
Subject: problem converting FORTRAN to IDL
Posted by [brooker](#) on Thu, 13 Apr 1995 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

This is an observation I have just made about IDL.

When you compile a FORTRAN program, you can specify G_floating implementations of REAL*8. This extends the range of numbers to +/-0.56D308. (For default D_floating, the maximum number allowed is 0.29D38.)

On the other hand, IDL has no option for the larger G_floating numbers. This makes for problems when you convert a "G_floating REAL*8 " FORTRAN program to IDL.

Peter Brooker

Subject: Re: problem converting FORTRAN to IDL
Posted by [rivers](#) on Sat, 15 Apr 1995 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

In article <3mlnpd\$Qnd@reznor.larc.nasa.gov>, zawodny@arbd0.larc.nasa.gov (Joseph M Zawodny) writes:

> In article <D6zHn2.LyF@ireq.hydro.qc.ca> brooker@toka.ireq-ccfm.hydro.qc.ca writes:

>> This is an observation I have just made about IDL.

>>

>> When you compile a FORTRAN program, you can specify G_floating
>> implementations of REAL*8. This extends the range of numbers to +/-0.56D308.
>> (For default D_floating, the maximum number allowed is 0.29D38.)

>>

>> On the other hand, IDL has no option for the larger G_floating numbers. This
>> makes for problems when you convert a "G_floating REAL*8 " FORTRAN program to
>> IDL.

>>

>> Peter Brooker

>

> Gee, maybe I do not understand your problem, but I did this quick test.

>

> IDL> a=.5d308

> IDL> print,a

> 5.0000000e+307

>

> Therefor you should be able to use the IDL DOUBLE to implement FORTRAN

> G_floating calculations.

I belive the original post was probably referring to a DEC Alpha machine. On the Alpha one can compile double precision code to be D_FLOAT, G_FLOAT or

IEEE_FLOAT. The default for the DEC C compiler is G_FLOAT. IDL is clearly compiled with D_FLOAT, presumably so that the Alpha version is compatible with old VAX binary files, where D_FLOAT is the default. It would be possible and perhaps nice if RSI would provide 3 different versions (D_FLOAT, G_FLOAT and IEEE_FLOAT) of IDL for the Alpha platform. If you use CALL_EXTERNAL on the Alpha you need to make sure any routines you call are compiled with the same floating point format used for IDL.

Mark Rivers	(312) 702-2279 (office)
CARS	(312) 702-9951 (secretary)
Univ. of Chicago	(312) 702-5454 (FAX)
5640 S. Ellis Ave.	(708) 922-0499 (home)
Chicago, IL 60637	rivers@cars3.uchicago.edu (Internet)

Subject: Re: problem converting FORTRAN to IDL
Posted by [thompson](#) on Mon, 17 Apr 1995 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

rivers@cars3.uchicago.edu (Mark Rivers) writes:

> In article <3mlnpd\$wnd@reznor.larc.nasa.gov>, zawodny@arbd0.larc.nasa.gov (Joseph M Zawodny) writes:

>> In article <D6zHn2.LyF@ireq.hydro.qc.ca> brooker@toka.ireq-ccfm.hydro.qc.ca writes:

>>> This is an observation I have just made about IDL.

>>>

>>> When you compile a FORTRAN program, you can specify G_floating

>>> implementations of REAL*8. This extends the range of numbers to +-0.56D308.

>>> (For default D_floating, the maximum number allowed is 0.29D38.)

>>>

>>> On the other hand, IDL has no option for the larger G_floating numbers. This

>>> makes for problems when you convert a "G_floating REAL*8 " FORTRAN program to

>>> IDL.

>>>

>>> Peter Brooker

>>

>> Gee, maybe I do not understand your problem, but I did this quick test.

>>

>> IDL> a=.5d308

>> IDL> print,a

>> 5.0000000e+307

>>

>> Therefor you should be able to use the IDL DOUBLE to implement FORTRAN

>> G_floating calculations.

> I belive the original post was probably referring to a DEC Alpha machine. On

> the Alpha one can compile double precision code to be D_FLOAT, G_FLOAT or
> IEEE_FLOAT. The default for the DEC C compiler is G_FLOAT. IDL is clearly
> compiled with D_FLOAT, presumably so that the Alpha version is compatible with
> old VAX binary files, where D_FLOAT is the default. It would be possible and
> perhaps nice if RSI would provide 3 different versions (D_FLOAT, G_FLOAT and
> IEEE_FLOAT) of IDL for the Alpha platform. If you use CALL_EXTERNAL on the
> Alpha you need to make sure any routines you call are compiled with the same
> floating point format used for IDL.

Actually, the only thing that can be said for certain is that the original post
referred to a machine running VMS. You can't tell whether that's VMS on the
VAX or on the Alpha/AXP platform.

When running IDL under OSF/1, DEC's implementation of Unix for the Alpha, the
floating point format used is IEEE.

Bill Thompson
