1\_Primer

Audience:

A user who handles an HDF5 file with knowledge of the HDF5 data model,

but does not know about the HDF5 library internals.

An HDF5 file consists of information for file metadata and user data. The HDF5 library manages the allocation of file space for storing the file’s metadata and user data. It also manages the free space that results from the manipulation of the file’s objects.

The HDF5 library provides command line utilities for users to examine a file’s contents and the distribution of file space, and to recreate a file with a specified strategy for handling file space.

The first utility is *h5dump*, which the user can use to examine a file’s content. The following *h5dump* output examines the empty file *not\_persist.h5*:

HDF5 "not\_persist.h5" {

GROUP "/" {

}

}

Even though the file is empty, the library automatically creates the root group and allocates space for file metadata to describe the file.

The second utility, *h5stat -S*, allows the user to find out more about the file’s storage information as follows:

Filename: not\_persist.h5

Summary of storage information:

File metadata: 800 bytes

Raw data: 0 bytes

Amount/Percent of tracked free space: 0 bytes/0.0%

Unaccounted space: 0 bytes

Total space: 800 bytes

Note that *not\_persist.h5* has 800 bytes of file metadata in the file and nothing else. There is no user data or free space in the file. The file size of *not\_persist.h5* equals the size of the file’s metadata.

If the user adds four datasets (*dset1*, *dset2*, *dset3 and dset4*) of different sizes to the file *not\_persist.h5*, *h5dump* *–H* produces the following output:

HDF5 "not\_persist.h5" {

GROUP "/" {

DATASET "dset1" {

DATATYPE H5T\_STD\_I32LE

DATASPACE SIMPLE { ( 10 ) / ( 10 ) }

}

DATASET "dset2" {

DATATYPE H5T\_STD\_I32LE

DATASPACE SIMPLE { ( 30000 ) / ( 30000 ) }

}

DATASET "dset3" {

DATATYPE H5T\_STD\_I32LE

DATASPACE SIMPLE { ( 50 ) / ( 50 ) }

}

DATASET "dset4" {

DATATYPE H5T\_STD\_I32LE

DATASPACE SIMPLE { ( 100 ) / ( 100 ) }

}

}

}

See the following *h5stat –S* output for *not\_persist.h5*:

Filename: not\_persist.h5

Summary of storage information:

File metadata: 2216 bytes

Raw data: 120640 bytes

Amount/Percent of tracked free space: 0 bytes/0.0%

Unaccounted space: 1976 bytes

Total space: 124832 bytes

The above output indicates that there is no tracked free space but there are 1,976 bytes of unaccounted space. This is due to the file-space-handling strategy in use for *not\_persist.h5*. When a user creates an HDF5 file, the HDF5 library’s default file-space-handling strategy is not to persist free space at file closing. This means that the library does not save the information to track the file’s free space at file closing, and the user cannot reuse the free space that exists.

If the user reopens *not\_persist.h5*, adds one new dataset (*dset5*), and then deletes one existing dataset (*dset2*) from the file, *h5dump –H* outputs the following:

HDF5 "./not\_persist.h5" {

GROUP "/" {

DATASET "dset1" {

DATATYPE H5T\_STD\_I32LE

DATASPACE SIMPLE { ( 10 ) / ( 10 ) }

}

DATASET "dset3" {

DATATYPE H5T\_STD\_I32LE

DATASPACE SIMPLE { ( 50 ) / ( 50 ) }

}

DATASET "dset4" {

DATATYPE H5T\_STD\_I32LE

DATASPACE SIMPLE { ( 100 ) / ( 100 ) }

}

DATASET "dset5" {

DATATYPE H5T\_STD\_I32LE

DATASPACE SIMPLE { ( 1000 ) / ( 1000 ) }

}

}

}

Look at the following *h5stat –S* output for *not\_persist.h5*:

Filename: ./not\_persist.h5

Summary of storage information:

File metadata: 2216 bytes

Raw data: 4640 bytes

Amount/Percent of tracked free space: 0 bytes/0.0%

Unaccounted space: 124024 bytes

Total space: 130880 bytes

Note that the amount of unaccounted space consists not only of space incurred due to the deletion and addition of the datasets. It also includes the unaccounted space of 1,976 bytes that was there before the user reopened the file. Thus, *not\_persist.h5* now contains fragments of lost space resulting from the user’s manipulation of the HDF5 objects in the file.

Look at a different scenario in creating another HDF5 file *persist.h5*. If the user creates this file with a different file-space-handling strategy that persists free space on file closing and then creates the four datasets (*dset1, dset2, dset3 and dset4*), as is done for *not\_persist.h5*, *h5stat –S* outputs the following:

Filename: ./persist.h5

Summary of storage information:

File metadata: 2391 bytes

Raw data: 120640 bytes

Amount/Percent of tracked free space: 1854 bytes/1.5%

Unaccounted space: 0 bytes

Total space: 124885 bytes

Note that *persist.h5* does not have any unaccounted space but there are 1,854 bytes of free space tracked by the file’s free-space manager. The amount of file metadata is a bit bigger than that of *not\_persist.h5*. This is due to the extra metadata needed by the library for persisting the free-space information. The user can find out more about the distribution of free space in *persist.h5* via *h5stat -s*:

Filename: persist.h5

Small size free-space sections (< 10 bytes):

Total # of small size sections: 0

Free-space section bins:

# of sections of size 10 - 99: 1

# of sections of size 1000 - 9999: 1

Total # of sections: 2

There are two free-space sections in *persist.h5*: one smaller section and one bigger section.

If the user reopens *persist.h5*, adds *dset5* and deletes *dset2,* as is done for *not\_persist.h5*, *h5stat –S* outputs the following:

Filename: ./persist.h5

Summary of storage information:

File metadata: 2427 bytes

Raw data: 4640 bytes

Amount/Percent of tracked free space: 121854 bytes/94.5%

Unaccounted space: 0 bytes

Total space: 128921 bytes

The amount of tracked free space after the addition of *dset5* and deletion of *dset2* reflects the 1,854 bytes of tracked free space that was previously in the file and the free space adjustments resulting from the dataset operations. In particular, the HDF5 library allocates the file metadata needed to add *dset5* from the file’s existing pool of tracked free space. When *dset2* is deleted, the bytes that were used for the dataset’s raw data and file metadata are added to the file’s tracked free space by the HDF5 library.

*h5stat –s* shows the new distribution of free space:

Filename: ./persist.h5

Small size free-space sections (< 10 bytes):

Total # of small size sections: 0

Free-space section bins:

# of sections of size 10 - 99: 1

# of sections of size 100 - 999: 1

# of sections of size 1000 - 9999: 1

# of sections of size 100000 - 999999: 1

Total # of sections: 4

Note that *persist.h5* now has two more free-space sections resulting from the user’s manipulation of the HDF5 objects in the file.

Finally, the third command line utility, *h5repack*, allows the user to re-create a file from an existing HDF5 file with a specified file-space-handling strategy. The *–S* option is used for this. For example, the user can re-pack *not\_persist.h5* with a strategy that allocates file space always from the end of file (strategy #4) to form *not\_persist\_outvfd.h5* as follows*:*

h5repack –S 4 not\_persist.h5 not\_persist\_outvfd.h5

The outputfrom *h5stat –S* shows the following:

Filename: not\_persist\_outvfd.h5

Summary of storage information:

File metadata: 1632 bytes

Raw data: 4640 bytes

Amount/Percent of tracked free space: 0 bytes/0.0%

Unaccounted space: 0 bytes

Total space: 6272 bytes

Comparing the above output with the previous *h5stat –S* output for *not\_persist.h5*, there is no unaccounted space and there is a huge decrease in file size.

The next section, *User Guide*, will document in detail the file-space-handling strategies and the public routines that users can use to create a file with a strategy different from the library default.