
Subject: Re: Fortran and byte swapping
Posted by [Tim Patterson](#) on Fri, 13 Dec 1996 08:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

If the DEC Fortran on the OSF side supports the byteswap flag, perhaps you could use some Fortran code and CALL_EXTERNAL to read in the data, or convert the datafile before using IDL?

Neither solution is particularly elegant.

Tim

Subject: Re: Fortran and byte swapping
Posted by [Simon Williams](#) on Fri, 13 Dec 1996 08:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

Liam Gumley wrote:

>
> Simon Williams wrote:
>> I have a large dataset which is in Unformatted variable-length
>> records written in FORTRAN on a SUN (big endian). I want to
>> read it on a DEC alpha running OSF (little endian) using IDL.
>> It is not really practical to convert the whole dataset.
>>
>> 1. I Can't use "openr, /f77_unformatted" because the
>> record length information stored in the file is byte-swapped
>> so it falls over on the first read.
>>
>> 2. I could drop the /f77_unformatted keyword and handle the
>> record structure myself, but even then I can't see a simple
>> way of doing it - I would need an IDL function which behaves
>> like readu, so that I can pass an arbitrary number of arguments
>> to it or I will have to clog up my routine with a load of
>> calls to byteorder for each parameter I read.
>>
>> Any ideas? The nicest solution would be a "byteswap keyword" to
>> the open routines - like the one provided by dec fortran, but I
>> guess that's a bit much to ask!
>
> If you don't want to convert the datafile, then:
>
> First, open the file without the /F77_UNFORMATTED keyword.
>
> Create a record length index array by reading (READU) the first length
> word, swapping it using SWAP_ENDIAN, store it in the array, skip to the
> next length word (POINT_LUN), swap it, etc. to the end of the file.
>

> You now have a record length index array that you can use to read the
> file. Point back to the beginning of the file, and skip the length
> word, read the number of data words indicated in the length array, and
> then swap the data array using SWAP_ENDIAN.
>
> Seems simple enough. I'd write some code for you but I haven't had my
> coffee yet.
>
> Cheers,
> Liam.

--

Thanks Liam,

What you suggested is more or less what I've come up with - (although I'd not come across SWAP_ENDIAN - my documentation's out of date - thanks).

However it doesn't really solve the problem - I've been given a stack of read code which runs on a SUN and contains MANY lines like this:

```
readu, lun, param1, param2 .....paramN  
readu, lun, next_param1, next_param2 ....  
....  
....
```

where the parameters may be scalars or arrays of any type. If I can possibly avoid it I'd rather not have to call swap_endian for each parameter after each readu call - I'd have to do it for several hundred named parameters. It would be much nicer to incorporate the swapping into a "readu-like" call, but I can't see any way of doing it. I'm likely to have to do similar things with other datasets in the future, so a more general solution would be welcome.

Simon Williams

Simon Williams British Atmospheric Data Centre
s.r.williams@rl.ac.uk Tel : 01235 446314 FAX : 01235 445848

Subject: Re: Fortran and byte swapping

Posted by [Liam Gumley](#) on Fri, 13 Dec 1996 08:00:00 GMT

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

Simon Williams wrote:

- > I have a large dataset which is in Unformatted variable-length
- > records written in FORTRAN on a SUN (big endian). I want to
- > read it on a DEC alpha running OSF (little endian) using IDL.
- > It is not really practical to convert the whole dataset.
- >
- > 1. I Can't use "openr,, /f77_unformatted" because the
- > record length information stored in the file is byte-swapped
- > so it falls over on the first read.
- >
- > 2. I could drop the /f77_unformatted keyword and handle the
- > record structure myself, but even then I can't see a simple
- > way of doing it - I would need an IDL function which behaves
- > like readu, so that I can pass an arbitrary number of arguments
- > to it or I will have to clog up my routine with a load of
- > calls to byteorder for each parameter I read.
- >
- > Any ideas? The nicest solution would be a "byteswap keyword" to
- > the open routines - like the one provided by dec fortran, but I
- > guess that's a bit much to ask!

If you don't want to convert the datafile, then:

First, open the file without the /F77_UNFORMATTED keyword.

Create a record length index array by reading (READU) the first length word, swapping it using SWAP_ENDIAN, store it in the array, skip to the next length word (POINT_LUN), swap it, etc. to the end of the file.

You now have a record length index array that you can use to read the file. Point back to the beginning of the file, and skip the length word, read the number of data words indicated in the length array, and then swap the data array using SWAP_ENDIAN.

Seems simple enough. I'd write some code for you but I haven't had my coffee yet.

Cheers,
Liam.
