

---

Subject: Contour: difference between IDL 8.0 and IDL 8.4

Posted by [burkina](#) on Thu, 16 Feb 2017 11:59:36 GMT

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

---

Ciao,

I've upgraded from IDL 8.0 to IDL 8.4, and found a strange difference in the behaviour of CONTOUR.

Here's a simple test:

```
data = RANDOMU(seed, 9, 9)
smooth = CONTOUR(MIN_CURVE_SURF(data), TITLE='Smoothed', RGB_TABLE=1, /FILL,
c_value=[0, 0.2, 0.3, 0.5])
```

If you launch it in IDL 8.0, you get what you (more or less) expect: three filled contours, while the outer part is the white transparent background. I need to include the '0' level, otherwise the inner contour is left blank (BTW, I find it odd, and it is the same for IDL 8.4).

On the other hand, if you launch it in IDL 8.4, there's a big difference: also the outer part is filled! Of course, I can give an explicit white color to it, but it's not the same, because it is not treated as a completely transparent background. So, when you put a contour above another, in IDL 8.0 everything is fine, because the outer part is completely transparent, while in IDL 8.4, you see also this outer part. Using 'max\_value=0.5' is not an equivalent solution.

What changed between 8.0 and 8.4? How can I revert to the 8.0 behaviour? I basically want to draw three filled contours, and leave the white transparent background in the rest of the image.

Thanks,

Stefano

---

---

Subject: Re: Contour: difference between IDL 8.0 and IDL 8.4

Posted by [burkina](#) on Fri, 17 Feb 2017 09:55:45 GMT

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

---

On Thursday, 16 February 2017 12:59:39 UTC+1, burkina wrote:

> Ciao,

>  
> I've upgraded from IDL 8.0 to IDL 8.4, and found a strange difference in the behaviour of CONTOUR.

>  
> Here's a simple test:

>  
> data = RANDOMU(seed, 9, 9)  
> smooth = CONTOUR(MIN\_CURVE\_SURF(data), TITLE='Smoothed', RGB\_TABLE=1, /FILL,  
c\_value=[0, 0.2, 0.3, 0.5])

>  
> If you launch it in IDL 8.0, you get what you (more or less) expect: three filled contours, while the outer part is the white transparent background. I need to include the '0' level, otherwise the inner contour is left blank (BTW, I find it odd, and it is the same for IDL 8.4).  
>  
> On the other hand, if you launch it in IDL 8.4, there's a big difference: also the outer part is filled! Of course, I can give an explicit white color to it, but it's not the same, because it is not treated as a completely transparent background. So, when you put a contour above another, in IDL 8.0 everything is fine, because the outer part is completely transparent, while in IDL 8.4, you see also this outer part. Using 'max\_value=0.5' is not an equivalent solution.  
>  
> What changed between 8.0 and 8.4? How can I revert to the 8.0 behaviour? I basically want to draw three filled contours, and leave the white transparent background in the rest of the image.  
>  
> Thanks,  
>  
> Stefano

Hi,

just to be more specific, this is a plot I had in IDL 8.0:

<https://img42.com/MOnas>

and this is what I get (exactly with the same code) in IDL 8.4:

<https://img42.com/KSkRc>

As you can see, in the latter the background of the reddish contour plot is still visible, because it's indeed a filled contour level, while in IDL 8.0 it is simply background. How can I get rid of it?

Something clearly changed after IDL 8.0, I hope there's some kind of switch/parameter that I don't understand that can be set to have the old behaviour. I even thought about reconstructing the contour levels polygons, and then fill them, but, apparently, there's no way to get the polygons in function graphics (!?).

Thanks,

Stefano

P.S. I had a chance to test this code on IDL 8.2, and it behaves like 8.4.

---

Subject: Re: Contour: difference between IDL 8.0 and IDL 8.4

Posted by [burkina](#) on Fri, 17 Feb 2017 16:45:56 GMT

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

---

On Friday, 17 February 2017 16:30:47 UTC+1, Jim P wrote:

```

> On Friday, February 17, 2017 at 2:55:48 AM UTC-7, burkina wrote:
>> On Thursday, 16 February 2017 12:59:39 UTC+1, burkina wrote:
>>> Ciao,
>>>
>>> I've upgraded from IDL 8.0 to IDL 8.4, and found a strange difference in the behaviour of
CONTOUR.
>>>
>>> Here's a simple test:
>>>
>>> data = RANDOMU(seed, 9, 9)
>>> smooth = CONTOUR(MIN_CURVE_SURF(data), TITLE='Smoothed', RGB_TABLE=1, /FILL,
c_value=[0, 0.2, 0.3, 0.5])
>>>
>>> If you launch it in IDL 8.0, you get what you (more or less) expect: three filled contours, while
the outer part is the white transparent background. I need to include the '0' level, otherwise the
inner contour is left blank (BTW, I find it odd, and it is the same for IDL 8.4).
>>>
>>> On the other hand, if you launch it in IDL 8.4, there's a big difference: also the outer part is
filled! Of course, I can give an explicit white color to it, but it's not the same, because it is not
treated as a completely transparent background. So, when you put a contour above another, in
IDL 8.0 everything is fine, because the outer part is completely transparent, while in IDL 8.4, you
see also this outer part. Using 'max_value=0.5' is not an equivalent solution.
>>>
>>> What changed between 8.0 and 8.4? How can I revert to the 8.0 behaviour? I basically want
to draw three filled contours, and leave the white transparent background in the rest of the image.
>>>
>>> Thanks,
>>>
>>> Stefano
>>
>> Hi,
>>
>> just to be more specific, this is a plot I had in IDL 8.0:
>>
>> https://img42.com/MOnas
>>
>> and this is what I get (exactly with the same code) in IDL 8.4:
>>
>> https://img42.com/KSkRc
>>
>> As you can see, in the latter the background of the reddish contour plot is still visible, because
it's indeed a filled contour level, while in IDL 8.0 it is simply background. How can I get rid of it?
>>
>> Something clearly changed after IDL 8.0, I hope there's some kind of switch/parameter that I
don't understand that can be set to have the old behaviour. I even thought about reconstructing
the contour levels polygons, and then fill them, but, apparently, there's no way to get the polygons
in function graphics (!?).
>>

```

```
>> Thanks,
>>
>> Stefano
>>
>> P.S. I had a chance to test this code on IDL 8.2, and it behaves like 8.4.
>
> I don't have a specific solution, but here are a couple things to investigate.
>
> According to the docs for the CONTOUR function,
http://www.harrisgeospatial.com/docs/CONTOUR.html, a number of keywords were added in IDL
8.2 that may address this. These include BACKGROUND_COLOR and
BACKGROUND_TRANSPARENCY, and probably changed the default behaviors.
>
> Comparing the contour properties in 8.0 versus later versions may help, as well, e.g.,
>
> IDL> print, smooth
>
> For example, the "c_color" property represented a color table index in 8.0 [0, 47, 73, 125], while
in later versions c_color contains RGB triplets, [[0, 0, 13], [0, 13, 141], ...].
```

Dear Jim,

thank you for your reply.

As for the new keywords, BACKGROUND\_COLOR and BACKGROUND\_TRANSPARENCY, they have no effect in my case, because they simply change the background color and transparency (as expected), but my problem is that in 8.2+ that part is not background, but a filled contour!

On the other hand, I had a look at the contour properties, as you suggested, and it seems to me that there are two main differences.

1. IDL 8.0: N\_LEVELS = 4 -- IDL 8.4: N\_LEVELS = 0

If I include a n\_levels=3 keyword in the command line, nothing changes for IDL 8.4 (as expected, the c\_values are dominant), while in IDL 8.0 the c\_values are ignored (which is not what I would expect). I don't know how this could affect my issue

2. As you correctly pointed out, c\_color is different. So I changed the keyword in

```
c_value=[level_0,level_1, level_2, level_3], c_color=['blue', 'green', 'red']
```

which still reproduces the issue I'm trying to solve, but it seems to me easier to interpret. And here's a hint:

```
IDL 8.0: print, smooth.c_color
```

```
0 0 255
0 127 0
```

255 0 0

IDL8.4: print, smooth.c\_color

```
0 0 255
0 127 0
255 0 0
0 0 255
0 127 0
```

There are indeed two further colors defined! I think this is why the contours are filled also outside. No matter if I define less colors, always n+1 colors are stored in c\_color (repeated cyclically). So how can I delete these two colors and have only background after the first three contours?

Further ideas?

In any case, I found the fill contour implementation in IDL counter-intuitive, because it seems to me that it actually fills the outside of the contour, not the inside...

Thanks,

Stefano

---

Subject: Re: Contour: difference between IDL 8.0 and IDL 8.4

Posted by [burkina](#) on Fri, 17 Feb 2017 17:13:46 GMT

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

---

On Friday, 17 February 2017 17:45:59 UTC+1, burkina wrote:

> On Friday, 17 February 2017 16:30:47 UTC+1, Jim P wrote:

>> On Friday, February 17, 2017 at 2:55:48 AM UTC-7, burkina wrote:

>>> On Thursday, 16 February 2017 12:59:39 UTC+1, burkina wrote:

>>>> Ciao,

>>>>

>>>> I've upgraded from IDL 8.0 to IDL 8.4, and found a strange difference in the behaviour of CONTOUR.

>>>>

>>>> Here's a simple test:

>>>>

>>>> data = RANDOMU(seed, 9, 9)

>>>> smooth = CONTOUR(MIN\_CURVE\_SURF(data), TITLE='Smoothed', RGB\_TABLE=1,  
/FILL, c\_value=[0, 0.2, 0.3, 0.5])

>>>>

>>>> If you launch it in IDL 8.0, you get what you (more or less) expect: three filled contours, while the outer part is the white transparent background. I need to include the '0' level, otherwise the inner contour is left blank (BTW, I find it odd, and it is the same for IDL 8.4).

>>>>

>>>> On the other hand, if you launch it in IDL 8.4, there's a big difference: also the outer part is

filled! Of course, I can give an explicit white color to it, but it's not the same, because it is not treated as a completely transparent background. So, when you put a contour above another, in IDL 8.0 everything is fine, because the outer part is completely transparent, while in IDL 8.4, you see also this outer part. Using 'max\_value=0.5' is not an equivalent solution.

>>>>

>>>> What changed between 8.0 and 8.4? How can I revert to the 8.0 behaviour? I basically want to draw three filled contours, and leave the white transparent background in the rest of the image.

>>>>

>>>> Thanks,

>>>>

>>>> Stefano

>>>

>>> Hi,

>>>

>>> just to be more specific, this is a plot I had in IDL 8.0:

>>>

>>> <https://img42.com/MOnas>

>>>

>>> and this is what I get (exactly with the same code) in IDL 8.4:

>>>

>>> <https://img42.com/KSkRc>

>>>

>>> As you can see, in the latter the background of the reddish contour plot is still visible, because it's indeed a filled contour level, while in IDL 8.0 it is simply background. How can I get rid of it?

>>>

>>> Something clearly changed after IDL 8.0, I hope there's some kind of switch/parameter that I don't understand that can be set to have the old behaviour. I even thought about reconstructing the contour levels polygons, and then fill them, but, apparently, there's no way to get the polygons in function graphics (!?).

>>>

>>> Thanks,

>>>

>>> Stefano

>>>

>>> P.S. I had a chance to test this code on IDL 8.2, and it behaves like 8.4.

>>

>> I don't have a specific solution, but here are a couple things to investigate.

>>

>> According to the docs for the CONTOUR function, <http://www.harrisgeospatial.com/docs/CONTOUR.html>, a number of keywords were added in IDL 8.2 that may address this. These include BACKGROUND\_COLOR and BACKGROUND\_TRANSPARENCY, and probably changed the default behaviors.

>>

>> Comparing the contour properties in 8.0 versus later versions may help, as well, e.g.,

>>

>> IDL> print, smooth

```

>>
>> For example, the "c_color" property represented a color table index in 8.0 [0, 47, 73, 125],
while in later versions c_color contains RGB triplets, [[0, 0, 13], [0, 13, 141], ...].
>
> Dear Jim,
>
> thank you for your reply.
>
> As for the new keywords, BACKGROUND_COLOR and BACKGROUND_TRANSPARENCY,
they have no effect in my case, because they simply change the background color and
transparency (as expected), but my problem is that in 8.2+ that part is not background, but a filled
contour!
>
> On the other hand, I had a look at the contour properties, as you suggested, and it seems to
me that there are two main differences.
>
> 1. IDL 8.0: N_LEVELS = 4 -- IDL 8.4: N_LEVELS = 0
>
> If I include a n_levels=3 keyword in the command line, nothing changes for IDL 8.4 (as
expected, the c_values are dominant), while in IDL 8.0 the c_values are ignored (which is not
what I would expect). I don't know how this could affect my issue
>
> 2. As you correctly pointed out, c_color is different. So I changed the keyword in
>
> c_value=[level_0,level_1, level_2, level_3], c_color=['blue', 'green', 'red']
>
> which still reproduces the issue I'm trying to solve, but it seems to me easier to interpret. And
here's a hint:
>
> IDL 8.0: print, smooth.c_color
>
> 0 0 255
> 0 127 0
> 255 0 0
>
> IDL8.4: print, smooth.c_color
>
> 0 0 255
> 0 127 0
> 255 0 0
> 0 0 255
> 0 127 0
>
> There are indeed two further colors defined! I think this is why the contours are filled also
outside. No matter if I define less colors, always n+1 colors are stored in c_color (repeated
cyclically). So how can I delete these two colors and have only background after the first three
contours?
>

```

> Further ideas?  
>  
> In any case, I found the fill contour implementation in IDL counter-intuitive, because it seems to me that it actually fills the outside of the contour, not the inside...  
>  
> Thanks,  
>  
> Stefano

As a further test, I re-define c\_color after the plot:

```
print, c2.c_color
```

```
help, c2.c_color
```

```
c2.c_color=[[0,0, 255],[0 ,127 , 0],[255,  0 , 0]]
```

```
print, c2.c_color
```

```
help, c2.c_color
```

But, although now I only have three colors as in IDL8.0, nothing changes in the plot:

```
  0  0 255
  0 127  0
255  0  0
  0  0 255
  0 127  0
<Expression>  BYTE    = Array[3, 5]
200 200 255
  0 127  0
255  0  0
<Expression>  BYTE    = Array[3, 3]
```

---

---

Subject: Re: Contour: difference between IDL 8.0 and IDL 8.4

Posted by [burkina](#) on Fri, 17 Feb 2017 17:14:44 GMT

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

---

On Friday, 17 February 2017 17:45:59 UTC+1, burkina wrote:

> On Friday, 17 February 2017 16:30:47 UTC+1, Jim P wrote:

>> On Friday, February 17, 2017 at 2:55:48 AM UTC-7, burkina wrote:

>>> On Thursday, 16 February 2017 12:59:39 UTC+1, burkina wrote:

>>>> Ciao,

>>>>

>>>> I've upgraded from IDL 8.0 to IDL 8.4, and found a strange difference in the behaviour of CONTOUR.

>>>>



```
>>>> Here's a simple test:
>>>>
>>>> data = RANDOMU(seed, 9, 9)
>>>> smooth = CONTOUR(MIN_CURVE_SURF(data), TITLE='Smoothed', RGB_TABLE=1,
/FILL, c_value=[0, 0.2, 0.3, 0.5])
>>>>
>>>> If you launch it in IDL 8.0, you get what you (more or less) expect: three filled contours,
while the outer part is the white transparent background. I need to include the '0' level, otherwise
the inner contour is left blank (BTW, I find it odd, and it is the same for IDL 8.4).
>>>>
>>>> On the other hand, if you launch it in IDL 8.4, there's a big difference: also the outer part is
filled! Of course, I can give an explicit white color to it, but it's not the same, because it is not
treated as a completely transparent background. So, when you put a contour above another, in
IDL 8.0 everything is fine, because the outer part is completely transparent, while in IDL 8.4, you
see also this outer part. Using 'max_value=0.5' is not an equivalent solution.
>>>>
>>>> What changed between 8.0 and 8.4? How can I revert to the 8.0 behaviour? I basically
want to draw three filled contours, and leave the white transparent background in the rest of the
image.
>>>>
>>>> Thanks,
>>>>
>>>> Stefano
>>>
>>> Hi,
>>>
>>> just to be more specific, this is a plot I had in IDL 8.0:
>>>
>>> https://img42.com/MOnas
>>>
>>> and this is what I get (exactly with the same code) in IDL 8.4:
>>>
>>> https://img42.com/KSkRc
>>>
>>> As you can see, in the latter the background of the reddish contour plot is still visible,
because it's indeed a filled contour level, while in IDL 8.0 it is simply background. How can I get
rid of it?
>>>
>>> Something clearly changed after IDL 8.0, I hope there's some kind of switch/parameter that I
don't understand that can be set to have the old behaviour. I even thought about reconstructing
the contour levels polygons, and then fill them, but, apparently, there's no way to get the polygons
in function graphics (!?).
>>>
>>> Thanks,
>>>
>>> Stefano
>>>
>>> P.S. I had a chance to test this code on IDL 8.2, and it behaves like 8.4.
```

```

>>
>> I don't have a specific solution, but here are a couple things to investigate.
>>
>> According to the docs for the CONTOUR function,
http://www.harrisgeospatial.com/docs/CONTOUR.html, a number of keywords were added in IDL
8.2 that may address this. These include BACKGROUND_COLOR and
BACKGROUND_TRANSPARENCY, and probably changed the default behaviors.
>>
>> Comparing the contour properties in 8.0 versus later versions may help, as well, e.g.,
>>
>> IDL> print, smooth
>>
>> For example, the "c_color" property represented a color table index in 8.0 [0, 47, 73, 125],
while in later versions c_color contains RGB triplets, [[0, 0, 13], [0, 13, 141], ...].
>
> Dear Jim,
>
> thank you for your reply.
>
> As for the new keywords, BACKGROUND_COLOR and BACKGROUND_TRANSPARENCY,
they have no effect in my case, because they simply change the background color and
transparency (as expected), but my problem is that in 8.2+ that part is not background, but a filled
contour!
>
> On the other hand, I had a look at the contour properties, as you suggested, and it seems to
me that there are two main differences.
>
> 1. IDL 8.0: N_LEVELS = 4 -- IDL 8.4: N_LEVELS = 0
>
> If I include a n_levels=3 keyword in the command line, nothing changes for IDL 8.4 (as
expected, the c_values are dominant), while in IDL 8.0 the c_values are ignored (which is not
what I would expect). I don't know how this could affect my issue
>
> 2. As you correctly pointed out, c_color is different. So I changed the keyword in
>
> c_value=[level_0,level_1, level_2, level_3], c_color=['blue', 'green', 'red']
>
> which still reproduces the issue I'm trying to solve, but it seems to me easier to interpret. And
here's a hint:
>
> IDL 8.0: print, smooth.c_color
>
> 0 0 255
> 0 127 0
> 255 0 0
>
> IDL8.4: print, smooth.c_color
>

```

```

> 0 0 255
> 0 127 0
> 255 0 0
> 0 0 255
> 0 127 0
>
> There are indeed two further colors defined! I think this is why the contours are filled also
outside. No matter if I define less colors, always n+1 colors are stored in c_color (repeated
cyclically). So how can I delete these two colors and have only background after the first three
contours?
>
> Further ideas?
>
> In any case, I found the fill contour implementation in IDL counter-intuitive, because it seems to
me that it actually fills the outside of the contour, not the inside...
>
> Thanks,
>
> Stefano

```

As a further test, I re-define c\_color after the plot:

```

print, c2.c_color

help, c2.c_color

c2.c_color=[[0,0, 255],[0 ,127 , 0],[255, 0 , 0]]

print, c2.c_color

help, c2.c_color

```

But, although now I only have three colors as in IDL8.0, nothing changes in the plot:

```

0 0 255
0 127 0
255 0 0
0 0 255
0 127 0
<Expression>  BYTE  = Array[3, 5]
0 0 255
0 127 0
255 0 0
<Expression>  BYTE  = Array[3, 3]

```

---

Subject: Re: Contour: difference between IDL 8.0 and IDL 8.4

On Friday, February 17, 2017 at 10:14:48 AM UTC-7, burkina wrote:

> On Friday, 17 February 2017 17:45:59 UTC+1, burkina wrote:

>> On Friday, 17 February 2017 16:30:47 UTC+1, Jim P wrote:

>>> On Friday, February 17, 2017 at 2:55:48 AM UTC-7, burkina wrote:

>>>> On Thursday, 16 February 2017 12:59:39 UTC+1, burkina wrote:

>>>> > Ciao,

>>>> >

>>>> > I've upgraded from IDL 8.0 to IDL 8.4, and found a strange difference in the behaviour of CONTOUR.

>>>> >

>>>> > Here's a simple test:

>>>> >

>>>> > data = RANDOMU(seed, 9, 9)

>>>> > smooth = CONTOUR(MIN\_CURVE\_SURF(data), TITLE='Smoothed', RGB\_TABLE=1,  
/FILL, c\_value=[0, 0.2, 0.3, 0.5])

>>>> >

>>>> > If you launch it in IDL 8.0, you get what you (more or less) expect: three filled contours, while the outer part is the white transparent background. I need to include the '0' level, otherwise the inner contour is left blank (BTW, I find it odd, and it is the same for IDL 8.4).

>>>> >

>>>> > On the other hand, if you launch it in IDL 8.4, there's a big difference: also the outer part is filled! Of course, I can give an explicit white color to it, but it's not the same, because it is not treated as a completely transparent background. So, when you put a contour above another, in IDL 8.0 everything is fine, because the outer part is completely transparent, while in IDL 8.4, you see also this outer part. Using 'max\_value=0.5' is not an equivalent solution.

>>>> >

>>>> > What changed between 8.0 and 8.4? How can I revert to the 8.0 behaviour? I basically want to draw three filled contours, and leave the white transparent background in the rest of the image.

>>>> >

>>>> > Thanks,

>>>> >

>>>> > Stefano

>>>>

>>>> Hi,

>>>>

>>>> just to be more specific, this is a plot I had in IDL 8.0:

>>>>

>>>> <https://img42.com/MOnas>

>>>>

>>>> and this is what I get (exactly with the same code) in IDL 8.4:

>>>>

>>>> <https://img42.com/KSkRc>

>>>>

>>>> As you can see, in the latter the background of the reddish contour plot is still visible, because it's indeed a filled contour level, while in IDL 8.0 it is simply background. How can I get

rid of it?

>>>>

>>>> Something clearly changed after IDL 8.0, I hope there's some kind of switch/parameter that I don't understand that can be set to have the old behaviour. I even thought about reconstructing the contour levels polygons, and then fill them, but, apparently, there's no way to get the polygons in function graphics (!?).

>>>>

>>>> Thanks,

>>>>

>>>> Stefano

>>>>

>>>> P.S. I had a chance to test this code on IDL 8.2, and it behaves like 8.4.

>>>

>>> I don't have a specific solution, but here are a couple things to investigate.

>>>

>>> According to the docs for the CONTOUR function,

<http://www.harrisgeospatial.com/docs/CONTOUR.html>, a number of keywords were added in IDL 8.2 that may address this. These include BACKGROUND\_COLOR and BACKGROUND\_TRANSPARENCY, and probably changed the default behaviors.

>>>

>>> Comparing the contour properties in 8.0 versus later versions may help, as well, e.g.,

>>>

>>> IDL> print, smooth

>>>

>>> For example, the "c\_color" property represented a color table index in 8.0 [0, 47, 73, 125], while in later versions c\_color contains RGB triplets, [[0, 0, 13], [0, 13, 141], ...].

>>

>> Dear Jim,

>>

>> thank you for your reply.

>>

>> As for the new keywords, BACKGROUND\_COLOR and BACKGROUND\_TRANSPARENCY, they have no effect in my case, because they simply change the background color and transparency (as expected), but my problem is that in 8.2+ that part is not background, but a filled contour!

>>

>> On the other hand, I had a look at the contour properties, as you suggested, and it seems to me that there are two main differences.

>>

>> 1. IDL 8.0: N\_LEVELS = 4 -- IDL 8.4: N\_LEVELS = 0

>>

>> If I include a n\_levels=3 keyword in the command line, nothing changes for IDL 8.4 (as expected, the c\_values are dominant), while in IDL 8.0 the c\_values are ignored (which is not what I would expect). I don't know how this could affect my issue

>>

>> 2. As you correctly pointed out, c\_color is different. So I changed the keyword in

>>

>> c\_value=[level\_0,level\_1, level\_2, level\_3], c\_color=['blue', 'green', 'red']

```

>>
>> which still reproduces the issue I'm trying to solve, but it seems to me easier to interpret. And
here's a hint:
>>
>> IDL 8.0: print, smooth.c_color
>>
>> 0 0 255
>> 0 127 0
>> 255 0 0
>>
>> IDL8.4: print, smooth.c_color
>>
>> 0 0 255
>> 0 127 0
>> 255 0 0
>> 0 0 255
>> 0 127 0
>>
>> There are indeed two further colors defined! I think this is why the contours are filled also
outside. No matter if I define less colors, always n+1 colors are stored in c_color (repeated
cyclically). So how can I delete these two colors and have only background after the first three
contours?
>>
>> Further ideas?
>>
>> In any case, I found the fill contour implementation in IDL counter-intuitive, because it seems
to me that it actually fills the outside of the contour, not the inside...
>>
>> Thanks,
>>
>> Stefano
>
> As a further test, I re-define c_color after the plot:
>
> print, c2.c_color
>
> help, c2.c_color
>
> c2.c_color=[[0,0, 255],[0 ,127 , 0],[255, 0 , 0]]
>
> print, c2.c_color
>
> help, c2.c_color
>
> But, although now I only have three colors as in IDL8.0, nothing changes in the plot:
>
> 0 0 255
> 0 127 0

```

```

> 255 0 0
> 0 0 255
> 0 127 0
> <Expression> BYTE = Array[3, 5]
> 0 0 255
> 0 127 0
> 255 0 0
> <Expression> BYTE = Array[3, 3]

```

One other debugging trick is comparing the source code of the two classes IDLitVISContour and IDLitVISContourLevel, between the version 8.0 and 8.X.

I found that if I executed the contour() function in IDL 8.5 first (to ensure the routines were compiled), then recompiled the two class definition files from my IDL 8.0 distribution on the same computer, my IDL 8.5 results now look like my IDL 8.0 results.

They are in the lib\itools\components subdirectory of each distribution.

The challenge, then, is to find the causal difference between the two versions. I don't know if it's necessary to investigate both classes. Only one of the two may be the problem.

If you open two projects in your IDL Workbench, one based in the root of the IDL 8.0 lib directory and one in that of IDL 8.X, you can multi-select the two source files for each class then execute a "compare" via a right click in the Project Explorer window.

Jim P.

---

Subject: Re: Contour: difference between IDL 8.0 and IDL 8.4  
 Posted by [burkina](#) on Mon, 20 Feb 2017 10:30:12 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

```

On Sunday, 19 February 2017 04:55:05 UTC+1, Jim P wrote:
> On Friday, February 17, 2017 at 10:14:48 AM UTC-7, burkina wrote:
>> On Friday, 17 February 2017 17:45:59 UTC+1, burkina wrote:
>>> On Friday, 17 February 2017 16:30:47 UTC+1, Jim P wrote:
>>>> On Friday, February 17, 2017 at 2:55:48 AM UTC-7, burkina wrote:
>>>> > On Thursday, 16 February 2017 12:59:39 UTC+1, burkina wrote:
>>>> > > Ciao,
>>>> > >
>>>> > > I've upgraded from IDL 8.0 to IDL 8.4, and found a strange difference in the behaviour
of CONTOUR.
>>>> > >
>>>> > > Here's a simple test:
>>>> > >
>>>> > > data = RANDOMU(seed, 9, 9)
>>>> > > smooth = CONTOUR(MIN_CURVE_SURF(data), TITLE='Smoothed', RGB_TABLE=1,
/FILL, c_value=[0, 0.2, 0.3, 0.5])

```

```

>>>> > >
>>>> > > If you launch it in IDL 8.0, you get what you (more or less) expect: three filled contours,
while the outer part is the white transparent background. I need to include the '0' level, otherwise
the inner contour is left blank (BTW, I find it odd, and it is the same for IDL 8.4).
>>>> > >
>>>> > > On the other hand, if you launch it in IDL 8.4, there's a big difference: also the outer part
is filled! Of course, I can give an explicit white color to it, but it's not the same, because it is not
treated as a completely transparent background. So, when you put a contour above another, in
IDL 8.0 everything is fine, because the outer part is completely transparent, while in IDL 8.4, you
see also this outer part. Using 'max_value=0.5' is not an equivalent solution.
>>>> > >
>>>> > > What changed between 8.0 and 8.4? How can I revert to the 8.0 behaviour? I basically
want to draw three filled contours, and leave the white transparent background in the rest of the
image.
>>>> > >
>>>> > > Thanks,
>>>> > >
>>>> > > Stefano
>>>> >
>>>> > Hi,
>>>> >
>>>> > just to be more specific, this is a plot I had in IDL 8.0:
>>>> >
>>>> > https://img42.com/MOnas
>>>> >
>>>> > and this is what I get (exactly with the same code) in IDL 8.4:
>>>> >
>>>> > https://img42.com/KSkRc
>>>> >
>>>> > As you can see, in the latter the background of the reddish contour plot is still visible,
because it's indeed a filled contour level, while in IDL 8.0 it is simply background. How can I get
rid of it?
>>>> >
>>>> > Something clearly changed after IDL 8.0, I hope there's some kind of switch/parameter
that I don't understand that can be set to have the old behaviour. I even thought about
reconstructing the contour levels polygons, and then fill them, but, apparently, there's no way to
get the polygons in function graphics (!?).
>>>> >
>>>> > Thanks,
>>>> >
>>>> > Stefano
>>>> >
>>>> > P.S. I had a chance to test this code on IDL 8.2, and it behaves like 8.4.
>>>>
>>>> I don't have a specific solution, but here are a couple things to investigate.
>>>>
>>>> According to the docs for the CONTOUR function,
http://www.harrisgeospatial.com/docs/CONTOUR.html, a number of keywords were added in IDL

```



8.2 that may address this. These include BACKGROUND\_COLOR and BACKGROUND\_TRANSPARENCY, and probably changed the default behaviors.

>>>>

>>>> Comparing the contour properties in 8.0 versus later versions may help, as well, e.g.,

>>>>

>>>> IDL> print, smooth

>>>>

>>>> For example, the "c\_color" property represented a color table index in 8.0 [0, 47, 73, 125], while in later versions c\_color contains RGB triplets, [[0, 0, 13], [0, 13, 141], ...].

>>>

>>> Dear Jim,

>>>

>>> thank you for your reply.

>>>

>>> As for the new keywords, BACKGROUND\_COLOR and BACKGROUND\_TRANSPARENCY, they have no effect in my case, because they simply change the background color and transparency (as expected), but my problem is that in 8.2+ that part is not background, but a filled contour!

>>>

>>> On the other hand, I had a look at the contour properties, as you suggested, and it seems to me that there are two main differences.

>>>

>>> 1. IDL 8.0: N\_LEVELS = 4 -- IDL 8.4: N\_LEVELS = 0

>>>

>>> If I include a n\_levels=3 keyword in the command line, nothing changes for IDL 8.4 (as expected, the c\_values are dominant), while in IDL 8.0 the c\_values are ignored (which is not what I would expect). I don't know how this could affect my issue

>>>

>>> 2. As you correctly pointed out, c\_color is different. So I changed the keyword in

>>>

>>> c\_value=[level\_0,level\_1, level\_2, level\_3], c\_color=['blue', 'green', 'red']

>>>

>>> which still reproduces the issue I'm trying to solve, but it seems to me easier to interpret. And here's a hint:

>>>

>>> IDL 8.0: print, smooth.c\_color

>>>

>>> 0 0 255

>>> 0 127 0

>>> 255 0 0

>>>

>>> IDL8.4: print, smooth.c\_color

>>>

>>> 0 0 255

>>> 0 127 0

>>> 255 0 0

>>> 0 0 255

>>> 0 127 0

```

>>>
>>> There are indeed two further colors defined! I think this is why the contours are filled also
outside. No matter if I define less colors, always n+1 colors are stored in c_color (repeated
cyclically). So how can I delete these two colors and have only background after the first three
contours?
>>>
>>> Further ideas?
>>>
>>> In any case, I found the fill contour implementation in IDL counter-intuitive, because it seems
to me that it actually fills the outside of the contour, not the inside...
>>>
>>> Thanks,
>>>
>>> Stefano
>>
>> As a further test, I re-define c_color after the plot:
>>
>> print, c2.c_color
>>
>> help, c2.c_color
>>
>> c2.c_color=[[0,0, 255],[0,127, 0],[255, 0, 0]]
>>
>> print, c2.c_color
>>
>> help, c2.c_color
>>
>> But, although now I only have three colors as in IDL8.0, nothing changes in the plot:
>>
>> 0 0 255
>> 0 127 0
>> 255 0 0
>> 0 0 255
>> 0 127 0
>> <Expression>  BYTE      = Array[3, 5]
>> 0 0 255
>> 0 127 0
>> 255 0 0
>> <Expression>  BYTE      = Array[3, 3]
>
> One other debugging trick is comparing the source code of the two classes IDLitVISContour
and IDLitVISContourLevel, between the version 8.0 and 8.X.
>
> I found that if I executed the contour() function in IDL 8.5 first (to ensure the routines were
compiled), then recompiled the two class definition files from my IDL 8.0 distribution on the same
computer, my IDL 8.5 results now look like my IDL 8.0 results.
>
> They are in the lib\tools\components subdirectory of each distribution.

```

>  
> The challenge, then, is to find the causal difference between the two versions. I don't know if it's necessary to investigate both classes. Only one of the two may be the problem.  
>  
> If you open two projects in your IDL Workbench, one based in the root of the IDL 8.0 lib directory and one in that of IDL 8.X, you can multi-select the two source files for each class then execute a "compare" via a right click in the Project Explorer window.  
>  
> Jim P.

Dear Jim,

thank you very much! At least now I have a workaround, and can recover the 'old' results with IDL 8.4, still taking advantages of all the other updates (of course not those for contour...). By the way, I put the two old class definition files in my working directory, since I couldn't find a way to re-compile them from my own procedure (.compile does not work inside procedures, and resolve\_routine does not accept absolute paths): is there a better way?

As for the debugging, unfortunately I always used IDL from the command-line, and have no familiarity at all with the graphical workbench, so I don't know how to specifically follow your suggestion to compare the two versions of classes. The files themselves are very different, so it would be for me very difficult to do a manual debugging.

Is some IDL developer from Harris actually reading these posts? I think this issue needs an official bugfix. It may well be that the new contour behaviour is more 'correct', but it must be ensured back-compatibility, and, most of all, there must be a way with the new contour to reproduce the kind of plots I did with the old IDL. A transparent contour plot with only some levels on top of another contour plot is a very typical plot in physics.

Thanks,

Stefano

---

Subject: Re: Contour: difference between IDL 8.0 and IDL 8.4  
Posted by [Markus Schmassmann](#) on Mon, 20 Feb 2017 11:41:48 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

On 02/20/2017 11:30 AM, burkina wrote:

> thank you very much! At least now I have a workaround, and can  
> recover the 'old' results with IDL 8.4, still taking advantages of all the other  
> updates (of course not those for contour...). By the way, I put the two  
> old class definition files in my working directory, since I couldn't  
> find a way to re-compile them from my own procedure (.compile does not  
> work inside procedures, and resolve\_routine does not accept absolute  
> paths): is there a better way?

I don't know whether there is a way without a separate directory, but it needn't be in your working directory. You can use:

```
if !PATH.contains('/path/to/files') then begin
  pathSplit=strsplit(!PATH,':',/extract)
  wp=where(pathSplit eq '/path/to/files',/null,complement=wpc)
  !PATH=strjoin(pathSplit[[wp,wpc]],':')
endif else !PATH='/path/to/files:'+!PATH
```

to make sure the files in '/path/to/files' are selected first.

I hope that helps, Markus

---

Subject: Re: Contour: difference between IDL 8.0 and IDL 8.4  
Posted by [chris\\_torrence@NOSPAM](mailto: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
```

---

---

Subject: Re: Contour: difference between IDL 8.0 and IDL 8.4

Posted by [burkina](#) on Wed, 22 Feb 2017 07:57:46 GMT

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

---

On Tuesday, 21 February 2017 17:25:34 UTC+1, Chris Torrence wrote:

> 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)

[color=blue]> colors = [colors, 255b+bytarr(1,dim[1])] ; change to RGBA[/color]

> 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)
[color=blue]> colors = [colors, 255b+bytarr(1,dim[1])] ; change to RGBA[/color]
> colors[3,3] = 0 ; make transparent
> c1.c_color = colors
```

Hi Chris,

thanks!

This is a great solution. However, it obviously overrides any 'transparency' set in the contour command. To preserve it, I have to define it for all the colors, like this:

```
colors[3,0]=150
colors[3,1]=150
colors[3,2]=150
```

With this other fix, I can actually get exactly the same result I have with IDL8.0.

May I suggest you a more 'elegant' fix for the next releases? You could consider a modification of the 'transparency' parameter: if it's a single number, it could be treated as it is now, as the transparency for all the colors. On the other hand, if it's an array with the same number of elements of c\_color, it could represent the alpha values for each color/contour. What do you think?

Thanks again,

Stefano

---

Subject: Re: Contour: difference between IDL 8.0 and IDL 8.4  
Posted by [chris\\_torrence@NOSPAM](mailto:chris_torrence@NOSPAM) on Thu, 23 Feb 2017 19:39:25 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

On Wednesday, February 22, 2017 at 12:57:52 AM UTC-7, burkina wrote:

> On Tuesday, 21 February 2017 17:25:34 UTC+1, Chris Torrence wrote:

>> 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)

[color=teal]>> colors = [colors, 255b+bytarr(1,dim[1])] ; change to RGBA[/color]

>> 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)

[color=teal]>> colors = [colors, 255b+bytarr(1,dim[1])] ; change to RGBA[/color]

>> colors[3,3] = 0 ; make transparent

>> c1.c\_color = colors

>

> Hi Chris,

>

> thanks!

>

> This is a great solution. However, it obviously overrides any 'transparency' set in the contour command. To preserve it, I have to define it for all the colors, like this:

>

> colors[3,0]=150

> colors[3,1]=150

> colors[3,2]=150

>

> With this other fix, I can actually get exactly the same result I have with IDL8.0.

>

> May I suggest you a more 'elegant' fix for the next releases? You could consider a modification of the 'transparency' parameter: if it's a single number, it could be treated as it is now, as the transparency for all the colors. On the other hand, if it's an array with the same number of

elements of `c_color`, it could represent the alpha values for each color/contour. What do you think?

>

> Thanks again,

>

> Stefano

That's a great idea. I'm not sure if it will make it into the next release, but I'll add it to the list!  
-Chris

---