On Monday, December 10, 2012 9:07:04 AM UTC-7, Dry in water wrote:
>> Hi,
>
>>
>
>> you might have to wait for some experts or some Tiff-experienced users to answer your question.
>
>>
>
>> My guess would be that if you cannot query the file, you cannot read the file.
>
>>
>
>> I never had problems importing tiff. I did this for 8-bit and 16-bit images, for black and white and for 24 color images.
>
>>
>
>>
>
>>
>
>> Do you have any idea what type of data you have in the image?
>
>>
>
>> How big is your image? About 380 Mb or 740 Mb?
>
>
>
>
>
>
>> Do you have enough free RAM to read that much data?
>
>>
>
>> Have you tried using the SUB_RECT keyword for Read_tiff, to check if that is a memory issue?
>
>>
>
>>
>

>>
>
>> Sorry, I'm just making questions and have no ready answer...
>
>>
>
>>
>
>>
>
>> Success,
>
>>
>
>> Helder
>
>
>
> Hello,
>
> Yes, I know interior value of this image should be between -1.7 and 27.6. I saw it using Arcgis.
>
> The size of file is 1.36 GB. I'm working on PC with 8 GB RAM and 64 bit operating system.
Disks space are more than 50GB.
>
>
>
> Anyway, thanks you very much Helder.

Try the QUERY_IMAGE function in IDL.  It will march down through a series of image format
types and attempt to open the image with each in turn until it is successful or it runs out of options,
ignoring the "hint" provided by the file name extension.  If this function returns a 1, then you can
check the INFO.TYPE structure tag that's returned through a positional argument to see what type
the file really is.

You could also try the READ_IMAGE function if you don't trust your file extensions.

Finally, you could simply open up the file in IDL and read the "magic number" from the start of the
file.  Most image file formats have magic numbers.

```
IDL> openr, lun, /get_lun, dialog_pickfile()
IDL> b = bytarr(4)
IDL> readu, lun, b
IDL> print, b, b, string(b), format = '(4z,4i,z4)'
IDL> free_lun, lun
```

Standard TIFFs should begin with "MM*" or "II*", in ASCII, depending on the byte order of the
data.  If the file is indeed a TIFF you should see output resembling

49   49   2a   0   73   73   42   0 II*

If you have some other byte sequence at the start of your file, you don't have a standard TIFF.

---