
Subject: Re: reading past leading and training bytes
Posted by [Paul Van Delst\[1\]](#) on Thu, 06 Jul 2006 18:20:29 GMT
[View Forum Message](#) <> [Reply to Message](#)

glaciologist wrote:

- > I'm using readu to read a 101 X 55 (2D) binary file
- >
- > The file has leading and trailing 4 byte tags (00 00 56 CC in
- > hexadecimal).
- >
- > How to skip past these??!
- >
- > I use this code, which successfully reads in array, but first column
- > gets scrambled!~
- >
- > readu,1,data
- > byteorder,data,/xdrtof ; byteswap needed
- >
- > data are output by Linux Fedora 4 Portland Group FORTRAN executable
- > with -byteswapio flag set on compile

Without seeing the actual OPEN() statement in your Fortran code (or your IDL code for that matter), what happens if you use the /f77_unformatted flag in your IDL open? (If you're not doing it already).

I'm guessing that the 4byte "tags" you're seeing are actually the record markers that sequential, unformatted Fortran files generally contain to mark the begin and end of variable length records (with the actual value being the record length).

So try

OPEN, lun, file, /get_lun, /f77_unformatted

You might also need the /swap_endian keyword as well depending on the platform on which you are reading the byteswapped files. If the linux box that created the files is little-endian, then the PGI -byteswapio option will produce big-endian files (or vice versa).

paulv

--

Paul van Delst Ride lots.
CIMSS @ NOAA/NCEP/EMC Eddy Merckx
Ph: (301)763-8000 x7748
Fax:(301)763-8545

Subject: Re: reading past leading and training bytes
Posted by peter.albert@gmx.de on Fri, 07 Jul 2006 06:47:06 GMT

Hi Jason,

my previous poster's response using the `f77_unformatted` keyword is most likely the correct solution to your current problem, but a general solution for skipping any number of bytes would be:

```
IDL> openw, lun, filename, /get_lun
IDL> skip_it = bytarr(4) ; replace 4 by the actual number of bytes
you'd like to skip
IDL> readu, lun, skip_it
```

after that, the file pointer is at the correct position to read the "real" dataset. Well, that was how I read in those fortran-generated files all the time. While writing this, the idea comes to my head that it might have been smarter to just use

```
IDL> openw, lun, filename, /get_lun
IDL> point_lun, lun, 4
```

instead. Might be worth a try ...

Cheers,

Peter

glaciologist schrieb:

```
> I'm using readu to read a 101 X 55 (2D) binary file
>
> The file has leading and trailing 4 byte tags ( 00 00 56 CC in
> hexadecimal).
>
> How to skip past these??!
>
> I use this code, which successfully reads in array, but first column
> gets scrambled!~
>
> readu,1,data
> byteorder,data,/xdrtof ; byteswap needed
>
> data are output by Linux Fedora 4 Portland Group FORTRAN executable
> with -byteswapio flag set on compile
>
```

> With thanks!
>
> Jason Box

Subject: Re: reading past leading and training bytes
Posted by [Michael Galloy](#) on Fri, 07 Jul 2006 15:01:08 GMT
[View Forum Message](#) <> [Reply to Message](#)

Peter Albert wrote:

> While writing this, the idea comes to my head that
> it might have benn smarter to just use
>
> IDL> openw, lun, filename, /get_lun
> IDL> point_lun, lun, 4
>
> instead. Might be worth a try ...

You can even use:

```
IDL> openw, lun, filename, /get_lun  
IDL> skip_lun, lun, 4
```

to skip 4 bytes. (And this is easier to modify to skip 4 bytes multiple times instead of just at the beginning of the file.)

Mike

--

www.michaelgalloy.com
