Subject: Re: Reading sequence of HDF files fails after some number of successes
Posted by BLesht on Wed, 21 Nov 2012 16:32:15 GMT
View Forum Message <> Reply to Message

Thanks, Klemen - doesn't help, I'm afraid. The code I'm working on is a shell script that creates a SeaDAS batch file that includes both IDL and SeaDAS procedures. Now I'm wondering if the problem lies somewhere else in the code. I've tried different combinations of the the HDF file access calls without success. The code processes a number of files with no trouble but then stops with the "Invalid HDF file or filename" error, and that error occurs at the same point no matter what file is next in the input list. In the past I've had IDL code stop because I forgot to free a LUN and there was a limit to the number of fileids that could be assigned. That is what started me thinking about how I was closing the HDF files.

Barry

On Wednesday, November 21, 2012 8:22:02 AM UTC-6, Klemen wrote:
> I am not sure it it helps, but I think you can delete the first line with:
>
> HDF_SD_ENDACCESS, varid & $
>
>
>
> As long as you work on the same file it works like this as code below(I use it open MODIS albedo Level 3 product). This is how I processed over 1000 files.
>
>
>
> Cheers, Klemen
>
>
>
>
>
>
>
>
> FOR y=0,n-1 DO BEGIN
>
> ; Open EOS-HDF
>
> i_fid = HDF_ISHDF(s_list[y])
>
> IF i_fid EQ 0 THEN BEGIN
>
> PRINT, ' !!! ' + s_list[y]
>
> PRINT, 'The input file does not exist or is not HDF format!'
>
> CONTINUE

```
>
>   ENDIF
>
>   sd_id = HDF_SD_START(s_list[y], /READ)
>
>   FOR i=0,6 DO BEGIN
>
>    out[i,*,*] = albedo_process_read_HDF(sd_id, 'Albedo_WSA_Band' + STRTRIM(i+1, 1),
extent[*,t], sds_id=sds_id)
>
>   ENDFOR
>
>   HDF_SD_ENDACCESS, sds_id
>
>   HDF_SD_END, sd_id
>
>
>
>
>
>  Where the called function "albedo_process_read_HDF" is:
>
>
>
>  FUNCTION albedo_process_read_HDF, sd_id, sd_name, extent, sds_id=sds_id
>
>    sd_index = HDF_SD_NAMETOINDEX(sd_id, sd_name)
>
>    sds_id = HDF_SD_SELECT(sd_id, sd_index)
>
>    HDF_SD_GETDATA, sds_id, tmp, START=extent[0:1], COUNT=extent[2:3]
>
>    RETURN, tmp
>
>  END
```