
Subject: IDL code with fortran subroutine and C linker on OSX

Posted by [laurel](#) on Tue, 28 Jun 2005 20:20:10 GMT

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

In nutshell, my problem is with linking all of my different language codes together. One of the languages seem to thing there should be a leading underscore on a subroutine name, and I just don't know why. Here's a more detailed explanation of what is going on.

I have an IDL code, and a fortran subroutine called mainsub. I have a C linker as a go-between from IDL and fortran called field_c. They both compile fine on their own, but don't seem to want to be linked. Here is my set of compiling commands so far:

```
f90 -c mainsub.f
gcc -c field_c.c
(these both work fine)
gcc -bundle -flat_namespace -fno-leading-underscore -o field_c.so
field_c.o mainsub.o KT_2003_sub.o
(KT is another fortran subroutine called in mainsub)
```

My problem is that when I enter the last line I get:

"ld: Undefined symbols:

```
__mainsub_
__A_FWF
__CLOSE...."
```

and several more.

In field_c.c, I call "mainsub_" because I think that fortran adds the extra "_" as a suffix. I have tried with and without the suffix, and with and without a prefix. I can't call the subroutine "__mainsub" in the fortran code because subroutines can't start with underscores. I really don't know where these extra underscores are coming from, and what the subroutines are called in the .o files.

So, does anyone know where I could find the symbol tables for my .o files (both C and fortran?). Or, even better, has anyone done this on a Mac OSX, ran into this problem, and been able to fix it. Any help would be appreciated as I think I might be nearing the end of my creativity on the matter.

Thanks so much,
Laurel

Subject: Re: IDL code with fortran subroutine and C linker on OSX

laurel wrote:

- > In nutshell, my problem is with linking all of my different language
- > codes together. One of the languages seem to thing there should be a
- > leading underscore on a subroutine name, and I just don't know why.
- > Here's a more detailed explanation of what is going on.
- >

Typically FORTRAN compilers add underscores to external names to differentiate them from identically named externals in C. This allows you to have C code with an external identifier of MYFUNC call a FORTRAN function called MYFUNC without any conflict. The C code would call the external name `_MYFUNC` or `_MYFUNC_` or `MYFUNC_` depending on where the FORTRAN compiler adds underscores.

- > I have an IDL code, and a fortran subroutine called mainsub. I have a C
- > linker as a go-between from IDL and fortran called field_c. They both
- > compile fine on their own, but don't seem to want to be linked. Here is
- > my set of compiling commands so far:
- >
- > f90 -c mainsub.f
- > gcc -c field_c.c
- > (these both work fine)
- > gcc -bundle -flat_namespace -fno-leading-underscore -o field_c.so
- > field_c.o mainsub.o KT_2003_sub.o
- > (KT is another fortran subroutine called in mainsub)

Normally you would use the FORTRAN compiler to the linking. That way it knows which FORTRAN libraries to link in. What's the purpose of the `-fno-leading-underscore` flag? Does it do anything during linkage?

- >
- > My problem is that when I enter the last line I get:
- > "ld: Undefined symbols:
- > `_mainsub_`
- > `__A_FWF`
- > `__CLOSE....`"
- >
- > and several more.
- >
- > In field_c.c, I call "mainsub_" because I think that fortran adds the
- > extra "_" as a suffix. I have tried with and without the suffix, and
- > with and without a prefix. I can't call the subroutine "_mainsub" in
- > the fortran code because subroutines can't start with underscores. I
- > really don't know where these extra underscores are coming from, and
- > what the subroutines are called in the .o files.

If looks like FORTRAN is adding both a leading and trailing underscore. Try calling `_mainsub_`. I would think that `__A_FWF` and `__CLOSE` are FORTRAN intrinsics which are in whatever FORTRAN library the FORTRAN compiler would normally link in. Try linking with `f90` rather than `gcc`.

>
> So, does anyone know where I could find the symbol tables for my .o
> files (both C and fortran?). Or, even better, has anyone done this on a
> Mac OSX, ran into this problem, and been able to fix it. Any help would
> be appreciated as I think I might be nearing the end of my creativity
> on the matter.
>
> Thanks so much,
> Laurel

The `nm` command should show you the complete list of external names in a object file. I've never used OSX, or OpenBSD for that matter. But it's not too disimilar from Linux, and the `gcc` parts should be the same.

--

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 2523548, Fax : +44 (0)116 2523555

Subject: Re: IDL code with fortran subroutine and C linker on OSX

Posted by [Haje Korth](#) on Fri, 01 Jul 2005 12:39:43 GMT

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

Besides the underscore that Nigel mentions, typical mistakes are spelling of the function; they are CASE SENSITIVE.

Haje

"Nigel Wade" <nmw@ion.le.ac.uk> wrote in message
news:d9tro2\$6gd\$1@south.jnrs.ja.net...

> laurel wrote:

>

>> In nutshell, my problem is with linking all of my different language
>> codes together. One of the languages seem to thing there should be a
>> leading underscore on a subroutine name, and I just don't know why.
>> Here's a more detailed explanation of what is going on.

>>

>

> Typically FORTRAN compilers add underscores to external names to

> differentiate them from identically named externals in C. This allows you
 > to have C code with an external identifier of MYFUNC call a FORTRAN
 > function
 > called MYFUNC without any conflict. The C code would call the external
 > name
 > _MYFUNC or _MYFUNC_ or MYFUNC_ depending on where the FORTRAN compiler
 > adds
 > underscores.
 >
 >> I have an IDL code, and a fortran subroutine called mainsub. I have a C
 >> linker as a go-between from IDL and fortran called field_c. They both
 >> compile fine on their own, but don't seem to want to be linked. Here is
 >> my set of compiling commands so far:
 >>
 >> f90 -c mainsub.f
 >> gcc -c field_c.c
 >> (these both work fine)
 >> gcc -bundle -flat_namespace -fno-leading-underscore -o field_c.so
 >> field_c.o mainsub.o KT_2003_sub.o
 >> (KT is another fortran subroutine called in mainsub)
 >
 > Normally you would use the FORTRAN compiler to the linking. That way it
 > knows which FORTRAN libraries to link in. What's the purpose of the
 > -fno-leading-underscore flag? Does it do anything during linkage?
 >
 >>
 >> My problem is that when I enter the last line I get:
 >> "ld: Undefined symbols:
 >> _mainsub_
 >> __A_FWF
 >> __CLOSE...."
 >>
 >> and several more.
 >>
 >> In field_c.c, I call "mainsub_" because I think that fortran adds the
 >> extra "_" as a suffix. I have tried with and without the suffix, and
 >> with and without a prefix. I can't call the subroutine "_mainsub" in
 >> the fortran code because subroutines can't start with underscores. I
 >> really don't know where these extra underscores are coming from, and
 >> what the subroutines are called in the .o files.
 >
 > It looks like FORTRAN is adding both a leading and trailing underscore.
 > Try
 > calling _mainsub_. I would think that __A_FWF and __CLOSE are FORTRAN
 > intrinsics which are in whatever FORTRAN library the FORTRAN compiler
 > would
 > normally link in. Try linking with f90 rather than gcc.
 >

>>
>> So, does anyone know where I could find the symbol tables for my .o
>> files (both C and fortran?). Or, even better, has anyone done this on a
>> Mac OSX, ran into this problem, and been able to fix it. Any help would
>> be appreciated as I think I might be nearing the end of my creativity
>> on the matter.
>>
>> Thanks so much,
>> Laurel
>
> The nm command should show you the complete list of external names in a
> object file. I've never used OSX, or OpenBSD for that matter. But it's not
> too dissimilar from Linux, and the gcc parts should be the same.
>
> --
> 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 2523548, Fax : +44 (0)116 2523555
