Subject: Re: converting VMS idl files to UNIX pvwave
Posted by thompson on Thu, 16 Jul 1992 14:47:00 GMT
View Forum Message <> Reply to Message

In article <1992Jul16.004709.31221@urz.unibas.ch>, hofer@urz.unibas.ch (Remo Hofer) writes...
> In article <1992Jul11.034822.19367@bnlux1.bnl.gov>,
> jacobsen@bnlls1.nsls.bnl.gov (Chris Jacobsen) writes:
>> I had the same question. Mark Rivers of University Chicago/Brookhaven
>> Lab set things up using the XDR file format option. That allows
>> me to ship files back and forth between VMS and Unix
>
> Since XDR uses a description tag for every scalar data and not for a whole
> array, arrays written with XDR are often two times as big as without XDR.
> We ended up only wrtiting a stream of bytes to our data files using our own
> portable convertion routines to convert e.g. an array of floats into an array
> of bytes. We are transfering data (floats, longints, ints, bytes) between a VAX
> running VMS (little endian, VMS floats) and a Silicon Graphics Iris
> (big endian, IEEE floats) without any troubles this way.

I thought I might mention that IDL now supports converting floating point numbers with the BYTEORDER function. The relevant calls would be for converting from XDR (IEEE) formats to the host format:

```
  BYTEORDER,DATA,/NTOHS ;Short integer
  BYTEORDER,DATA,/NTOHL ;Long integer
  BYTEORDER,DATA,/XDRTOF ;Floating point
  BYTEORDER,DATA,/XDRTOD ;Double precision
  BYTEORDER,DATA,/XDRTOF ;Complex
```

and from host to XDR (IEEE) formats

```
  BYTEORDER,DATA,/HTONS ;Short integer
  BYTEORDER,DATA,/HTONL ;Long integer
  BYTEORDER,DATA,/FTOXDR ;Floating point
  BYTEORDER,DATA,/DTOXDR ;Double precision
  BYTEORDER,DATA,/FTOXDR ;Complex
```

That way you avoid the description tags. These calls are very fast.

> The bigger problem is the record formats on VMS.
> I'm not able to read a big file in stream format with pv~wave routines.
> (An error message saying that some buffer is too small is shown.)
> Converting the file to stream_lf format helps about this.
> Transfering a file written with pv~wave on the VAX in variable lenght record
> format to the Iris using ftp results in a mixture between actual data and
> record info on the unix side. And these are just two examples of troubles with
> VMS record orientated files.

If you use fixed length or stream records, rather than variable length, then it is possible to access particular bytes in a file using

OPEN,/BLOCK Opens the file for block access (on VAX)
POINT_LUN Select starting byte to read
READU Read data
WRITEU Write data

This is more portable than ASSOC, or any of the other more record-associated I/O calls. That way, on the VAX, then the actual file attributes are ignored. The file has to be written so that the bytes follow one another without interruption. Therefore, it won't work with variable length record files which have extra embedded bytes describing the records, which is different between VAXs and other computers.

I use the above commands to read and write FITS files, which can be transported without change between different computers, in particular, VAX/VMS computers and Sun workstations.

Bill Thompson