Subject: NCDF_IS_NCDF
Posted by khyde on Thu, 21 Jan 2016 20:21:00 GMT

View Forum Message <> Reply to Message

Hello,

Is there a netcdf equivalent to HDF_ISHDF? I am basically looking for a way to test if a file is a netcdf file before trying to open the file (and the having the program crash because it isn't actually a netcdf file).

Thank you,
Kim

Subject: Re: NCDF_IS_NCDF
Posted by Michael Galloy on Fri, 22 Jan 2016 03:26:26 GMT

View Forum Message <> Reply to Message

On 1/21/16 1:21 PM, KH wrote:
> Hello,
>
> Is there a netcdf equivalent to HDF_ISHDF? I am basically looking
> for a way to test if a file is a netcdf file before trying to open
> the file (and the having the program crash because it isn't actually
> a netcdf file).

Yes, I would normally recommend just trying to open the file and catching the error if it fails, but, as you noted, it actually crashes IDL:

IDL> id = ncdf_open(filename)
Assertion failed: (0), function NC4_open, file nc4file.c, line 2696.
Abort trap: 6

Are your files netCDF 4 files? You can use H5F_IS_HDF5 if you don't want pre-4 netCDF files:

IDL> help, h5f_is_hdf5(file_which('ncgroup.nc'))
<Expression>   LONG   =   1
IDL> help, h5f_is_hdf5(file_which('sample.nc'))
<Expression>   LONG   =   0

Mike
--
Michael Galloy
www.michaelgalloy.com
Modern IDL: A Guide to IDL Programming (http://modernidl.idldev.com)

Subject: Re: NCDF_IS_NCDF
Posted by Michael Galloy on Fri, 22 Jan 2016 05:23:32 GMT
View Forum Message <> Reply to Message

On 1/21/16 8:26 PM, Michael Galloy wrote:
> On 1/21/16 1:21 PM, KH wrote:
>> Hello,
>>
>> Is there a netcdf equivalent to HDF_ISHDF? I am basically looking
>> for a way to test if a file is a netcdf file before trying to open
>> the file (and the having the program crash because it isn't actually
>> a netcdf file).
>
> Yes, I would normally recommend just trying to open the file and
> catching the error if it fails, but, as you noted, it actually crashes IDL:
>
> IDL> id = ncdf_open(filename)
> Assertion failed: (0), function NC4_open, file nc4file.c, line 2696.
> Abort trap: 6
>
> Are your files netCDF 4 files? You can use H5F_IS_HDF5 if you don't want
> pre-4 netCDF files:
>
> IDL> help, h5f_is_hdf5(file_which('ncgroup.nc'))
> <Expression>   LONG   =   1
> IDL> help, h5f_is_hdf5(file_which('sample.nc'))
> <Expression>   LONG   =   0
>
>
> Mike

I have a C routine that can tell in general. I can clean that up tomorrow.

Mike
--
Michael Galloy
www.michaelgalloy.com
Modern IDL: A Guide to IDL Programming (http://modernidl.idldev.com)

Subject: Re: NCDF_IS_NCDF
Posted by khyde on Fri, 22 Jan 2016 19:20:21 GMT
View Forum Message <> Reply to Message

On Thursday, January 21, 2016 at 10:26:36 PM UTC-5, Michael Galloy wrote:
> On 1/21/16 1:21 PM, KH wrote:
>> Hello,
>>

>> Is there a netcdf equivalent to HDF_ISHDF? I am basically looking
>> for a way to test if a file is a netcdf file before trying to open
>> the file (and the having the program crash because it isn't actually
>> a netcdf file).
>
> Yes, I would normally recommend just trying to open the file and
> catching the error if it fails, but, as you noted, it actually crashes IDL:
>
> IDL> id = ncdf_open(filename)
> Assertion failed: (0), function NC4_open, file nc4file.c, line 2696.
> Abort trap: 6
>
> Are your files netCDF 4 files? You can use H5F_IS_HDF5 if you don't want
> pre-4 netCDF files:
>
> IDL> help, h5f_is_hdf5(file_which('ncgroup.nc'))
> <Expression>   LONG   =   1
> IDL> help, h5f_is_hdf5(file_which('sample.nc'))
> <Expression>   LONG   =   0
>
>
> Mike
> --
> Michael Galloy
> www.michaelgalloy.com
> Modern IDL: A Guide to IDL Programming (http://modernidl.idldev.com)

These particular files are netcdf4 so I can use the H5f program, but not all of my files are netcdf4
so I was hoping to find a way to automatically test what type of files they are.
Thanks for your assistance,
Kim

Subject: Re: NCDF_IS_NCDF
Posted by khyde on Fri, 22 Jan 2016 19:26:20 GMT
View Forum Message <> Reply to Message

On Friday, January 22, 2016 at 2:20:25 PM UTC-5, kimber...@noaa.gov wrote:
> On Thursday, January 21, 2016 at 10:26:36 PM UTC-5, Michael Galloy wrote:
>> On 1/21/16 1:21 PM, KH wrote:
>>> Hello,
>>>
>>> Is there a netcdf equivalent to HDF_ISHDF? I am basically looking
>>> for a way to test if a file is a netcdf file before trying to open
>>> the file (and the having the program crash because it isn't actually
>>> a netcdf file).
>>
>> Yes, I would normally recommend just trying to open the file and

>> catching the error if it fails, but, as you noted, it actually crashes IDL:
>>
>> IDL> id = ncdf_open(filename)
>> Assertion failed: (0), function NC4_open, file nc4file.c, line 2696.
>> Abort trap: 6
>>
>> Are your files netCDF 4 files? You can use H5F_IS_HDF5 if you don't want
>> pre-4 netCDF files:
>>
>> IDL> help, h5f_is_hdf5(file_which('ncgroup.nc'))
>> <Expression>   LONG   =   1
>> IDL> help, h5f_is_hdf5(file_which('sample.nc'))
>> <Expression>   LONG   =   0
>>
>>
>> Mike
>> --
>> Michael Galloy
>> www.michaelgalloy.com
>> Modern IDL: A Guide to IDL Programming (http://modernidl.idldev.com)
>
> These particular files are netcdf4 so I can use the H5f program, but not all of my files are
netcdf4 so I was hoping to find a way to automatically test what type of files they are.
> Thanks for your assistance,
> Kim

In the end, I am mainly trying to quickly determine if the netcdf files I downloaded are corrupt or
incomplete. That way if they are bad, I can remove them so that they don't crash during
subsequent processing. Currently I am opening the file and then using NCDF_INQUIRE to get
the basic structure information, but if there is an easier way to check the file, I am open for ideas.

Kim

Subject: Re: NCDF_IS_NCDF
Posted by Michael Galloy on Fri, 22 Jan 2016 19:53:17 GMT
View Forum Message <> Reply to Message

On 1/22/16 12:26 PM, kimberly.hyde@noaa.gov wrote:
> On Friday, January 22, 2016 at 2:20:25 PM UTC-5, kimber...@noaa.gov
> wrote:
>> On Thursday, January 21, 2016 at 10:26:36 PM UTC-5, Michael Galloy
>> wrote:
>>> On 1/21/16 1:21 PM, KH wrote:
>>>> Hello,
>>>>
>>>> Is there a netcdf equivalent to HDF_ISHDF? I am basically

>>>> looking for a way to test if a file is a netcdf file before
>>>> trying to open the file (and the having the program crash
>>>> because it isn't actually a netcdf file).
>>>
>>> Yes, I would normally recommend just trying to open the file and
>>> catching the error if it fails, but, as you noted, it actually
>>> crashes IDL:
>>>
>>> IDL> id = ncdf_open(filename) Assertion failed: (0), function
>>> NC4_open, file nc4file.c, line 2696. Abort trap: 6
>>>
>>> Are your files netCDF 4 files? You can use H5F_IS_HDF5 if you
>>> don't want pre-4 netCDF files:
>>>
>>> IDL> help, h5f_is_hdf5(file_which('ncgroup.nc')) <Expression>
>>> LONG   =   1 IDL> help,
>>> h5f_is_hdf5(file_which('sample.nc')) <Expression>   LONG   =
>>> 0
>>>
>>>
>>> Mike -- Michael Galloy www.michaelgalloy.com Modern IDL: A Guide
>>> to IDL Programming (http://modernidl.idldev.com)
>>
>> These particular files are netcdf4 so I can use the H5f program,
>> but not all of my files are netcdf4 so I was hoping to find a way
>> to automatically test what type of files they are. Thanks for your
>> assistance, Kim
>
>
> In the end, I am mainly trying to quickly determine if the netcdf
> files I downloaded are corrupt or incomplete. That way if they are
> bad, I can remove them so that they don't crash during subsequent
> processing. Currently I am opening the file and then using
> NCDF_INQUIRE to get the basic structure information, but if there is
> an easier way to check the file, I am open for ideas.
>
> Kim
>

I have a routine, MG_NC_ISNCDF, in my library mglib (at
github.com/mgalloy/mglib). You have to build the library to get the
routine (and, of course, put the location of where the DLM is installed
into your IDL DLM path). You will need the netCDF, HDF5, curl, libsz,
and libz libraries installed. Change the paths to the proper locations,
but something like the following should work for your system:

mkdir build
cd build

```
cmake \
  -DCMAKE_INSTALL_PREFIX:PATH=~/software/mglib \
  -DNETCDF_INCLUDE_DIR:PATH=/usr/local/include \
  -DNETCDF_LIBRARY:PATH=/usr/local/lib/libnetcdf.a \
  -DHDF5_LIBRARY:PATH=/usr/local/lib/libhdf5.a \
  -DHDF5_LA_LIBRARY:PATH=/usr/local/lib/libhdf5_hl.a \
  -DCURL_LIBRARY:PATH=/usr/lib/libcurl.dylib \
  -DSZ_LIBRARY:PATH=/usr/local/lib/libsz.a \
  -DZ_LIBRARY:PATH=/usr/lib/libz.dylib \
  ..
```

Mike
--
Michael Galloy
www.michaelgalloy.com
Modern IDL: A Guide to IDL Programming (http://modernidl.idldev.com)

Subject: Re: NCDF_IS_NCDF
Posted by khyde on Fri, 22 Jan 2016 20:04:56 GMT

View Forum Message <> Reply to Message

On Friday, January 22, 2016 at 2:48:38 PM UTC-5, Mike Galloy wrote:
> On 1/22/16 12:26 PM, kimberly.hyde@noaa.gov wrote:
>> On Friday, January 22, 2016 at 2:20:25 PM UTC-5, kimber...@noaa.gov
>> wrote:
>>> On Thursday, January 21, 2016 at 10:26:36 PM UTC-5, Michael Galloy
>>> wrote:
>>>> On 1/21/16 1:21 PM, KH wrote:
>>>> > Hello,
>>>> >
>>>> > Is there a netcdf equivalent to HDF_ISHDF? I am basically
>>>> > looking for a way to test if a file is a netcdf file before
>>>> > trying to open the file (and the having the program crash
>>>> > because it isn't actually a netcdf file).
>>>>
>>>> Yes, I would normally recommend just trying to open the file and
>>>> catching the error if it fails, but, as you noted, it actually
>>>> crashes IDL:
>>>>
>>>> IDL> id = ncdf_open(filename) Assertion failed: (0), function
>>>> NC4_open, file nc4file.c, line 2696. Abort trap: 6
>>>>
>>>> Are your files netCDF 4 files? You can use H5F_IS_HDF5 if you
>>>> don't want pre-4 netCDF files:
>>>>
>>>> IDL> help, h5f_is_hdf5(file_which('ncgroup.nc')) <Expression>

```

>>>> LONG    =         1 IDL> help,
>>>> h5f_is_hdf5(file_which('sample.nc')) <Expression>   LONG    =
>>>> 0
>>>>
>>>>
>>>> Mike -- Michael Galloy www.michaelgalloy.com Modern IDL: A Guide
>>>> to IDL Programming (http://modernidl.idldev.com)
>>>
>>> These particular files are netcdf4 so I can use the H5f program,
>>> but not all of my files are netcdf4 so I was hoping to find a way
>>> to automatically test what type of files they are. Thanks for your
>>> assistance, Kim
>>
>>
>> In the end, I am mainly trying to quickly determine if the netcdf
>> files I downloaded are corrupt or incomplete.  That way if they are
>> bad, I can remove them so that they don't crash during subsequent
>> processing.  Currently I am opening the file and then using
>> NCDF_INQUIRE to get the basic structure information, but if there is
>> an easier way to check the file, I am open for ideas.
>>
>> Kim
>>
>
> I have a routine, MG_NC_ISNCDF, in my library mglib (at
> github.com/mgalloy/mglib). You have to build the library to get the
> routine (and, of course, put the location of where the DLM is installed
> into your IDL DLM path). You will need the netCDF, HDF5, curl, libsz,
> and libz libraries installed. Change the paths to the proper locations,
> but something like the following should work for your system:
>
> mkdir build
> cd build
>
> cmake \
>   -DCMAKE_INSTALL_PREFIX:PATH=~/software/mglib \
>   -DNETCDF_INCLUDE_DIR:PATH=/usr/local/include \
>   -DNETCDF_LIBRARY:PATH=/usr/local/lib/libnetcdf.a \
>   -DHDF5_LIBRARY:PATH=/usr/local/lib/libhdf5.a \
>   -DHDF5_LA_LIBRARY:PATH=/usr/local/lib/libhdf5_hl.a \
>   -DCURL_LIBRARY:PATH=/usr/lib/libcurl.dylib \
>   -DSZ_LIBRARY:PATH=/usr/local/lib/libsz.a \
>   -DZ_LIBRARY:PATH=/usr/lib/libz.dylib \
>   ..
>
> Mike
> --
> Michael Galloy

> www.michaelgalloy.com
> Modern IDL: A Guide to IDL Programming (http://modernidl.idldev.com)

Hello Mike,
Is this a C program or IDL?  It definitely isn't clear to me how to implement this.  All I really know is basic IDL and the code in the program doesn't look familiar to me.
Kim

---

## Subject: Re: NCDF_IS_NCDF
Posted by Michael Galloy on Fri, 22 Jan 2016 23:07:04 GMT
View Forum Message <> Reply to Message

On 1/22/16 1:04 PM, KH wrote:
> On Friday, January 22, 2016 at 2:48:38 PM UTC-5, Mike Galloy wrote:
>> On 1/22/16 12:26 PM, kimberly.hyde@noaa.gov wrote:
>>> On Friday, January 22, 2016 at 2:20:25 PM UTC-5, kimber...@noaa.gov
>>> wrote:
>>>> On Thursday, January 21, 2016 at 10:26:36 PM UTC-5, Michael Galloy
>>>> wrote:
>>>> > On 1/21/16 1:21 PM, KH wrote:
>>>> >> Hello,
>>>> >>
>>>> >> Is there a netcdf equivalent to HDF_ISHDF?  I am basically
>>>> >> looking for a way to test if a file is a netcdf file before
>>>> >> trying to open the file (and the having the program crash
>>>> >> because it isn't actually a netcdf file).
>>>> >
>>>> > Yes, I would normally recommend just trying to open the file and
>>>> > catching the error if it fails, but, as you noted, it actually
>>>> > crashes IDL:
>>>> >
>>>> > IDL> id = ncdf_open(filename) Assertion failed: (0), function
>>>> > NC4_open, file nc4file.c, line 2696. Abort trap: 6
>>>> >
>>>> > Are your files netCDF 4 files? You can use H5F_IS_HDF5 if you
>>>> > don't want pre-4 netCDF files:
>>>> >
>>>> > IDL> help, h5f_is_hdf5(file_which('ncgroup.nc')) <Expression>
>>>> > LONG      =          1 IDL> help,
>>>> > h5f_is_hdf5(file_which('sample.nc')) <Expression>    LONG     =
>>>> > 0
>>>> >
>>>> >
>>>> > Mike -- Michael Galloy www.michaelgalloy.com Modern IDL: A Guide
>>>> > to IDL Programming (http://modernidl.idldev.com)
>>>>
>>>> These particular files are netcdf4 so I can use the H5f program,

>>>> but not all of my files are netcdf4 so I was hoping to find a way
>>>> to automatically test what type of files they are. Thanks for your
>>>> assistance, Kim
>>>
>>>
>>> In the end, I am mainly trying to quickly determine if the netcdf
>>> files I downloaded are corrupt or incomplete.  That way if they are
>>> bad, I can remove them so that they don't crash during subsequent
>>> processing.  Currently I am opening the file and then using
>>> NCDF_INQUIRE to get the basic structure information, but if there is
>>> an easier way to check the file, I am open for ideas.
>>>
>>> Kim
>>>
>>
>> I have a routine, MG_NC_ISNCDF, in my library mglib (at
>> github.com/mgalloy/mglib). You have to build the library to get the
>> routine (and, of course, put the location of where the DLM is installed
>> into your IDL DLM path). You will need the netCDF, HDF5, curl, libsz,
>> and libz libraries installed. Change the paths to the proper locations,
>> but something like the following should work for your system:
>>
>> mkdir build
>> cd build
>>
>> cmake \
>>     -DCMAKE_INSTALL_PREFIX:PATH=~/software/mglib \
>>     -DNETCDF_INCLUDE_DIR:PATH=/usr/local/include \
>>     -DNETCDF_LIBRARY:PATH=/usr/local/lib/libnetcdf.a \
>>     -DHDF5_LIBRARY:PATH=/usr/local/lib/libhdf5.a \
>>     -DHDF5_LA_LIBRARY:PATH=/usr/local/lib/libhdf5_hl.a \
>>     -DCURL_LIBRARY:PATH=/usr/lib/libcurl.dylib \
>>     -DSZ_LIBRARY:PATH=/usr/local/lib/libsz.a \
>>     -DZ_LIBRARY:PATH=/usr/lib/libz.dylib \
>>     ..
>>
>> Mike
>> --
>> Michael Galloy
>> www.michaelgalloy.com
>> Modern IDL: A Guide to IDL Programming (http://modernidl.idldev.com)
>
> Hello Mike,
> Is this a C program or IDL? It definitely isn't clear to me how to
implement this. All I really know is basic IDL and the code in the
program doesn't look familiar to me.
> Kim
>

It's a DLM written in C that can be called from IDL. The code above is what you would type at the (system) command line, after grabbing the source code from my library, to configure on your system.

Mike
--
Michael Galloy
www.michaelgalloy.com
Modern IDL: A Guide to IDL Programming (http://modernidl.idldev.com)

---

Subject: Re: NCDF_IS_NCDF
Posted by benjaminforeback on Tue, 09 Feb 2016 22:08:25 GMT
View Forum Message <> Reply to Message

Hi folks,

Coincidentally, I have the need for this same routine in some ENVI development, so I've written a NCDF_IS_NCDF function in IDL, which will be available in the next IDL release. Basically, according to the NCDF FAQ page, under the section "How can I tell which format a netCDF file uses?" the first four bytes of the file define a NetCDF-3 file. The first three bytes are the characters "CDF" and the fourth is either 1 or 2, depending on whether it is a "classic" NCDF3 file or a 64-bit offset file. Below is the code, which when it's released it will be in the idl lib directory. I hope this helps.

```
;--
; Copyright (c) 2002-2016, Exelis Visual Information Solutions, Inc. All
;   rights reserved. Unauthorized reproduction is prohibited.
;+
; NAME:
;   NCDF_IS_NCDF
;
; PURPOSE:
;   This function determines if a file (or files) is NetCDF-3 format.
;
;   This function is modeled after the H5F_IS_HDF5 and HDF_ISHDF functions.
;
; SYNTAX:
;
;   Result = NCDF_IS_NCDF(filenames)
;
; RETURN VALUE:
;   NCDF_IS_NCDF returns 1 (true) if the file exists and is NetCDF-3 format,
;   0 (false) otherwise. If the input argument is an array of filenames, then
;   an array of 1 or 0 values will be returned.
;
; INPUTS:
```

```
;
;   Filename (required): A string or array of strings containing the filename
;   to check.
;
;
;
; KEYWORD PARAMETERS:
;
;   None.
;
;
; MODIFICATION HISTORY:
;   Written by: Ben Foreback, Harris, February 2016
;

FUNCTION NCDF_IS_NCDF, filenames
  COMPILE_OPT IDL2, HIDDEN

  ON_ERROR, 2

  nFiles = N_ELEMENTS(filenames)
  IF nFiles EQ 0 THEN BEGIN
    MESSAGE, 'Incorrect number of arguments.'
  ENDIF

  IF~ISA(filenames, /STRING) THEN BEGIN
    MESSAGE, 'Filename must be a string.'
  ENDIF

  ; Start with a FILE_TEST on all of the files; if a given file doesn't exist,
  ; then it is, of course, not a NetCDF file.
  isNCDF = FILE_TEST(filenames)

  ; Define a catch block in order to make absolute sure that the LUN will always
  ; be freed.
  CATCH, err
  IF (err NE 0) THEN BEGIN
    CATCH, /CANCEL
    IF (N_ELEMENTS(unit) GT 0) THEN BEGIN
      FREE_LUN, unit
    ENDIF
    MESSAGE, /REISSUE_LAST
  ENDIF

  ; To determine if the file is a NetCDF-3 file, check the first four bytes
  ; of the file. The first three bytes will be "CDF" and the fourth is either
  ; the byte value 1 or the byte value 2. This comes from the NetCDF FAQ page
  ; found at http://www.unidata.ucar.edu/software/netcdf/docs/faq.html under
  ; the section "How can I tell which format a netCDF file uses?" (site
  ; accessed 4 February 2016).
```

```
header = BYTARR(4, /NOZERO)
validCDFHeader1 = [67B, 68B, 70B, 1B]
validCDFHeader2 = [67B, 68B, 70B, 2B]
FOR i = 0, nFiles - 1 DO BEGIN
  IF ~isNCDF[i] THEN CONTINUE
  OPENR, unit, filenames[i], /GET_LUN
  READU, unit, header
  FREE_LUN, unit
  IF ~ARRAY_EQUAL(header, validCDFHeader1) && $
    ~ARRAY_EQUAL(header, validCDFHeader2) THEN isNCDF[i] = 0B
ENDFOR

  RETURN, isNCDF

END
```