Subject: Re: Copying files in IDL
Posted by muswick on Mon, 15 Jul 2002 04:49:23 GMT
View Forum Message <> Reply to Message

Reimar Bauer <r.bauer@fz-juelich.de> wrote in message
news:<3D1180BA.1CB14940@fz-juelich.de>...
> Marc Schellens wrote:
>>
>> Reimar Bauer wrote:
>>>
>>> Sean Raffuse wrote:
>>>>
>>>> Hello folks. I would like to make a copy of an hdf file, changing only to
>>>> filename and the content of one of datasets in the hdf. I have no problem
>>>> putting in the new content, but this overwrites the old file, which I need
>>>> to preserve. How do I copy and/or rename the file in IDL? I don't want to
>>>> have to rebuild the entire hdf file and it's dozens of attributes. Just a
>>>> near exact copy with a different name.
>>>>
>>>> Thanks in advance,
>>>>
>>>> Sean
>>>
>>> Dear Sean,
>>>
>>> what's with this primitive way.
>>>
>>> spawn,'cp file.hdf new_file.hdf'
>>>
>>> and then operations on then new file.
>>>
>>> regards
>>
>> The disadvantage of this solution is that it's not portable.
>> If the filesize is not to large, you can do something like:
>>
>> openr,1,'file.hdf'
>> fs=fstat(1)
>> b=bytarr(fs.size)
>> readu,1,b
>> close,1
>> openw,1,'new_file.hdf'
>> writeu,1,b
>> close,1
>>
>> cheers,
>> marc
>

> Then we should do a feature request for file_copy
> probably using this routine with /GET_LUN.
>
> If bytarr is replaced with b=make_array(/nozero,ft.size,/byte)
> the routine is much faster if filesize is big.
>
>
> One problem could occure if HDF file size is very large.
> Because data is named and index sequentiell organized in this file
> and there is no reason by reading/writing with HDF commands
> of the whole file at once.
> HDF filesize could be bigger than 1 or 10 Giga Byte .
>
> In this case a system command will be better and
> you can do a case depending of !version.os_family.
>
>
> regards
> Reimar
>
>
>
>
>
>
>
>
>
>
>
>
>
>
>
> --
> Reimar Bauer
>
> Institut fuer Stratosphaerische Chemie (ICG-I)
> Forschungszentrum Juelich
> email: R.Bauer@fz-juelich.de
> -- -------
> a IDL library at ForschungsZentrum Juelich
> http://www.fz-juelich.de/icg/icg1/idl_icglib/idl_lib_intro.h tml
> == =======

I wrote a copy file procedure awhile ago that handles any size file -
within the constraints of IDL's fstat limits. Similar to what was
posted, but includes error handling.

Gary Muswick

muswick@uhrad.com
muswick@zalenllc.com

```
;-----Start CopyFile.pro

; CopyFile.pro v1.1 2002/06/14
; Copyright ï¿½ 2000, 2001, 2002 Zalen LLC
;
; Purpose: A procedure to copy files of any size.
;
; Author: Gary Muswick, September 21, 2000
;

;+
; INPUTS:
;       sScrFile : The full filename which contains the file to be
copied.
;       sDstFile : The full filename which contains the destination
file.
;
; KEYWORD PARAMETERS:
;   None.
;
; OUTPUTS:
;   None.
;
; RESTRICTIONS:
;   Only scalar strings are allowed for input.
;
; EXAMPLE:
;   CopyFile,'C:\WINNT\Temp\test.txt', 'D:\copy of test.txt'

; MODIFICATION HISTORY:
;   June 14, 2002 - Gary Muswick
;       Fixed problem with buffer size when multiple reads are
;       required.  Added NOZERO keyword to BYTARR.
;
FUNCTION ZLLCErrorHandler, lErrorStatus

  IF lErrorStatus EQ 0L THEN RETURN, 0
  HELP, /TRACEBACK, OUTPUT = asTraceback
  ; Get rid of this routine from the traceback
  IF N_ELEMENTS(asTraceback) GT 1 THEN asTraceback = asTraceback[1:*]
  asRoutineInfo = STR_SEP(STRCOMPRESS(asTraceback[0]), ' ') ; Breakup
info
  asError = asRoutineInfo[1]              ; Current Routine is 2nd
item
  asError = [asError,'Error index: ' + STRING(lErrorStatus)]
```

```
  asError = [asError,'Error message: ' + !ERROR_STATE.MSG]
  asError = [asError,'System Error: ' + !ERROR_STATE.SYS_MSG]
  asError = [asError, asTraceback]
  a = DIALOG_MESSAGE(asError)
  RETURN, 1
END

PRO CopyFile, sSrcFile, sDstFile

  CATCH, lErrorStatus
  IF ZLLCErrorHandler(lErrorStatus) THEN RETURN
  OPENR, iSlun, sSrcFile[0], /GET_LUN
  OPENW, iDlun, sDstFile[0], /GET_LUN
  mFS = FSTAT(iSlun)
  lSizeToWrite = 1024L*1024L
  lBytesWritten = 0L
  bData = BYTARR(lSizeToWrite < mFS.size, /NOZERO)
  WHILE lBytesWritten LT mFS.size DO BEGIN
    READU, iSlun, bData
    WRITEU, iDlun, bData
    lBytesWritten = lBytesWritten + N_ELEMENTS(bData)
    IF lBytesWritten LT mFS.size THEN BEGIN
      IF (mFS.size - lBytesWritten) LT lSizeToWrite THEN BEGIN
        bData = BYTARR(lSizeToWrite < (mFS.size - lBytesWritten),
/NOZERO)
      ENDIF
    ENDIF
  ENDWHILE
  FREE_LUN, iSlun   ; Close and free the LUN
  FREE_LUN, iDlun   ; Close and free the LUN

END

;-----End CopyFile.pro
```