Subject: Re: linkimage query
Posted by Nigel Wade on Thu, 15 Mar 2001 16:18:18 GMT
View Forum Message <> Reply to Message

> hi,
> I realise many people don't use Linkimage, but maybe someone might know
> what's happening here. Basically I've written some code that links to
IDL
> with LinkImage, it seems to work fine in most circumstances, but the fact
> I've a problem in one case makes me wonder if I've just been lucky in all
> the other cases.

> The code is used on two platforms, gcc/Linux/x86 and cc/DEC
> unix/alpha. What compiler flags should be used when compiling a binary
to
> be used with linkimage? I've been using:

> -shared for gcc

That's fine.

> -std -shared for cc

That's fine also, the -std just means apply fairly strict ISO compliance.

> Are these o.k. or should different flags be used? The problem I
mentioned
> above is a segmentation fault that occurs with only one of the alpha
boxes
> I use (I think it might have a different version of the alpha chip, but
> the binary should work on all alphas). The reason I asked about the
> compiler flags is that cc produces the following warning but gcc doesn't:

> ld:
> Warning: Unresolved:
> sqrt
> IDL_VarEnsureSimple
> IDL_Message
> exp
> log

The Digital linker warns about unresolved externals when creating shared
objects (DSO). Most linkers don't bother to warn because they assume that
those externals will be resolved when an executable is linked to the
object.
For dynamic loading the externals must be resolved in either the DSO
by linking to other libraries, or by the executable which loads the DSO.

> ...etc, even thought I've #include <math.h> and export.h.

Header files only supply function declarations, not definitions. The
actual code for the functions is usually provided by external libraries.
In the case of sqrt and friends you could link to libm (-lm) but that's
not necessary as IDL has already been linked to it and these externals
will be resolved at run time when your DSO is loaded by IDL. Similarly
the IDL_... functions can be resolved by linking explicitly to libidl
but IDL is already linked to it.

> any info would be much appreciated,
> Rich

None of the above should cause a segmentation fault. If you still have
unresolved externals at run time the DSO will fail to load and you
will be told by IDL that it failed.

Most likely the problem is due to the Alpha being a 64bit platform, and
on
that platform the C type 'long' is 64 bits. Check your C code for use of
long's, particularly if accessing IDL_LONG variables which will be 32bit.

--

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: linkimage query
Posted by src on Fri, 16 Mar 2001 16:24:22 GMT
View Forum Message <> Reply to Message

On Thu, 15 Mar 2001, Nigel Wade wrote:
>
> Most likely the problem is due to the Alpha being a 64bit platform, and
> on
> that platform the C type 'long' is 64 bits. Check your C code for use of
> long's, particularly if accessing IDL_LONG variables which will be 32bit.
>
But I think export.h defines all IDL_LONG's to be 32 bit to avoid
confusion, but I might have a go replacing IDL_LONG's with IDL_LONG64's to
see if it helps. What I did find was that the segmentation fault occurs
on alphas running True64 (version 5? of OSF) but NOT OSF4 (version 4?),
it's not a chip thing like I originally thought. I wonder what changed
between version 4 and version 5 of OSF?

cheers,
R