Subject: big file navigation
Posted by btt on Tue, 28 Oct 2003 16:07:33 GMT
View Forum Message <> Reply to Message

Hello,

After a longish hiatus from IDL I'm happily back at it. I am capturing underwater video footage and all is well until 'we' want more images from each 1.5 hour dive. So, I up the capture rate and simply stream more images into the file. Fine, except...

The file is composed some 55855 (640x240 byte) images which makes for a rather large 8 GB file. I discovered, to late of course, that I can't navigate around this file using ASSOC or POINT_LUN in this version of IDL because the machine is limited to 32-bit file navigation pointers.

```
IDL> help, !version, /str
** Structure !VERSION, 8 tags, length=76, data length=76:
   ARCH            STRING    'ppc'
   OS              STRING    'darwin'
   OS_FAMILY       STRING    'unix'
   OS_NAME         STRING    'Mac OS X'
   RELEASE         STRING    '6.0'
   BUILD_DATE      STRING    'Jun 27 2003'
   MEMORY_BITS     INT              32
   FILE_OFFSET_BITS
                   INT              32
```

See the File_Offset_Bits field? From this I know I can navigate through a 4 billion byte file (4GB if I have shuffled my decimal point correctly) since unsigned long integers go from 0 to about 4 billion. But bigger than that isn't possible on this platform with this version of IDL.

But here's something that took me a while to stumble upon. I can step through the file one image (or more) at a time reading in each one-at-a-time.

```
IDL> openR, U, file, /get_LUN
IDL> n = 55855
IDL> for i = 0L, 55855-1 do readU, U,d
```

Problem solved! I can break the file into smaller 'chunks' of 4GB by reading in images and writing them to a new file. OK!

But here's what I wish I understood (and could exploit!) Somewhere the system is keeping track of the file location so that READU knows where to read. So, deep down in its guts, IDL is navigating its way through with 64-bit file offsets. Ain't it? Is there anything here I could get my hooks into so I don't have to

break the original file into smaller ones?

Cheers,
Ben