RFC: Supporting Display of Region References in H5dump Tool

Allen Byrne

This RFC proposes a method to display data associated with a dataset region reference using the h5dump tool.

1. Region Reference Introduction

Currently, the h5dump tool only describes the path of the dataset referenced and the selected elements. The elements are either individual coordinates or corner coordinates of a hyperslab's upper left and lower right. This section introduces region references to those unfamiliar with concept. Those with region reference experience are invited to scan ahead to section 2.

A selection can be used to create a pointer to a set of selected elements of a *dataset*, called a region reference. The selection can be either a point selection or a hyperslab selection.

A region reference is a datatype maintained by the HDF5 Library. The region reference can be stored in a dataset or attribute, and then read. The dataset or attribute is defined to have the special datatype, H5T_STD_REF_DSETREG.

To discover the elements and/or read the data, the region reference can be deferenced. The H5Rdefrerence call returns a handle for the *dataset*, and then the selected dataspace can be retrieved with H5Rget_select call. The selected *dataspace* can be used to read the selected data elements.

A region reference points to a dataset and a region within that dataset. When stored in a dataset, an array of region references can provide a unified view of the data stored in the different datasets in a file.

In general, region references are useful for directly accessing a portion of a dataset. Notably, region references play an important role in large datasets by providing a convenient and efficient way to point to data of interest.

Figure 1 illustrates the concept of region references. A user can create a file, *FileA.h5*, having a group (*Group_1D*) containing a dataset with data values stored in a one dimensional array (*DS1*), a group (*Group_2D*) containing a dataset with data in a two dimensional array (*DS2*), and a group (*Group_3D*) containing a dataset with data in a three dimensional array (*DS3*). In order to quickly and efficiently access data within a dataset, an array of region references is created (*R1*). Each element in the region reference array points to a different selection of elements in the datasets.

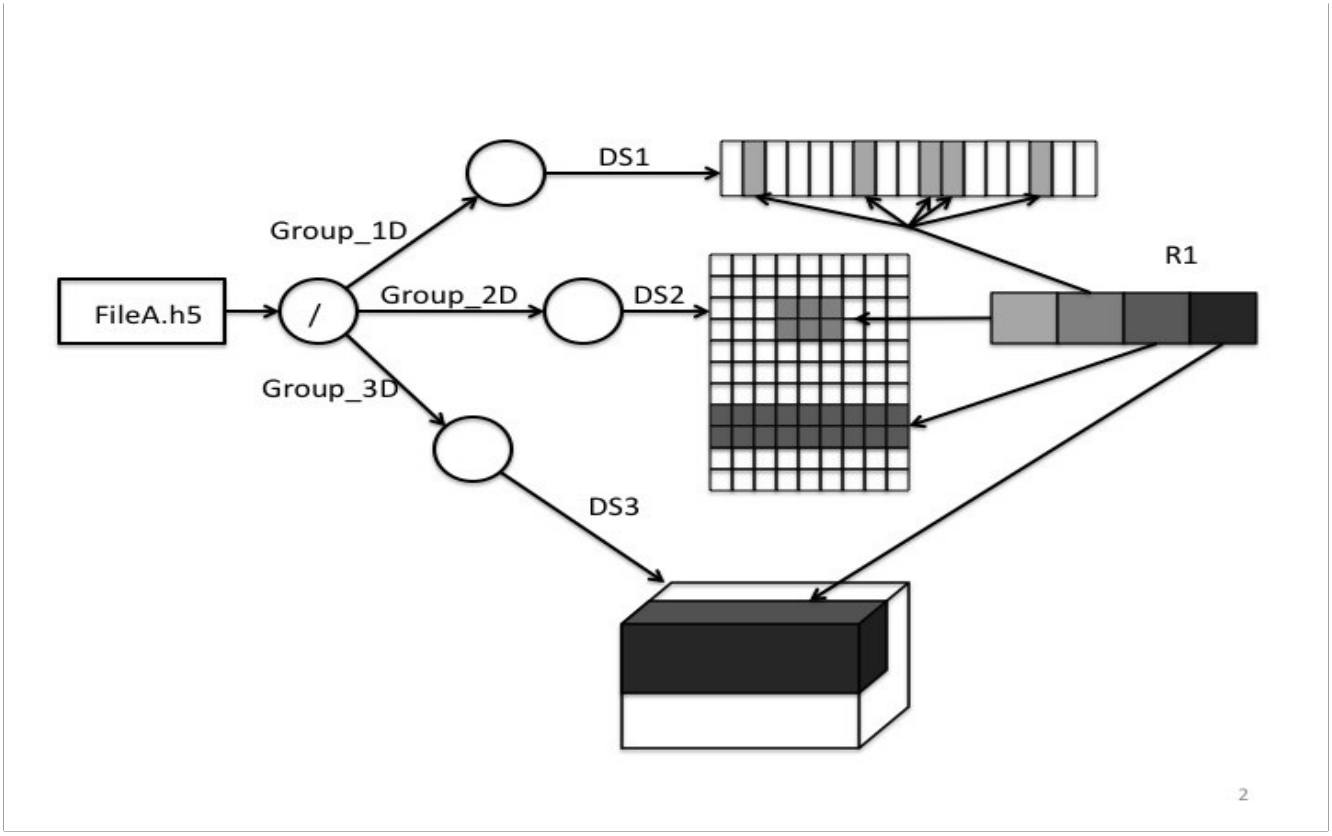

Figure 1

Below is an output of the h5dump utility (version 1.8) run on the FileA.h5 file depicted above. In the output the three groups are listed along with each group's dataset and the dataset R1 with region references. Each element of R1 is displayed as a path to a referenced dataset and a description of the selected elements. Point selection in one-dimensional dataset DS1 is displayed as a list of the selected elements coordinates; selections in the two and three-dimensional datasets, DS2 and DS3, are displayed as lists of hyperslab's upper left and low right corners coordinates. Coordinates of the elements are 0-based.

```
HDF5 "FileA.h5" {
GROUP "/" {
   GROUP "Group_1D" {
      DATASET "DS1" {
         DATATYPE  H5T_STD_I32LE
         DATASPACE  SIMPLE { ( 17 ) / ( 17 ) }
         DATA { …
                  }
      }
   }
   GROUP "Group_2D" {
      DATASET "DS2" {
         DATATYPE  H5T_STD_I32LE
         DATASPACE  SIMPLE { ( 9, 8 ) / ( 9, 8 ) }
         DATA { …
         }
```

```
      }
   }
   GROUP "Group_3D" {
      DATASET "DS3" {
         DATATYPE  H5T_STD_I32LE
         DATASPACE  SIMPLE { ( 6, 6, 6 ) / ( 6, 6, 6 ) }
         DATA { …
         }
      }
   }
   DATASET "R1" {
      DATATYPE  H5T_REFERENCE
      DATASPACE  SIMPLE { ( 4 ) / ( 4 ) }
      DATA {
       (0):  DATASET /Group_1D/DS1 {(1), (6), (9), (10), (14)},
       (1):  DATASET /Group_2D/DS2 {(3,3)-(5,4)},
       (2):  DATASET /Group_2D/DS2 {(0,5)-(8,6)},
       (3):  DATASET /Group_3D/DS3 {(3,3,3)-(4,4,4)}
      }
   }
}
}
```

2. Command Option, Display Format

In order for h5dump to display the data associated with region references, there must be a command line option and data display format defined. The following command line option is recommended:

```
  -R, --region    Print dataset pointed by region references
```

The recommendation for the data display format is that the referenced data be displayed immediately after the dataset path and element coordinates. For the example above the output from h5dump run on file, FileA.h5:

```
HDF5 "FileA.h5" {
GROUP "/" {
   GROUP "Group_1D" {
      DATASET "DS1" {
         DATATYPE  H5T_STD_I32LE
         DATASPACE  SIMPLE { ( 17 ) / ( 17 ) }
         DATA { …
         }
      }
   }
   GROUP "Group_2D" {
      DATASET "DS2" {
         DATATYPE  H5T_STD_I32LE
         DATASPACE  SIMPLE { ( 9, 8 ) / ( 9, 8 ) }
         DATA { …
         }
      }
   }
   GROUP "Group_3D" {
      DATASET "DS3" {
         DATATYPE  H5T_STD_I32LE
```

```
         DATASPACE  SIMPLE { ( 6, 6, 6 ) / ( 6, 6, 6 ) }
         DATA { …
         }
      }
   }
   DATASET "R1" {
      DATATYPE  H5T_REFERENCE
      DATASPACE  SIMPLE { ( 4 ) / ( 4 ) }
      DATA {
      (0): DATASET /Group_1D/DS1 {
         (0): REGION_TYPE POINT  (1), (6), (9), (10), (14)
         (0): DATATYPE  H5T_STD_I32LE
         (0): DATASPACE  SIMPLE { ( 17 ) / ( 17 ) }
         (0): DATA {
            (1): 1,
            (6): 6,
            (9): 9,
            (10): 10,
            (14): 14
         (0):  }
      (0): }
      (1): DATASET /Group_2D/DS2  {
         (1): REGION_TYPE BLOCK  (3,3)-(5,4)
         (1): DATATYPE  H5T_STD_I32LE
         (1): DATASPACE  SIMPLE { ( 9, 8 ) / ( 9, 8 ) }
         (1): DATA {
            (3,3): 65, 45,
            (4,3): 54, 55,
            (5,3): 64, 65
         (1):  }
      (1): }
      (2): DATASET /Group_2D/DS2  {
         (2): REGION_TYPE BLOCK  (0,5)-(8,6)
         (2): DATATYPE  H5T_STD_I32LE
         (2): DATASPACE  SIMPLE { ( 9, 8 ) / ( 9, 8 ) }
         (2): DATA {
            (0,5): 16, 17,
            (1,5): 26, 27,
            (2,5): 36, 37,
            (3,5): 46, 47,
            (4,5): 56, 57,
            (5,5): 11, 12,
            (6,5): 13, 21,
            (7,5): 22, 23,
            (8,5): 96, 97
         (2):  }
      (2): }
      (3): DATASET /Group_3D/DS3  {
         (3): REGION_TYPE BLOCK  (3,3,3)-(4,4,4)
         (3): DATATYPE  H5T_STD_I32LE
         (3): DATASPACE  SIMPLE { ( 6, 6, 6 ) / ( 6, 6, 6 ) }
         (3): DATA {
            (3,3,3): 444, 445,
            (3,4,3): 454, 455,
            (4,3,3): 544, 545,
            (4,4,3): 554, 555
         (3):  }
      (3): }
      }
   }
}
}
```

3. Additions to Output

```
(0): DATASET /Group_1D/DS1 {(1), (6), (9), (10), (14)},
```

to

```
(0): DATASET /Group_1D/DS1 {
   (0): REGION_TYPE POINT  (1), (6), (9), (10), (14)
   (0): DATATYPE  H5T_STD_I32LE
   (0): DATASPACE  SIMPLE { ( 17 ) / ( 17 ) }
   (0): DATA {
      (1): 1,
      (6): 6,
      (9): 9,
      (10): 10,
      (14): 14
   (0):  }
(0): }
```

Added `REGION_TYPE [ POINT | BLOCK ]` in between the '`{`' and the selection dimensions. The `DATATYPE` and `DATASPACE` of the selection is displayed as in normal display of `DATASET`. Finally the actual data is displayed between
'`DATA {`' and '`}`'. It is suggested that the data be displayed prefixed by the selection indices. For example;

```
(3): DATASET /Group_3D/DS3  {
   (3): REGION_TYPE BLOCK  (3,3,3)-(4,4,4)
   (3): DATATYPE  H5T_STD_I32LE
   (3): DATASPACE  SIMPLE { ( 6, 6, 6 ) / ( 6, 6, 6 ) }
   (3): DATA {
      (3,3,3): 444,445,454,455,544,545,554,555
   (3):  }
(3): }
```

would be displayed as:

```
(3): DATASET /Group_3D/DS3  {
   (3): REGION_TYPE BLOCK  (3,3,3)-(4,4,4)
   (3): DATATYPE  H5T_STD_I32LE
   (3): DATASPACE  SIMPLE { ( 6, 6, 6 ) / ( 6, 6, 6 ) }
   (3): DATA {
      (3,3,3): 444, 445,
      (3,4,3): 454, 455,
      (4,3,3): 544, 545,
      (4,4,3): 554, 555
   (3):  }
(3): }
```

4. Combining the Region Option with Other Options

The -R (--region) option is expected to be combined with other h5dump options. First, the following options will ignore the region option:

```
-n, --contents       Print a list of the file contents and exit
```

```
-B, --superblock     Print the content of the super block
-H, --header         Print the header only; no data is displayed
-V, --version        Print version number and exit
-x, --xml            Output in XML using Schema
-u, --use-dtd        Output in XML using DTD
-D U, --xml-dtd=U    Use the DTD or schema at U
-X S, --xml-ns=S      (XML Schema) Use qualified names n the XML
```

-f D, --filedriver=D Specify which driver to open the file with

The region reference option can be used with these options:

```
-h, --help           Print a usage message and exit
-A, --onlyattr       Print the header and value of attributes
-i, --object-ids     Print the object ids
-r, --string         Print 1-byte integer datasets as ASCII
-e, --escape         Escape non printing characters
-o F, --output=F     Output raw data into file F

-a P, --attribute=P  Print the specified attribute
-d P, --dataset=P    Print the specified dataset
-y, --noindex        Do not print array indices with the data
-p, --properties     Print dataset filters, storage layout and fill value
-g P, --group=P      Print the specified group and all members
-l P, --soft-link=P  Print the value(s) of the specified soft link
-t P, --datatype=P   Print the specified named datatype
-w N, --width=N      Set the number of columns of output
-m T, --format=T     Set the floating point output format
-q Q, --sort_by=Q    Sort groups and attributes by index Q
-z Z, --sort_order=Z Sort groups and attributes by order Z
```

The region reference option output will be modified by these options:

```
-b B, --binary=B     Binary file output, of form B
```

```
Subsetting is available by using the following options with a dataset
attribute. Subsetting is done by selecting a hyperslab from the data.
Thus, the options mirror those for performing a hyperslab selection.
The START and COUNT parameters are mandatory if you do subsetting.
```

```
The STRIDE and BLOCK parameters are optional and will default to 1 in
each dimension.

      -s L, --start=L     Offset of start of subsetting selection
      -S L, --stride=L    Hyperslab stride
      -c L, --count=L     Number of blocks to include in selection
      -k L, --block=L     Size of block in hyperslab
```

To specify a subset of a region reference, the -d or -a option must be used. Then the subset parameters will be applied against the dataspace of the H5T_REFERENCE as follows:

```
      -s L, --start=L     Offset of start of reference dataset selection
      -S L, --stride=L    Hyperslab stride
      -c L, --count=L     Number of datasets to include in selection
      -k L, --block=L     Size of block in hyperslab
```

For example;

```
>H5dump -R -d /R1 -s 3 -c 1 FileA.h5
HDF5 "FileA.h5" {
DATASET "/R1" {
   DATATYPE  H5T_REFERENCE { H5T_STD_REF_DSETREG }
   DATASPACE  SIMPLE { ( 4 ) / ( 4 ) }
   SUBSET {
      START ( 3 );
      STRIDE ( 1 );
      COUNT ( 1 );
      BLOCK ( 1 );
      DATA {
      (3): DATASET /Group_3D/DS3 {
         (3): REGION_TYPE BLOCK  (3,3,3)-(4,4,4)
         (3): DATATYPE  H5T_STD_I32LE
         (3): DATASPACE  SIMPLE { ( 6, 6, 6 ) / ( 6, 6, 6 ) }
         (3): DATA {
            (3,3,3): 444, 445,
            (3,4,3): 454, 455,
            (4,3,3): 544, 545,
            (4,4,3): 554, 555
         (3):  }
      (3): }
      }
   }
}
}
```

5. BNF Issues

The current h5dump does not display the H5T_REFERENCE line according to the documented BNF. The ddl specifies:

```
<reference> ::= H5T_REFERENCE { <ref_type> }
```

```
<ref_type> ::= H5T_STD_REF_OBJECT | H5T_STD_REF_DSETREG
```

for backward compatibility reasons, this will only be used by the -R option. Also, the following BNF alterations will be necessary:

```
<data_region_data> ::= H5T_STD_REF_DSETREG <object_num> {
<data_region_data_list> }
```

will be changed to match the current output as follows:

```
<data_region_data> ::= DATASET <dataset_name> {
<data_region_type>opt <data_region_data_list> <dataset_type>opt
<dataset_space>opt <data>opt }
```

```
<data_region_type> ::= REGION_TYPE <data_region_data_type>
```

```
<data_region_data_type> ::= POINT | BLOCK
```

6. Effects to Other Tools

The h5ls tool would also need to incorporate the same option as it uses the same library calls as h5dump. In addition, h5ls adds a prefix (Ptn: or Blkn:, where n is a 0-based number) to the coordinates within the dataset {}. Also h5ls shortens the DATASET to DSET.

7. Recommendation

The implementation of this option requires document changes to both the h5dump and h5ls tools. When this option is selected, the output will show the data that is being referenced. It is yet to be decided if or how references beyond the first region reference will be handled. This may require an additional option that specifies recursive calls into additional references. Also, should there be additional option modifier to display the datatype and dataspace of the region data or just print the datatype and dataspace before the data block. There is also a concern that gathering the information about region references can slow the performance of the h5dump tool. Therefore, there is a question of adding an option to limit the display of region reference information a single line display of the dataset type.

Acknowledgements

This work was partially supported by The HDF GROUP and NPOESS contract.

Revision History

April 7, 2009: Version 1 circulated for comment within The HDF Group.

April 22, 2009: Additional note added about the display changes and suggested prefix identification.

April 27, 2009: Basics section 6 replaces section 1. Added note for advanced readers. Reworked suggested output format.

April 29, 2009: Version 2 changed suggested output format to match the actual output of h5dump.

May 6, 2009: Version 3 added the subset example and the actual output of h5dump. Also, changed sentences using the word combined.

May 8, 2009: Version 3.1 corrected text in section one to indicate that groups contain datasets, which have data values.

May 12, 2009: Version 3.2 added the dataspace line after the datatype line to the display of region references. Also added the question of needing an option to not display region reference information.

[References]

RFC THG 2009-03-25 **RFC: Support Reg. Ref. and Bitfield in HDFView** by Peter Cao

RFC THG 2009-03-23 **RFC: High-Level Functions for Handling Region References and Hyperslab Selections** by M. Scot Breitenfeld and Elena Pourmal