Subject: Reading HDF5 Compound Datasets in IDL
Posted by Justin Bronn on Wed, 18 May 2005 16:48:32 GMT
View Forum Message <> Reply to Message

Hello All,

I'm stumped. Here's the situation: I have an HDF5 dataset that I want to read, and I cannot figure out from the IDL documentation how to selectively read only parts of the dataset. Read on for the gory details...

The HDF5 dataset is relatively simple; there are 4 groups, each containing a compound data type (the HDF5 compound data type is analagous to the IDL struct). There can be N-elements of this compound data type (again, like an array of IDL structures). The compound data type contains 4 different fields: a filename, a time-stamp, and three floating point arrays. Below is a representation of this HDF5 file with the equivalent IDL datatypes.

```
/--+
  |- Group_1
  |    |
  |    |- compound (Can have N number of elements
  |         |     => IDL Structure Array)
  |         |
  |         |--- filename (String, 256 Characters long
  |         |            => IDL String)
  |         |--- time    (64-bit Float Value
  |         |            => IDL Double)
  |         |--- data1   (32-bit Float Array, 4096 Elements
  |         |             => IDL Float Array)
  |         |--- data2   (32-bit Float Array, 4096 Elements
  |         |             => IDL Float Array)
  |         |--- data3   (32-bit Float Array, 4096 Elements
  |         |             => IDL Float Array)
 ...
  |
  |- Group_4 (Same as Group_1)
```

Now if I want to read an entire one of these compound data types into IDL, here's what I can do:

```
;; Opening up the necessary HDF5 file IDs
h5fid = h5f_open('data.h5')
h5gid = h5g_open(h5fid, 'Group_1')
h5did = h5d_open(h5gid, 'compound')

;; Reading the data
```

```
data = h5d_read(h5did)

;; Cleaning up
h5d_close, h5did & h5g_close, h5gid & h5f_close, h5fid
```

When I do a 'help' on the data read in, I get exactly what I expected:

```
IDL> help, data
DATA            STRUCT    = -> <Anonymous> Array[263]
IDL> help, data, /ST
** Structure <8225794>, 5 tags, length=49172, data length=49172,
refs=1:
   FILENAME       STRING    '/path/to/ascii_file'
   TIME           DOUBLE          2452305.5
   DATA1          FLOAT     Array[4096]
   DATA2          FLOAT     Array[4096]
   DATA3          FLOAT     Array[4096]
IDL>
```

Now here's the problem: I do not want to have to read the ENTIRE compound data type. When the number of compound elements gets large (say N=3000), the read operation takes a _long_ time since the entire compound data type is read into memory. I want to selectively read only _portions_ of the compound data type, like the 'time' element, to determine what I only really need, and then read out that selection.

However, with the IDL HDF5 API I can figure out how to get _information_ on each of the data types, including the member name, class, type, and byte offset:

```
;; Opening up the necessary HDF5 file IDs
h5fid = h5f_open('data.h5')
h5gid = h5g_open(h5fid, 'Group_1')
h5tid = h5t_open(h5gid, 'compound')

;; Getting the number of members in the compund data type
n_mems = h5t_get_nmembers(h5tid)
print, n_mems

for i = 0, n_mems-1 do begin
   ;; Getting the name & HDF5 data type 'class'
   name = h5t_get_member_name(h5tid, i)
   class = h5t_get_member_class(h5tid, i)

   :; Getting the 'byte offset' of the compound type id.
   offset = h5t_get_member_offset(h5tid, i)
   typeid = h5t_get_member_type(h5tid, i)
```

```
   print, name, class, offset, typeid, FORMAT='(2A12,I10,I13)'

   ;; Closing the type id
   h5t_close, type
endfor

;; Cleaning up
h5f_close, h5fid & h5g_close, h5gid & h5t_close, h5tid
```

Running the previous code will produce this output:

```
    filename  H5T_STRING        0    268438443
        time   H5T_FLOAT      256    268438444
       data1   H5T_ARRAY      264    268438445
       data2   H5T_ARRAY    16648    268438446
       data3   H5T_ARRAY    33032    268438447
```

I have gotten this far, however I cannot figure out how to get a dataspace (using the IDL H5S_* routines) that will select only one of these elements. The IDL HDF5 API is rather cryptic, and if anyone out there has an experience or a suggestion on how I may accomplish this task please let me know!!

Thanks,
-Justin
