Subject: Re: HDF object/tag retrieval
Posted by David Fanning on Tue, 18 Sep 2007 13:17:42 GMT
View Forum Message <> Reply to Message

leatherback writes:

> When I use the HDF_SD_ATTRINFO command, all the global attribute are
> there, but the individual tags cannot be accessed: All data is just
> one big string of nested information.

Are you sure of that? Do you think that because you
tried to print the results? As far as I know, this
function returns a structure. Did you try accessing
the fields of this structure?

Cheers,

David

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

Subject: Re: HDF object/tag retrieval
Posted by leatherback on Tue, 18 Sep 2007 14:12:57 GMT
View Forum Message <> Reply to Message

Hi David,

Yes. I am sure. The function returns a variable of type string:

```
for i=0,nglobatts-1 do begin
   HDF_SD_ATTRINFO,sd_id,i,name=n,type=t,count=c,data=d
   if (t eq 'STRING' ) then print,strtrim(n),t,c,d[0,0],FORMAT=FS
else $
   if (t eq 'FLOAT' )  then print,strtrim(n),t,c,d[0,0],FORMAT=FF
else $
   if (t eq 'DOUBLE' ) then print,strtrim(n),t,c,d[0,0],FORMAT=FD
else $
   print,strtrim(n),t,c,d[0,0],FORMAT=FI
endfor
```

ENVI> help, d

D STRING = '
GROUP = ARCHIVEDMETADATA
GROUPTYPE = MASTERGROUP

The actual string returned is roughly 11000 chars long, which indicates it holds all the individual tags. When printed, the string is structures in an XML kind-a structure (Nested meta data format).

Any suggestions?

Cheers,

Jelle.

On 18 Sep, 14:17, David Fanning <n...@dfanning.com> wrote:
> Are you sure of that? Do you think that because you
> tried to print the results? As far as I know, this
> function returns a structure. Did you try accessing
> the fields of this structure?
>
> Cheers,
>
> David

Subject: Re: HDF object/tag retrieval
Posted by David Fanning on Tue, 18 Sep 2007 14:19:30 GMT
View Forum Message <> Reply to Message

leatherback writes:

> Yes. I am sure. The function returns a variable of type string:
>
> [/color]
[color=blue]> for i=0,nglobatts-1 do begin[/color]
[color=blue]> HDF_SD_ATTRINFO,sd_id,i,name=n,type=t,count=c,data=d[/color]
[color=blue]> if (t eq 'STRING') then print,strtrim(n),t,c,d[0,0],FORMAT=FS[/color]
[color=blue]> else $[/color]
[color=blue]> if (t eq 'FLOAT') then print,strtrim(n),t,c,d[0,0],FORMAT=FF[/color]
[color=blue]> else $[/color]
[color=blue]> if (t eq 'DOUBLE') then print,strtrim(n),t,c,d[0,0],FORMAT=FD[/color]
[color=blue]> else $[/color]
[color=blue]> print,strtrim(n),t,c,d[0,0],FORMAT=FI[/color]
[color=blue]> endfor[/color]
[color=blue]> [/color]
>
> ENVI> help, d

> D STRING = '
> GROUP = ARCHIVEDMETADATA
> GROUPTYPE = MASTERGROUP
>
> The actual string returned is roughly 11000 chars long, which
> indicates it holds all the individual tags. When printed, the string
> is structures in an XML kind-a structure (Nested meta data format).
>
> Any suggestions?

Ah, well, I guess I would switch to Liam's HDF_SD_ATTRINFO
function. :-)

Cheers,

David

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

Subject: Re: HDF object/tag retrieval
Posted by leatherback on Tue, 18 Sep 2007 14:25:37 GMT
View Forum Message <> Reply to Message

ehmm.. Is that different from the one in IDL itself? (See the code I
pasted)

> Ah, well, I guess I would switch to Liam's HDF_SD_ATTRINFO
> function. :-)
>
> Cheers,
>
> David
>
> --
> David Fanning, Ph.D.
> Fanning Software Consulting, Inc.
> Coyote's Guide to IDL Programming:http://www.dfanning.com/
> Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: HDF object/tag retrieval
Posted by David Fanning on Tue, 18 Sep 2007 14:26:37 GMT

David Fanning writes:

> Ah, well, I guess I would switch to Liam's HDF_SD_ATTRINFO
> function. :-)

Incidentally, I couldn't live without these HDF routines
Liam has written, but I don't see how to get them. (And
I can't remember how *I* got them!) They have the standard
GNU General Public License, and a reference to Liam's web
page. But I just can't find the programs there.

Does anyone know how the general public can get them?

Cheers,

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

Subject: Re: HDF object/tag retrieval
Posted by liamgumley on Tue, 18 Sep 2007 14:26:59 GMT

Dear Jelle,

The following IDL function is designed to extract a single PVL object
from a PVL formatted string, such as the CoreMetadata.0 global
attribute in MODIS HDF files.

ftp://ftp.ssec.wisc.edu/pub/incoming/get_metadata.pro

Here's an example. Note that hdf_sd_attinfo.pro is available from
http://gumley.com/PIP/Free_Software.html.

```
; Read a global attribute containing global metadata
file = 'MOD021KM.A2000115.1710.002.2000119195542.hdf'
hdfid = hdf_sd_start(file)
info = hdf_sd_attinfo(hdfid, '', 'CoreMetadata.0', /global)
hdf_sd_end, hdfid

; Extract a PVL object
pvlstring = info.data
```

objstring = 'RANGEBEGINNINGDATE'
result = get_metadata(pvlstring, objstring)
print, result

On a final note, you might be able to find a better tool for creating
GeoTIFF files from the MYD09A1 product. Here are a few options.

HDFLook
http://www-loa.univ-lille1.fr/Hdflook/hdflook_gb.html

MRTSwath
http://edcdaac.usgs.gov/landdaac/tools/mrtswath/

HEG
http://newsroom.gsfc.nasa.gov/sdptoolkit/HEG/HEGHome.html

Cheers,
Liam.
Practical IDL Programming
http://www.gumley.com/

Subject: Re: HDF object/tag retrieval
Posted by leatherback on Tue, 18 Sep 2007 15:54:57 GMT
View Forum Message <> Reply to Message

Dear Liam,

Thank you for your reply.

I see the solution you are offering. I'll give that a go. A subproblem
problem I have, is that all the functions I can find in IDL to
retrieve tag values, require you to know the tag ID and/or string to
retrieve the value. But I have not been able to find a way to create a
listing of these values. Would you know how to get this out?

I tried the HEG tool, which I was unable to get to work in batch mode.
I'll have a look at the other tools. Ideally I run the IDL route, as
all my processing at the moment is done in IDL, and it allows me to
have one linux box downloading my timeseries/images as they become
available, and then have IDL check whether newer data is available,
and process the lot whenever i need updated info.

Cheers,

Jelle.

On Sep 18, 3:26 pm, liamgum...@gmail.com wrote:

> Dear Jelle,
>
> The following IDL function is designed to extract a single PVL object
> from a PVL formatted string, such as the CoreMetadata.0 global
> attribute in MODIS HDF files.
>
> ftp://ftp.ssec.wisc.edu/pub/incoming/get_metadata.pro
>
> Here's an example. Note that hdf_sd_attinfo.pro is available
fromhttp://gumley.com/PIP/Free_Software.html.
>
> ; Read a global attribute containing global metadata
> file = 'MOD021KM.A2000115.1710.002.2000119195542.hdf'
> hdfid = hdf_sd_start(file)
> info = hdf_sd_attinfo(hdfid, '', 'CoreMetadata.0', /global)
> hdf_sd_end, hdfid
>
> ; Extract a PVL object
> pvlstring = info.data
> objstring = 'RANGEBEGINNINGDATE'
> result = get_metadata(pvlstring, objstring)
> print, result
>
> On a final note, you might be able to find a better tool for creating
> GeoTIFF files from the MYD09A1 product. Here are a few options.
>
> HDFLookhttp://www-loa.univ-lille1.fr/Hdflook/hdflook_gb.html
>
> MRTSwathhttp://edcdaac.usgs.gov/landdaac/tools/mrtswath/
>
> HEGhttp://newsroom.gsfc.nasa.gov/sdptoolkit/HEG/HEGHome.html
>
> Cheers,
> Liam.
> Practical IDL Programminghttp://www.gumley.com/

Subject: Re: HDF object/tag retrieval
Posted by James Kuyper on Tue, 18 Sep 2007 16:17:27 GMT
View Forum Message <> Reply to Message

leatherback wrote:
> Hi David,
>
> Yes. I am sure. The function returns a variable of type string:
>
> [/color]
[color=blue]> for i=0,nglobatts-1 do begin[/color]

[color=blue]>   HDF_SD_ATTRINFO,sd_id,i,name=n,type=t,count=c,data=d[/color]
[color=blue]>   if (t eq 'STRING') then print,strtrim(n),t,c,d[0,0],FORMAT=FS[/color]
[color=blue]> else $[/color]
[color=blue]>   if (t eq 'FLOAT') then print,strtrim(n),t,c,d[0,0],FORMAT=FF[/color]
[color=blue]> else $[/color]
[color=blue]>   if (t eq 'DOUBLE') then print,strtrim(n),t,c,d[0,0],FORMAT=FD[/color]
[color=blue]> else $[/color]
[color=blue]>   print,strtrim(n),t,c,d[0,0],FORMAT=FI[/color]
[color=blue]> endfor[/color]
[color=blue]> [/color]
>
> ENVI> help, d
> D   STRING   = '
> GROUP   = ARCHIVEDMETADATA
>   GROUPTYPE   = MASTERGROUP
>
> The actual string returned is roughly 11000 chars long, which
> indicates it holds all the individual tags. When printed, the string
> is structures in an XML kind-a structure (Nested meta data format).

What you're looking at is the ECS archived metadata string, which is stored in the HDF file attribute named "ArchivedMetadata.0". There's also an ECS inventory metadata string, which is stored in the HDF file attribute named "CoreMetadata.0". ECS metadata < http://observer.gsfc.nasa.gov/ECSInfo/ecsmetadata/index.html> is stored in Object Description Language (ODL) format <http://pds.jpl.nasa.gov/documents/qs/ODL.html>. It is supposed to be part of all EOS data products.

Liam Gumley's routine will handle almost all ordinary cases where you want to read ECS metadata. However, it will not handle correctly those unlikely cases where an ECS object's name or value contain the string "VALUE" or "END_OBJECT", or where an ECS metadata string is too big to fit in an single HDF attribute. In that latter case, the ECS metadata is split among multiple HDF file attributes named, for example "CoreMetadata.0.0", "CoreMetadata.0.1", etc. (I strongly suspect that the ".0" at the end of "CoreMetadata.0" was the result of misunderstanding of how this file attribute splitting would be done).

Somebody might have written IDL code to fully and correctly parse ECS metadata, but I'm not aware of any such program. The SDP Toolkit library <http://newsroom.gsfc.nasa.gov/sdptoolkit/toolkit.html> provides C and Fortran routines for writing and reading ECS metadata; the C routines all have names that start with "PGS_MET_".