00:14 GMT

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

Udo Grabowski writes:

```
> I struggle with an animated tool that uses the "new style"
> surface function, and I'm trying to update the vert_colors
> option (which is tagged as updateable in the docs), but
> it does not work at all. What I'm doing wrong here ?
>
```

```
> ; boring flat grey semi-transparent surface
> ; (poor man's transparency, as always in IDL....)
> v = dist(10)
> rgba = intarr(4,100)
> rgba[0:*,0:*] = 110
> S = surface(v,vert_colors=rgba)
>
> ; update to light grey nearly transparent
> rgba[0:*,0:*] = 210
> S.vert_colors=rgba
> ; still the same boring grey !
>
> I tried several variants, setting vert_colors to 0,
> refresh, different sequences in doing that, using
> S->SetProperty,vert_colors=..., etc.etc.,
> no help, it seems that this is just an immutable
> variable and the documentation is wrong.
```

It would seem so, although I just tested this with my cgSurface program and it would appear it is *possible* to change the vert_colors. In my case, though, it is necessary to first turn the vert colors off, then on again. In my test case, the code looks like this:

```
rgba = info.rgba
rgba[2,*] = 0
info.thisSurface -> SetProperty, Vert_Colors=0
info.thisSurface -> SetProperty, Vert_Colors=rgba
```

I did confirm this does NOT work with the Surface function though, so something else is broken there.

Cheers,

David

--

David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: <http://www.idlcoyote.com/>
Sepore ma de ni thue. ("Perhaps thou speakest truth.")

Subject: Re: VERT_COLORS Problem
Posted by chris_torrence@NOSPAM on Mon, 02 Dec 2013 17:57:50 GMT
[View Forum Message](#) <> [Reply to Message](#)

On Monday, November 25, 2013 7:00:14 AM UTC-7, David Fanning wrote:

```
> Udo Grabowski writes:
>
>
>
>> I struggle with an animated tool that uses the "new style"
>
>> surface function, and I'm trying to update the vert_colors
>
>> option (which is tagged as updateable in the docs), but
>
>> it does not work at all. What I'm doing wrong here ?
>
>>
>
>> ; boring flat grey semi-transparent surface
>
>> ; (poor man's transparency, as always in IDL....)
>
>> v = dist(10)
>
>> rgba = intarr(4,100)
>
>> rgba[0:*,0:*] = 110
>
>> S = surface(v,vert_colors=rgba)
>
>>
>
>> ; update to light grey nearly transparent
>
>> rgba[0:*,0:*] = 210
>
>> S.vert_colors=rgba
>
>> ; still the same boring grey !
>
>>
>
>> I tried several variants, setting vert_colors to 0,
>
>> refresh, different sequences in doing that, using
>
>> S->SetProperty,vert_colors=..., etc.etc.,
>
>> no help, it seems that this is just an immutable
>
>> variable and the documentation is wrong.
>
```

>
>
> It would seem so, although I just tested this with my cgSurface program
>
> and it would appear it is *possible* to change the vert_colors. In my
>
> case, though, it is necessary to first turn the vert colors off, then on
>
> again. In my test case, the code looks like this:
>
>
> rgba = info.rgba
>
> rgba[2,*] = 0
>
> info.thisSurface -> SetProperty, Vert_Colors=0
>
> info.thisSurface -> SetProperty, Vert_Colors=rgba
>
>
>
> I did confirm this does NOT work with the Surface function though, so
>
> something else is broken there.
>
>
>
> Cheers,
>
>
>
> David
>
>
>
>
>
> --
>
> David Fanning, Ph.D.
>
> Fanning Software Consulting, Inc.
>
> Coyote's Guide to IDL Programming: <http://www.idlcoyote.com/>
>
> Sepore ma de ni thue. ("Perhaps thou speakest truth.")

Hi all,

This is indeed a bug. For whatever reason, the `vert_colors` property is being swallowed. I've fixed the code for the next IDL release (after IDL 8.3), but in the meantime, you have two options: you can either pass in the `vert_colors` when you create the surface, or you can hack in the fix. To fix the code, edit `<idl_dir>/lib/itools/components/idlitvissurface__define.pro`. In the `::SetProperty` method, look for the following line:

```
VERT_COLORS=swallow, $
```

Change this to:

```
VERT_COLORS=vertColors, $
```

Then, right after the `compile_opt` line, add the following code:

```
if (N_ELEMENTS(vertColors) gt 0) then begin
  oVertColor = self->getParameter('VERTEX COLORS')
  if (ISA(oVertColor)) then begin
    success = oVertColor->SetData(vertColors)
  endif else begin
    self._oSurface->SetProperty, VERT_COLORS=vertColors
  endelse
endif
```

Hope this helps!

Cheers,

Chris

ExelisVIS
