

---

Subject: Re: legend not working with symcat  
Posted by [David Fanning](#) on Wed, 04 Nov 2009 09:41:33 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

maurya writes:

```
> The routine 'legend' is not working with the function 'symcat' for
> the following test program.
>
> ;;;;;;;;;;=====
> pro legend_test
>
> set_plot,'win'
>
> window,0,xs=600,ys=500
>
> n=12
> xx=findgen(200)
> for i=0,n-1 do begin
>   ii=i+1
>   yy=sin(xx!*dton*ii*0.2)
>   if (i eq 0) then plot,yy,yr=[0,1.1],psym=symcat(i),/yst $
>     else oplot,yy,psym=symcat(i),symsize=0.6,nsum=2
> endfor
>
> lines=indgen(n)
> item='p='+string(lines,format='(i2.2)')
> legend,item,psym=symcat(lines)
>
>
> stop
> end
> ;;;;;;;;;;=====
>
> Can anyone help.
```

Probably not. The problem is that SymCat is not "vectorized" in this way, and with good reason. Most of the symbols SymCat creates have to be built one-at-a-time with USERSYM. Then selected with the symbol index 8. So, if it was vectorized, you would certainly end up with a vector filled mostly with 8's.

```
IDL> symbols = IntArr(10)
IDL> FOR j=0,9 DO symbols[j] = SymCat(j + 10)
IDL> Print, symbols
 10  8  8  8  8  8  8  8  8  8
```

In other words, the symbol created by USERSYM is "ephemeral" and can only be used for a short time. Well, until the \*next\* symbol is defined.

The only way to fix this, as far as I can tell, would be to modify legend.pro to call SymCat whenever it creates a symbol.

On lines 425-427, in which PLOTS is being used to draw the symbol in the legend with PSYM, I changed "psym=psym[i]" to "psym=SymCat(psym[i]" and I changed your example program call from this:

```
legend, item, psym=SymCat(lines)
```

To this:

```
legend, item, psym=lines
```

And all works as you expect.

Wayne would have to confirm that this is the only place a change would have to be made. But I think he could add this to legend and gain access to 35 more symbols than he has now and not be worried by backward compatibility at all. ;-)

Cheers,

David

--

David Fanning, Ph.D.

Fanning Software Consulting, Inc.

Coyote's Guide to IDL Programming: <http://www.dfanning.com/>

Sepore ma de ni thui. ("Perhaps thou speakest truth.")

---

Subject: Re: legend not working with symcat  
Posted by [maurya](#) on Wed, 04 Nov 2009 14:21:41 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

On 4 Nov, 14:41, David Fanning <n...@dfanning.com> wrote:  
> maurya writes:  
>> The routine 'legend' is not working with the function 'symcat' for  
>> the following test program.  
>  
>> ;;;;;;;;;;=====

```

>> pro legend_test
>
>> set_plot,'win'
>
>> window,0,xs=600,ys=500
>
>> n=12
>> xx=findgen(200)
>> for i=0,n-1 do begin
>>   ii=i+1
>>   yy=sin(xx!*dior*ii*0.2)
>>   if (i eq 0) then plot,yy,yr=[0,1.1],psym=symcat(i),/yst $
>>       else oplot,yy,psym=symcat(i),symsize=0.6,nsum=2
>>   endfor
>
>> lines=indgen(n)
>> item='p'+string(lines,format='(i2.2)')
>> legend,item,psym=symcat(lines)
>
>> stop
>> end
>> ;;;;;;;;;;=====
>
>> Can anyone help.
>
> Probably not. The problem is that SymCat is not "vectorized"
> in this way, and with good reason. Most of the symbols SymCat
> creates have to be built one-at-a-time with USERSYM. Then
> selected with the symbol index 8. So, if it was vectorized,
> you would certainly end up with a vector filled mostly with
> 8's.
>
> IDL> symbols = IntArr(10)
> IDL> FOR j=0,9 DO symbols[j] = SymCat(j + 10)
> IDL> Print, symbols
>   10  8  8  8  8  8  8  8  8  8
>
> In other words, the symbol created by USERSYM is "ephemeral"
> and can only be used for a short time. Well, until the *next*
> symbol is defined.
>
> The only way to fix this, as far as I can tell, would be
> to modify legend.pro to call SymCat whenever is creates a
> symbol.
>
> On lines 425-427, in which PLOTS is being used
> to draw the symbol in the legend with PSYM, I changed
> "psym=psym[i]" to "psym=SymCat(psym[i])" and I changed

```

> your example program call from this:  
>  
> legend, item, psym=SymCat(lines)  
>  
> To this:  
>  
> legend, item, psym=lines  
>  
> And all works as you expect.  
>  
> Wayne would have to confirm that this is the only place a change  
> would have to be made. But I think he could add this to legend  
> and gain access to 35 more symbols then he has now and not be  
> worried by backward compatibility at all. ;-)  
>  
> Cheers,  
>  
> David  
>  
> --  
> David Fanning, Ph.D.  
> Fanning Software Consulting, Inc.  
> Coyote's Guide to IDL Programming:<http://www.dfanning.com/>  
> Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Thanks David,

I changed the "legend.pro" as you have suggested. Now, it is working for  $n \leq 10$  only. For larger values of  $n > 10$  it is giving following error:

PLOTS: PSYM (plotting symbol) out of range

When I commentated the line 223 (on\_error, 2), the program stop at the line 425-427 where the correction is made. Although, according to your "symcat" program it should work for 46 symbols. However, I need only for  $n \leq 12$ .

RAM

---

Subject: Re: legend not working with symcat  
Posted by [David Fanning](#) on Wed, 04 Nov 2009 14:35:56 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

maurya writes:

> I changed the "legend.pro" as you have suggested. Now, it is working

> for n<=10 only. For larger values of n>10 it is giving following  
> error:  
>  
> PLOTS: PSYM (plotting symbol) out of range  
>  
> When I commentated the line 223 (on\_error, 2), the program stop at the  
> line 425-427 where the correction is made. Although, according to your  
> "symcat" program it should work for 46 symbols. However, I need only  
> for n<=12.

Humm. I'm not sure you made the change successfully. :-)

There was a parenthesis missing in my example. Did you fix that?  
Can I see the line you fixed?

Cheers,

David

--

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

---

Subject: Re: legend not working with symcat  
Posted by [wlandsman](#) on Wed, 04 Nov 2009 19:28:38 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

On Nov 4, 9:21 am, maurya <ramaury...@gmail.com> wrote:  
> I changed the "legend.pro" as you have suggested. Now, it is working  
> for n<=10 only. For larger values of n>10 it is giving following  
> error:  
>  
> PLOTS: PSYM (plotting symbol) out of range

I suspect that it works for n>10 but is failing for n=10. That is because the IDL definition of PSYM=10 (which SYMCAT() follows) does not really define a symbol, but tells IDL to plot in histogram mode. It \*cannot\* be used with PLOTS, e.g.

```
IDL> plot,indgen(10),indgen(10)
IDL> plots,5,5,psym=symcat(10)
% PLOTS: PSYM (plotting symbol) out of range
```

So it is probably best to not use PSYM=10. (What symbol should one

use in a legend to distinguish a line draw in histogram mode, versus one connecting the points?)

--Wayne

---

---

Subject: Re: legend not working with symcat  
Posted by [wlandsman](#) on Wed, 04 Nov 2009 21:59:28 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

>  
> Wayne would have to confirm that this is the only place a change  
> would have to be made. But I think he could add this to legend  
> and gain access to 35 more symbols than he has now and not be  
> worried by backward compatibility at all. ;-)

That would be true if all users had symcat.pro in their !path. But in case there are some inexperienced users who don't have the Coyote library installed yet, I've slightly modified your update to legend.pro to only call symcat.pro for non-standard PSYM values.

<http://idlastro.gsfc.nasa.gov/ftp/pro/plot/legend.pro>

--Wayne

---

---

Subject: Re: legend not working with symcat  
Posted by [David Fanning](#) on Wed, 04 Nov 2009 23:19:08 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Wayne Landsman writes:

> That would be true if all users had symcat.pro in their !path. But  
> in case there are some inexperienced users who don't have the Coyote  
> library installed yet, I've slightly modified your update to  
> legend.pro to only call symcat.pro for non-standard PSYM values.  
>  
> <http://idlastro.gsfc.nasa.gov/ftp/pro/plot/legend.pro>

The problem now is that LEGEND won't compile \*unless\* you have SYMCAT in your path. :-)

Of course, that might force users to download the Coyote Library, a Very Good Thing, if you ask me, but...still.

I'm always in a state of confusion when I want to use a NASA or JHUAPL routine in a Coyote Library program. What do I do? Include the routine

\*with\* my routines. (Don't like it, since this is a maintenance nightmare.)  
Add the routine at the front of my routine. (Well, at least I can guarantee it \*works\*.) Or, rename the routine and put it in my file. (This is what I usually do, along with the copyright notice, etc.)

In this case, I think calling the routine `legend_symcat` and sticking it in front of `legend` makes sense. But then you can't draw the damn plot with the SYMCAT symbols unless you compile `legend` first. I don't know. That's why I suggested you should handle it. :-)

Cheers,

David

P.S. Why don't you write this as an object, like it should be, and then we can just subclass off it. ;-)

---

Subject: Re: legend not working with symcat  
Posted by [wlandsman](#) on Thu, 05 Nov 2009 00:48:24 GMT  
[View Forum Message](#) <> [Reply to Message](#)

On Nov 4, 6:19 pm, David Fanning <da...@dfanning.com> wrote:  
> Wayne Landsman writes:  
>> That would be true if all users had `symcat.pro` in their `!path`. But  
>> in case there are some inexperienced users who don't have the Coyote  
>> library installed yet, I've slightly modified your update to  
>> `legend.pro` to only call `symcat.pro` for non-standard PSYM values.  
>  
>> <http://idlastro.gsfc.nasa.gov/ftp/pro/plot/legend.pro>  
>  
> The problem now is that LEGEND won't compile \*unless\* you  
> have SYMCAT in your path. :-)

Well, I don't think compilation is a problem, e.g. the program

```
pro test,dum
if keyword_set(dum) then a = some_program(3) else print,'AOK'
return
end
```

will run fine so long as one doesn't try to access `Some_program()`.

```
IDL> test
% Compiled module: TEST.
```

AOK

```
IDL> test,1
```

```
% Variable is undefined: SOME_PROGRAM.
```

```
% Execution halted at: TEST          2 /Volumes/Apps_and_Docs/  
landsman/codev/test.pro
```

(If I had added a `compile_opt idl2` at the beginning, it would also know that `SOME_PROGRAM` is an underlined function/procedure rather than a variable.)

Similarly, `legend.pro` should now run fine without `symcat.pro` in one's !path, so long as one doesn't try to use a value of `PSYM>=11`.

But I agree the more general question is difficult. Currently, I keep JHUAPL routines I need in a separate directory, and tell users to delete the directory if they have the entire JHUAPL library installed.

--Wayne

---

Subject: Re: legend not working with symcat  
Posted by [maurya](#) on Thu, 05 Nov 2009 09:24:18 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

On 5 Nov, 05:48, wlandsman <wlands...@gmail.com> wrote:

> On Nov 4, 6:19 pm, David Fanning <da...@dfanning.com> wrote:

>

>> Wayne Landsman writes:

>>> That would be true if all users had `symcat.pro` in their !path. But  
>>> in case there are some inexperienced users who don't have the Coyote  
>>> library installed yet, I've slightly modified your update to  
>>> `legend.pro` to only call `symcat.pro` for non-standard `PSYM` values.

>

>>> <http://idlastro.gsfc.nasa.gov/ftp/pro/plot/legend.pro>

>

>> The problem now is that `LEGEND` won't compile \*unless\* you  
>> have `SYMCAT` in your path. :-(

>

> Well, I don't think compilation is a problem, e.g. the program

>

```
> pro test,dum
```

```
> if keyword_set(dum) then a = some_program(3) else print,'AOK'
```

```
> return
```

```
> end
```

>

> will run fine so long as one doesn't try to access `Some_program()`.

>

```
> IDL> test
```

```

> % Compiled module: TEST.
> AOK
> IDL> test,1
> % Variable is undefined: SOME_PROGRAM.
> % Execution halted at: TEST          2 /Volumes/Apps_and_Docs/
> landsman/codev/test.pro
>
> (If I had added a compile_opt idl2 at the beginning, it would also
> know that SOME_PROGRAM is an underlined function/procedure rather than
> a variable.)
> Similarly, legend.pro should now run fine without symcat.pro in one's !
> path, so long as one doesn't try to use a value of PSYM>=11.
>
> But I agree the more general question is difficult.  Currently, I
> keep JHUAPL routines I need in a separate directory, and tell users to
> delete the directory if they have the entire JHUAPL library
> installed.
>
> --Wayne

```

Thanks David and Wayne,

Actually, n=10 was not the symbol number in my previous post. It is maximum number of increment in the for loop. The suggestions given by David is most useful. Now, the "legend.pro" is working for all the symbols defined in the "symcat.pro" except 10. You can see one of the following test. Here, I am using new "legend.pro" program given by Wayne (<http://idlastro.gsfc.nasa.gov/ftp/pro/plot/legend.pro>). I have removed the symbol number 10.

```

-----
pro legend_test

set_plot,'win'

window,0,xs=600,ys=800

n=47
xx=findgen(200)
for i=0,n-1 do begin
  ii=i+1
  yy=sin(xx!*dior*ii*0.2)
  if (i eq 0) then plot,yy,yr=[0,1.1],psym=symcat(i),/yst $
    else oplot,yy,psym=symcat(i),symsize=0.6,nsum=2
endfor

lines=indgen(n)
lines(where(lines eq 10)) = 1

```

```
item='p'+string(lines,format='(i2.2)')
```

```
print,lines
```

```
legend, item, psym=lines
```

```
stop
```

```
end
```

```
-----
```

Cheers

---