
Subject: Re: Error in reading large Fortran unformatted files

Posted by [Nigel Wade](#) on Fri, 18 Feb 2011 09:45:07 GMT

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

On 17/02/11 18:15, OM wrote:

> On Feb 17, 6:16 pm, Nigel Wade <nmw-n...@ion.le.ac.uk> wrote:

>> On 17/02/11 15:02, Kenneth P. Bowman wrote:

>>

>>> In article

>>> <45a7d29c-1223-4e0e-8390-5a549f91c...@s11g2000yqh.googlegroup s.com >,

>>> OM <metu...@gmail.com> wrote:

>>

>>>> The output is now:

>>>> nb1=2147483657

>>>> nb2=995288272

>>

>>>> I still have no idea what this means.

>>

>>> nb1 is the largest possible positive 32-bit signed integer

>>

>>> IDL> print, 2L^31 - 1

>>> 2147483647

>>

>> The value quoted is 2147483657, which is 10 more than that. Assuming OM

>> cut'n'pasted the output, so it's not just a typo, it's a number which

>> has no immediate significance that I can think of.

>>

>> I do, however, agree that the problem is almost certainly due to trying
>> to write 4GB of data as a single FORTRAN unformatted record. I doubt
>> that when the FORTRAN unformatted format was devised it was never
>> envisioned that someone would try to output that much data in a single
>> write statement. The record length is a 32bit quantity. I don't see that
>> that can be altered based on platform, the format must be the same for
>> 32bit and 64bit platforms, and applications. I think the max. you can
>> possibly write in a single record is 2GB-1. To write 4GB will require at
>> least 3 records.

>>

>> --

>> Nigel Wade

>

> Well, here's the pickle - I'm getting no errors in writing the file,
> and with slight modifications I can read the data in Fortran and it
> seems to be valid.

>

> Ofer.

Well, maybe the underlying point is that the actual contents of FORTRAN
unformatted records are actually undefined, at least they never were

defined up to F77 which is last version of FORTRAN I used. They are an implementation issue, each compiler on each platform was free to define the format to be what it chose. Unformatted data was never meant to be portable, it was merely an efficient means of saving data from one FORTRAN program which could be read back by another FORTRAN program compiled by the same compiler on the same platform.

An ad hoc "standard" developed, which was that the first 4 and last 4 bytes contained the record length. This allowed some consistency check and limited portability (endian issues and other things). Maybe the FORTRAN compiler you are using has a different way of writing unformatted data records which extend beyond the limit of the previous 2GB "standard". Obviously it can read back data which it wrote, but IDL cannot.

--

Nigel Wade
