Subject: The good way to use cdfid's
Posted by Fabzou on Mon, 06 Jun 2011 09:42:39 GMT
View Forum Message <> Reply to Message

Dear IDL fans,

Like many of you, i made my own NCDF object to handle NCDF files.

When the NCDF object is built, the init function opens the ncdf file
just once and stores the ncdf ID (given by NCDF_OPEN) as an attribute
(as long). In the object destroy procedure, the file is closed again,
like in the example code at the end of this message.

The reason for this choice was just to avoid many open/close, each time
a variable is asked, for example.

It worked well for a long time, BUT recently a user wanted to create and
destroy many many ncdf objects and finally got the following error after
opening exactly 16382 files:

%W_NCDF::INIT: ERROR! NCDF_INQUIRE: -2147483648 is not a valid cdfid.

where -2147483648 is then the ID returned by NCDF_OPEN on the line just
before, which didn't throw any error.

Any idea where this error could come from?

My only explanation is that I don't close the files properly, but the
OBJ_DESTROY procedure was called at the end of each loop and after
testing, the files seem also well closed...

I looked into David's code from NCDF_DATA and saw that David is opening
and closing the file every time (e.g. in NCDF_DATA::ReadVariable). What
are the reasons for this choice? This would open and close the same file
many times, and throw the error I obtain even sooner (I guess).

Thanks for your help on this !

Fabien

IDL> print, !VERSION
{ x86_64 linux unix linux 7.1.1 Aug 21 2009   64   64}

```
 ;------------------------------------------------------------ ----------
pro w_NCDF__Define

  struct = {w_NCDF                , $
          path:             ''    , $ ;
```

```
        cdfid:          0L    ,  $ ; id given by NCDF_OPEN
        etc...
        }

end

function w_NCDF::Init, FILE = file
   ; code ....
   self.cdfid = NCDF_OPEN(file, /NOWRITE)
   ; code ...
end

pro w_NCDF::Cleanup

   NCDF_CLOSE, self.cdfid

END
 ;------------------------------------------------------------ ----------
```

Subject: Re: The good way to use cdfid's
Posted by David Fanning on Mon, 06 Jun 2011 13:44:47 GMT
View Forum Message <> Reply to Message

Fabzou writes:

> I am sorry, I made a mistake in my previous mail. The computer on which
> the error is happening is:
>
> { x86_64 Win32 Windows Microsoft Windows 7.1.2 Oct 28 2009 64 64}

Well, here is the code I used. No problem running the program
on my identical Windows machine:

```
file = "C:\IDL\data\netCDF\VECTORS.20070321_041224.nc"
FOR j=0L, 16500 DO BEGIN
  o = Obj_New('NCDF_FILE', file)
  obj_destroy, o
  print, j
ENDFOR
END


IDL> print, !version
{ x86_64 Win32 Windows Microsoft Windows 7.1.2 Oct 28 2009   64   64}
```

Cheers,

David
--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.idlcoyote.com/
Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: The good way to use cdfid's
Posted by Fabzou on Mon, 06 Jun 2011 14:13:24 GMT
View Forum Message <> Reply to Message

Well, I just made the simplest test ever on my linux computer:

```
PRO bug_ncdf
  file =  '/home/fab/disk/Data/TRMM/TRMM_3B42_Rashmi/3B42.081001.18.6A .nc'
  FOR j=0L, 33000 DO BEGIN
    cdfid = NCDF_OPEN(file)
    inq = NCDF_INQUIRE(cdfid)
    NCDF_CLOSE, cdfid
  ENDFOR
END

% Compiled module: BUG_NCDF.
IDL>  BUG_NCDF
% Loaded DLM: NCDF.
% NCDF_INQUIRE: -2147483648 is not a valid cdfid.
% Execution halted at: BUG_NCDF           5
/home/fab/disk/IDLWorkspace/ze_Dev/bug_ncdf.pro
%                    $MAIN$
IDL> print, j
       32767
IDL> print, !version
{ x86_64 linux unix linux 7.1.1 Aug 21 2009      64      64}
```

I cannot open more than 32766 files on my computer.

I'll make the test on Windows when I can...

Fabien

On 06/06/2011 03:44 PM, David Fanning wrote:
> Fabzou writes:
>
>> I am sorry, I made a mistake in my previous mail. The computer on which
>> the error is happening is:

>>
>> { x86_64 Win32 Windows Microsoft Windows 7.1.2 Oct 28 2009 64 64}
>
> Well, here is the code I used. No problem running the program
> on my identical Windows machine:
>
> file = "C:\IDL\data\netCDF\VECTORS.20070321_041224.nc"
> FOR j=0L, 16500 DO BEGIN
> o = Obj_New('NCDF_FILE', file)
> obj_destroy, o
> print, j
> ENDFOR
> END
>
>
> IDL> print, !version
> { x86_64 Win32 Windows Microsoft Windows 7.1.2 Oct 28 2009 64 64}
>
>
> Cheers,
>
> David

Subject: Re: The good way to use cdfid's
Posted by David Fanning on Mon, 06 Jun 2011 14:28:49 GMT
View Forum Message <> Reply to Message

Fabzou writes:

> I'll make the test on Windows when I can...

Your program runs fine on my Windows machine in both IDL 7.1.2
and IDL 8.1.

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.idlcoyote.com/
Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: The good way to use cdfid's
Posted by Fabzou on Tue, 07 Jun 2011 10:22:05 GMT
View Forum Message <> Reply to Message

Hi,

I made a standard test program and made it run on the computer from a colleague:

```
PRO bug_ncdf

  Catch, theError
  IF theError NE 0 THEN BEGIN
    Catch, /Cancel
    if N_ELEMENTS(j) eq 0 then j = 0
    print, !Error_State.Msg
    print, 'Error occured at j = ', j
    print, !VERSION
    RETURN
  ENDIF

  file = '/../3B42.081001.18.6A.nc' ; put your own file here
  FOR j=0L, 33000 DO BEGIN
    cdfid = NCDF_OPEN(file)
    inq = NCDF_INQUIRE(cdfid)
    NCDF_CLOSE, cdfid
  ENDFOR

END
```

The program throws an error:

```
IDL> bug_ncdf
% Loaded DLM: NCDF.
NCDF_INQUIRE: -2147483648 is not a valid cdfid.
Error occured at j =       32767
{ x86_64 Win32 Windows Microsoft Windows 7.1.2 Oct 28 2009      64      64}
IDL> bug_ncdf
NCDF_INQUIRE: -2147418112 is not a valid cdfid.
Error occured at j =           0
{ x86_64 Win32 Windows Microsoft Windows 7.1.2 Oct 28 2009      64      64}
```

IDL cannot open any further NCDF files afterwards. RESET_SESSION won't change anything, just restart the workbench will do it...

How is this possible ??? Could it be related to the fact that we also have ENVI installed on our computers? Is it related to the internal Ncdf libraries ???

Subject: Re: The good way to use cdfid's
Posted by David Fanning on Tue, 07 Jun 2011 11:25:57 GMT
View Forum Message <> Reply to Message

Fabzou writes:

> How is this possible ??? Could it be related to the fact that we also
> have ENVI installed on our computers? Is it related to the internal Ncdf
> libraries ???

My IDL session:

IDL Version 7.1.2, Microsoft Windows (Win32 x86_64 m64). (c) 2009, ITT Visual Information Solutions
Licensed for personal use by David Fanning only.
All other use is strictly prohibited.

Current Directory: C:\IDL

Finished executing IDL startup file...
IDL> Bug_NCDF
% Compiled module: BUG_NCDF.
% Loaded DLM: NCDF.
IDL>

As you can see, no problems whatsoever. Are you running the version
of IDL that came with ENVI? Really don't know....

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.idlcoyote.com/
Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: The good way to use cdfid's
Posted by Maarten[1] on Tue, 07 Jun 2011 14:04:59 GMT
View Forum Message <> Reply to Message

On Jun 7, 12:22 pm, Fabzou <fabien.mauss...@tu-berlin.de> wrote:

[...]

> IDL cannot open any further NCDF files afterwards. RESET_SESSION won't
> change anything, just restart the workbench will do it...
>
> How is this possible ??? Could it be related to the fact that we also
> have ENVI installed on our computers? Is it related to the internal Ncdf
> libraries ???

Here in the Netherlands (KNMI) we are struggling with a similar issue.

Sample code:

```
for i=0L,9999999 do begin
  print, i
  ncid = NCDF_OPEN('TM5_V1_griddef.nc')
  NCDF_CLOSE, ncid
endfor
end
```

This will crash (or rather hang) IDL on the system of a coworker with i between 100,000 and 900,000 (notice that nothing is done with the netcdf file at all, just open and close). My (supposedly identical) machine runs this without issue. We haven't been able to identify anything different in our environment, files are not accessed over the network, ... we're at a loss.

Some hints:

* there is an ITTVIS incident on the subject: 241865. They can reproduce the issue, perhaps you can help connect the dots.
* The issue also can occur with hdf-4 and hdf-5 files.
* A switch in the version of the HDF-5 library (1.6.X to 1.8.Y) in IDL8 may have to do with the appearance of the issue. But you are dealing with netcdf (3, I presume?) so that may not apply. Of course it is possible that IDL8 switched to netcdf4 and therefore inherits all behaviour (good and bad) of hdf-5. The HDF5 helpdesk is also aware of this issue. They have identified an issue in some of the files we're working with, and pointed to a possible cause. A resolution is not available though. HDF-5 1.6.X (X < 5) can read these files (minus the corrupted part), later versions of the HDF-5 library will simply crash when the corrupted part is read.

```
IDL> print, !VERSION
{ x86 linux unix linux 8.0.1 Oct  6 2010      32      64}
```
(as far as I know, this happened in IDL 7 as well).

All clues for a solution are apreciated,

Maarten

Subject: Re: The good way to use cdfid's
Posted by Jos de Laat on Mon, 20 Jun 2011 13:42:20 GMT
View Forum Message <> Reply to Message

Dear Fabzou,

I think I ran into a similar problem some time ago. I had a piece of IDL that subsequently opens and reads data from many - albeit different - NCDF files in IDL. This always worked well, but after the operating system of my workstation changed to Fedora 13 my IDL program started to "stall", it simply stopped running and could only be stopped by killing idl ("ctrl C" did not work; happens in both IDL v70 and v80). This occurs a bit randomly, but it always happens. After some fiddling around it turned out that opening and closing any NCDF files for many times leads to this error. This was also the case for HDF and HDF5 files. Interestingly, the problem did not occur for IDL running under windows (IDL v64, I think).

I contacted ITTVIS about it and they gave me the following code example "that caused a crash on several linux flavours:" (their words):

```
pro create_hand_sample,  file
  compile_opt idl2, logical_predicate
  print, file
  sample_var =  dindgen( 4, 3 )
  ncid = ncdf_create(file, /CLOBBER)
  dimidx = ncdf_dimdef( ncid, 'x', 4 )
  dimidy = ncdf_dimdef( ncid, 'y', 3 )
  varid = ncdf_vardef( ncid,  'variable', /double )
  ncdf_control, ncid, /ENDEF
  ncdf_varput,  ncid,  varid, sample_var
  ncdf_close,  ncid

end

pro crash_ncdf
  compile_opt idl2, logical_predicate

  catch,  theError
  print, 'theError = ',theError
  if theError ne 0 then begin
    Catch, /cancel
    print,  "i = ",  i
    print,  "ncdf_fid = ",  ncdf_fid
    message,  !ERROR_STATE.msg
  endif
```

```
  file = '/tmp/sample_hand.nc'
  create_hand_sample,  file

  long_iteration = 19000000L

  for i =  0l,  long_iteration do begin
    ncdf_fid =  ncdf_open( file,  write = 0 )
    ncdf_close,  ncdf_fid
  endfor

  print,  'Successfully completed first iterations'
  window,  /free
  wdelete

  for i =  0l,  long_iteration do begin
    ncdf_fid =  ncdf_open( file,  write = 0 )
    ncdf_close,  ncdf_fid
  endfor

  print,  'Successfully completed second iterations'

end


>>>  output >>>

theError =       -1091
ncdf_fid =       65536
% CRASH_NCDF: NCDF_OPEN: Unable to open the file "/tmp/
sample_hand.nc". (NC_ERROR=-31)
% Execution halted at: CRASH_NCDF         24 /usr/people/laatdej/
Documents/tmp/ITT_sample_code.pro
%                     CREATE_HAND_SAMPLE    3 /usr/people/laatdej/
Documents/tmp/ITT_sample_code.pro
%                     $MAIN$
```

...

They promised to look into this but I haven't heard since (so mailed them again today). I'll keep you posted if a solution is found.

Subject: Re: The good way to use cdfid's
Posted by Fabzou on Mon, 20 Jun 2011 14:18:30 GMT
View Forum Message <> Reply to Message

Hi,

Thanks for your reply. In my case, it is not occurring randomly but
after a certain number of open/close :

```
IDL> print, !VERSION
{ x86_64 linux unix linux 7.1.1 Aug 21 2009      64      64}
IDL> crash_ncdf
% Compiled module: CRASH_NCDF.
theError =           0
/tmp/sample_hand.nc
% Loaded DLM: NCDF.
theError =       -1005
i =       32766
ncdf_fid =  -2147483648
% CRASH_NCDF: NCDF_CLOSE: -2147483648 is not a valid cdfid.
% Execution halted at: CRASH_NCDF         24
/home/fab/disk/IDLWorkspace/ze_Dev/crash_ncdf.pro
%                    $MAIN$
```

Well... I've given up using IDL for low-level NCDF tools anyway
(post-processing, etc.) but I'll be happy to find a solution to this.

We will switch to CDO, but I now that some very big organizations (e.g.
http://oceancolor.gsfc.nasa.gov/) use IDL to generate some
post-processed products in NCDF format. Probably they a stable version
of IDL working...

Fabzou

On 06/20/2011 03:42 PM, Jos de Laat wrote:

```
>  pro create_hand_sample,  file
>      compile_opt idl2, logical_predicate
>      print, file
>      sample_var =  dindgen( 4, 3 )
>      ncid = ncdf_create(file, /CLOBBER)
>      dimidx = ncdf_dimdef( ncid, 'x', 4 )
>      dimidy = ncdf_dimdef( ncid, 'y', 3 )
>      varid = ncdf_vardef( ncid,  'variable', /double )
>      ncdf_control, ncid, /ENDEF
>      ncdf_varput,  ncid,  varid, sample_var
>      ncdf_close,  ncid
>
>  end
>
```

```
> pro crash_ncdf
>    compile_opt idl2, logical_predicate
>
>    catch,  theError
>    print, 'theError = ',theError
>    if theError ne 0 then begin
>       Catch, /cancel
>       print,  "i = ",  i
>       print,  "ncdf_fid = ",  ncdf_fid
>       message,  !ERROR_STATE.msg
>    endif
>
>    file = '/tmp/sample_hand.nc'
>    create_hand_sample,  file
>
>    long_iteration = 19000000L
>
>    for i =  0l,  long_iteration do begin
>       ncdf_fid =  ncdf_open( file,  write = 0 )
>       ncdf_close,  ncdf_fid
>    endfor
>
>    print,  'Successfully completed first iterations'
>    window,  /free
>    wdelete
>
>    for i =  0l,  long_iteration do begin
>       ncdf_fid =  ncdf_open( file,  write = 0 )
>       ncdf_close,  ncdf_fid
>    endfor
>
>    print,  'Successfully completed second iterations'
>
> end
```