Subject: Re: External c calls broken in IDL 5.5
Posted by Bob[1] on Tue, 03 Sep 2002 21:16:59 GMT
View Forum Message <> Reply to Message

Actually I was running Tru64 Unix 4.0 before (not 5.0 as I said on the
provious message).

Bob

> They worked fine on IDL 5.3 and Tru64 Unix
> 5.0 but now that I've upgraded to IDL 5.5 and Tru64 Unix 5.1A they will
> no longer work (they core dump).

Subject: Re: External c calls broken in IDL 5.5
Posted by Jaco van Gorkom on Tue, 03 Sep 2002 21:51:45 GMT
View Forum Message <> Reply to Message

"Bob" <b_o_b_1962@nospam.yahoo.com> wrote in message
news:3D752116.9016BC50@nospam.yahoo.com...
> I have some dlm's that I made using DLMFORM (see
> http://www.astro.uio.no/~steinhh/idl/dlmform.html). They call fortran
> code through a c wrapper. They worked fine on IDL 5.3 and Tru64 Unix
> 5.0 but now that I've upgraded to IDL 5.5 and Tru64 Unix 5.1A they will
> no longer work (they core dump). I've tried remaking them and that
> doesn't help.

Have you tried remaking them using the "export.h" that came with IDL5.5?
Something in there changed in 5.5, I believe the IDL string definition.
Googling the group for export.h 5.5 should turn up some info on this issue.

Hope this helps,
Jaco

Subject: Re: External c calls broken in IDL 5.5
Posted by Bob[1] on Tue, 03 Sep 2002 22:38:22 GMT
View Forum Message <> Reply to Message

Jaco van Gorkom wrote:

> Have you tried remaking them using the "export.h" that came with IDL5.5?
> Something in there changed in 5.5, I believe the IDL string definition.
> Googling the group for export.h 5.5 should turn up some info on this issue.

Yes, I remade them with the 5.5 version of export.h and that didn't help.

I just tried my old dlm files with IDL 5.3 and they still work so the problem is in using IDL 5.5. It even appears that the remade dlm work with IDL 5.3 as well (which surprises me as they use the 5.5 version of export.h). However, I've recently incorporated some 5.5 features in my code and do not want go back.

Bob

Subject: Re: External c calls broken in IDL 5.5
Posted by Nigel Wade on Wed, 04 Sep 2002 11:22:21 GMT
View Forum Message <> Reply to Message

Bob wrote:

> I have some dlm's that I made using DLMFORM (see
> http://www.astro.uio.no/~steinhh/idl/dlmform.html). They call fortran
> code through a c wrapper. They worked fine on IDL 5.3 and Tru64 Unix
> 5.0 but now that I've upgraded to IDL 5.5 and Tru64 Unix 5.1A they will
> no longer work (they core dump). I've tried remaking them and that
> doesn't help. Has anyone else had this problem?
>
> So I've tried a really simple example (below) using call_external and
> just using c code and even the simple code that follows doesn't work
> (it should print out "n = 25" and "x = 3.14159"). Anyone have any
> ideas.
>
> And last, is there a newer version of dlmform. I really like this
> routine but it is using some obsolete functions.
>
> Thanks,
> Bob
>
> In test.c
> #include <stdio.h>
> #include "export.h"
>
> void test(int argc, IDL_VPTR argv[])
> {
> IDL_LONG n = argv[0]->value.l;
> printf("n = %i\n", n);
> float x = argv[1]->value.f;
> printf("x = %f\n", x);
> }
>
> I complile and link with:
> cc -c -I/usr/local/rsi/idl/external -pthread test.c
> ld -S -expect_unresolved '*' -shared -all -hidden -o test.so test.o
>

> then trying it:
> $ idl
> IDL Version 5.5, Compaq Tru64 (OSF alpha). (c) 2001, Research Systems,
> Inc.
> IDL> n = 25l
> IDL> x=!PI
> IDL> help,n,x
> N LONG = 25
> X FLOAT = 3.14159
> IDL> dum = call_external('./test.so', 'test',n,x)
> n = 1074012760
> x = 0.000000
> IDL>

The interface you have specified is for a DLM/LINKIMAGE function, but you
have tried to call it with call_external (which has a completely different
execution method).

Try instead:

IDL> linkimage,'test','./test.so',0,'test'
IDL> test,n,x

--
Nigel Wade, System Administrator, Space Plasma Physics Group,
University of Leicester, Leicester, LE1 7RH, UK
E-mail : nmw@ion.le.ac.uk
Phone : +44 (0)116 2523568, Fax : +44 (0)116 2523555

Subject: Re: External c calls broken in IDL 5.5
Posted by Bob[1] on Wed, 04 Sep 2002 15:43:07 GMT
View Forum Message <> Reply to Message

Nigel Wade wrote:

> The interface you have specified is for a DLM/LINKIMAGE function, but you
> have tried to call it with call_external (which has a completely different
> execution method).
>
> Try instead:
>
> IDL> linkimage,'test','./test.so',0,'test'
> IDL> test,n,x

OK, thanks. I didn't realize that they were different. But this doesn't
change the fact that my old DLM's no longer work in IDL 5.5. Am I the only

one with this problem.

Subject: Re: External c calls broken in IDL 5.5
Posted by Stein Vidar Hagfors H[1] on Wed, 04 Sep 2002 19:05:45 GMT
View Forum Message <> Reply to Message

Bob <b_o_b_1962@nospam.yahoo.com> writes:

> Nigel Wade wrote:
>
>> The interface you have specified is for a DLM/LINKIMAGE function, but you
>> have tried to call it with call_external (which has a completely different
>> execution method).
>>
>> Try instead:
>>
>> IDL> linkimage,'test','./test.so',0,'test'
>> IDL> test,n,x
>
> OK, thanks. I didn't realize that they were different. But this doesn't
> change the fact that my old DLM's no longer work in IDL 5.5. Am I the only
> one with this problem.

Yo! Someone who's seen the light and is using DLMFORM, cool!

Now, to deal with the DLM/LINKIMAGE versus CALL_EXTERNAL first, Nigel is quite correct; however, if you make sure to set IDL_DLM_PATH to include the directory where your test.so file (and the "test.dlm" file produced by dlmform) is residing (before starting IDL) you should not have to use the linkimage procedure at all. Simply drop that line, and say:

IDL> test,n,x

That's it. The information provided in the linkimage statement is provided in the .dlm file, which IDL scans upon startup (as long as it has the IDL_DLM_PATH set to find it!), and in the IDL_Load() procedure in the C file, which is executed on loading the module. Try help,/dlm for a list of all the DLM's that it has found (a lot of IDL's packages, like mpeg/hdf/etc, are implemented this way nowadays). Note that the name of the procedure "advertised" in the .dlm file and provided in the IDL_Load() registration routine is based on the fortran function name plus prefix "F77_". If you edit the name of the function advertised in the C file (e.g. taking away the "F77_" prefix), you should also do so in the DLM file for the automated loading to work (i.e. without the linkimage statement).

Now, for the crashing, I bet your fortran routine uses complex numbers?? (Okay, I don't bet a lot, but this is my only "known" suspect). Guess what - IDL changed the spec of the IDL_CvtComplex call in IDL 5.4 and told no-one about it (in the edg.pdf at least, at the time). So, I suspect this is your problem (I haven't tested dlmform on 5.5, since I don't run that here.. be my guest).

Now, I did make a workaround involving a C preprocessor #if statement some time ago, I just hadn't uploaded it to the web page in Oslo. Beware that I also made some changes between the first "publication" and the current version to adjust to perl v 5. The #!/usr/local/bin/perl line at the beginning may have changed from the version you have also.

Pick up the new version now from http://www.astro.uio.no/~steinhh/idl/dlmform.html and tell me if it fixes your problem (no support promised if it doesn't, though!).

> And last, is there a newer version of dlmform. I really like this
> routine but it is using some obsolete functions.

Thanks. They weren't obsolete in 1999 ;-) That's when this project gave me a "tennis elbow" that hasn't left me since (it should be called a "perl elbow", from typing all those dollar signs using the pinkie and the index finger on the same hand!).

I know that there are a few anachronisms that should be updated in the code for dlmform... Supporting the new types like 64 bit types is one, but unless I see or hear of a need for it (I'm assuming this is mostly used for legacy code in fortran, and it wouldn't have those as inputs, would it?). Here's a deal: If you send me a list of the obsolete functions that you'd like to see replaced (with the replacements and, preferably, the IDL version in which the change was made!!), I'll see what I can do!

--

---------- ----------
Stein Vidar Hagfors Haugan
ESA SOHO SOC/European Space Agency Science Operations Coordinator for SOHO

NASA Goddard Space Flight Center, Email: shaugan@esa.nascom.nasa.gov
Mail Code 682.3, Bld. 26, Room G-1, Tel.: 1-301-286-9028/240-354-6066
Greenbelt, Maryland 20771, USA. Fax: 1-301-286-0264
---------- ----------

Subject: Re: External c calls broken in IDL 5.5

Posted by Bob[1] on Wed, 04 Sep 2002 21:01:39 GMT
View Forum Message <> Reply to Message

OK, it turned out to be something stupid on my part (doesn't it always).
I had an old extern.h file in my local directory from when I did it the
first time (a few years ago) and there were enough files in there that I
didn't notice it. Now all works fine (and I guess I've had the benefit of
learning lots more about IDL external calls).

Subject: Re: External c calls broken in IDL 5.5
Posted by Bob[1] on Wed, 04 Sep 2002 21:01:42 GMT
View Forum Message <> Reply to Message

Stein Vidar Hagfors Haugan wrote:

> Now, for the crashing, I bet your fortran routine uses complex
> numbers?? (Okay, I don't bet a lot, but this is my only "known"
> suspect). Guess what - IDL changed the spec of the IDL_CvtComplex
> call in IDL 5.4 and told no-one about it (in the edg.pdf at least, at
> the time). So, I suspect this is your problem (I haven't tested
> dlmform on 5.5, since I don't run that here.. be my guest).

No, it was totally my fault (see my other message).

> I know that there are a few anachronisms that should be updated in the
> code for dlmform... Supporting the new types like 64 bit types is one,
> but unless I see or hear of a need for it (I'm assuming this is mostly
> used for legacy code in fortran, and it wouldn't have those as inputs,
> would it?). Here's a deal: If you send me a list of the obsolete
> functions that you'd like to see replaced (with the replacements and,
> preferably, the IDL version in which the change was made!!), I'll see
> what I can do!

Well, the calls to IDL_AddSystemRoutine in IDL_Load is obsolete and should be
replaced with
IDL_SysRtnAdd which uses IDL_SYSFUN_DEF2 structure instead of the IDL_SYSFUN_DEF
structure (I think this changed in IDL 5.4 but am not sure). However, it does
still work the way you have it.

A bigger problem I have is that the array dimensions for output arrays are
usually off by one in the c-code generated by dlmform when there is an array in
the fortran code. For example the fortran program:

```
subroutine test(x,y,n)
integer n
real x(n), y(n)
do i=1,n
```

```
      y(i) = x(i)*x(i)
    enddo
    end
```

The c-code (test.c) has the following block where the variable y is defined:

```
 in = 0;          /* Y_ : REAL : (N) :  */
 if (in) {
  } else { /* Output */
   IDL_EXCLUDE_EXPR(Y_); /* Output cannot be expression */
   ndim = 1;
   dim[0] = N_->value.l+1 /*???*/;
   IDL_StoreScalarZero(Y_,IDL_TYP_FLOAT);  /* Free resources */
    IDL_MakeTempArray(IDL_TYP_FLOAT,ndim,dim,IDL_ARR_INI_ZERO,&a mp;tmp);
   IDL_VarCopy(tmp,Y_);
 }
```

The dim[0]=N_->value.l+1 should be dim[0]=n_value.l. I've gotten in the habit of always changing these but it would be nice I didn't have to.

And last, I'll ask you this since you seem to know you way around. Is it possible to catch errors in the fortran code so that one ends up at the IDL prompt instead of core dumping IDL?

Thanks.

Bob

Subject: New version of DLMFORM [Re: External c calls broken in IDL 5.5]
Posted by Stein Vidar Hagfors H[1] on Tue, 10 Sep 2002 19:11:24 GMT
View Forum Message <> Reply to Message

Bob <b_o_b_1962@nospam.yahoo.com> writes:

[....]

> Well, the calls to IDL_AddSystemRoutine in IDL_Load is obsolete and
> should be replaced with IDL_SysRtnAdd which uses IDL_SYSFUN_DEF2
> structure instead of the IDL_SYSFUN_DEF structure (I think this
> changed in IDL 5.4 but am not sure). However, it does still work
> the way you have it.

The new version corrects this, using IDL_SysRtnAdd for version > 5.3 (which, according to my edg.pdf is the version that obsoleted the other one).

> A bigger problem I have is that the array dimensions for output

> arrays are usually off by one in the c-code generated by dlmform
> when there is an array in the fortran code.

Fixed...

[...]
```
>       subroutine test(x,y,n)
>       integer n
>       real x(n), y(n)
>       do i=1,n
>          y(i) = x(i)*x(i)
>       enddo
>       end
```

However, I'd like to comment on this example (although it is obviously a trivialized one), that the same [better!] functionality can be written as

```
real function test(x)
real x
test = x*x
end
```

Instead of saying e.g.

```
TEST,X,Y,N_ELEMENTS(X)
```

you can now (thanks to the automatic arrayification of scalar routines) say

```
Y = TEST(X)
```

If you prefer procedures to functions, a scalar "subroutine test(x,y)" would do the trick, so you could say "TEST,X,Y".

In more complex, general cases, when (some) input and output arrays have the same dimensionality, it's almost trivial to use the input dimensionality directly, editing out the extra variable being passed (N).

> And last, I'll ask you this since you seem to know you way around. Is it
> possible to catch errors in the fortran code so that one ends up at the IDL
> prompt instead of core dumping IDL?

Not unless you can tell me how to catch those errors in fortran! (I don't believe fortran has exceptions...).

I've also done some other changes to the program, among other things

it will no longer create [core-dumping] functions trying to return
complex values (instead, a warning & no function). Pick it up at

http://www.astro.uio.no/~steinhh/idl/dlmform.html

--

Stein Vidar Hagfors Haugan
ESA SOHO SOC/European Space Agency Science Operations Coordinator for SOHO

NASA Goddard Space Flight Center, Email: shaugan@esa.nascom.nasa.gov
Mail Code 682.3, Bld. 26, Room G-1, Tel.: 1-301-286-9028/240-354-6066
Greenbelt, Maryland 20771, USA. Fax: 1-301-286-0264

Subject: Re: New version of DLMFORM [Re: External c calls broken in IDL 5.5]
Posted by Bob[1] on Thu, 12 Sep 2002 21:08:19 GMT
View Forum Message <> Reply to Message

Thanks for the fixes.

Bob

Stein Vidar Hagfors Haugan wrote:

> Pick it up at
>
> http://www.astro.uio.no/~steinhh/idl/dlmform.html
>
