RFC: Enhancement to HDF5 concurrent file access

Vailin Choi

This RFC describes the improvement to the HDF5 library so that concurrent access of an HDF5 file by writers and readers will be coherent.

# Introduction

The current HDF5 library does not detect concurrent file access by multiple writers and readers. This may incur problematic scenarios as follows:

* Multiple writers may concurrently open/modify the same file.
* A writer may concurrently open/modify a file that a reader has already opened.

To prevent unexpected concurrent accesses, the library will use the file locking mechanism to ensure file consistency.

# File Locking Mechanism

The HDF5 library will employ two means to implement file locking:

1. Use the system call *flock* to apply or remove an advisory lock on an open file:
   * There are two types of locks that can be applied to a file via *flock*: exclusive (LOCK\_EX) or shared (LOCK\_SH). As described in the *flock(2)* man page, “*only one process may hold an exclusive lock for a given file at a given time*” while “*more than one process may hold a shared lock for a given file at a given time*”.
   * The lock is released automatically when the file closes; the lock can also be released explicitly via *flock* with LOCK\_UN.
2. Use a flag in the file’s superblock to mark the file:
   * The library will mark the file in writing and/or SWMR writing mode based on file open access flags. The marking ensures file consistency for concurrent accesses.
   * The library will clear the flag when the file closes.

A writer or reader process will have different locking actions that further depend on whether the file is open with or without SWMR:

* When a writer process creates/opens a file without SWMR:
  + Place an exclusive lock on the file—the file will remain locked until it closes.
  + Ensure the file is not already marked in writing or SWMR writing mode.
  + Mark the file in writing mode.
* When a writer process creates/opens a file with SWMR write access:
* Place an exclusive lock on the file.
* Ensure the file is not already marked in writing or SWMR writing mode.
* Mark the file in writing and SWMR writing mode*.*
* Release the lock before returning from *H5Fopen/H5Fcreate.*
* When a reader process opens a file without SWMR:
  + Place a shared lock on the file.
  + Ensure the file is not already marked in writing or SWMR writing mode.
* When a reader process opens a file with SWMR read:
  + Place a shared lock on the file.
  + Ensure the file is marked in writing and SWMR writing mode

Once a writer successfully opens/creates a file, no other writer can open/create the file (with/without SWMR). Readers cannot access the file either but with one exception: if a writer has successfully opened the file with SWMR write, then a reader can access the file with SWMR read.

Once a reader successfully opens a file, no writer can open the file (with/without SWMR); but other readers can access the file (with/without SWMR).

The table below depicts the result of a second file open after the first open succeeds—file opens by writer and reader with combinations of access flags:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  | ***first open*** | | | |
| *file open access flags* | *write only* | *write & SWMR write* | *read & SWMR read* | *read only* |
| ***second***  ***open*** | *write only* | fail | fail | fail | fail |
| *write & SWMR write* | fail | fail | fail | fail |
| *read & SWMR read* | fail | succeed | succeed | succeed |
| *read only* | fail | fail | succeed | succeed |

# A Tool

As stated in the previous section, the library uses a flag in the file’s superblock to mark a file in writing and/or SWMR writing mode, and it will clear the flag when the file closes. However, situation may occur that an open file exits without going through normal library closing procedure, and the flag is not cleared as a result. This situation may prevent a user from opening the file. The library therefore provides a simple tool called *h5clear* so that a user can clear this flag.

The tool will open the input file with a file access property list that is set with a private property called H5F\_ACS\_CLEAR\_STATUS\_FLAGS\_NAME. Upon encountering this property during file open, the library will clear this flag in the file’s superblock. The tool will then close the file.

# Future Work

System such as *SunOS Solaris* does not support the system call *flock*. To provide file locking mechanism in the HDF5 library on such system, investigation will be needed to find out how to implement the function of *flock*.

# Acknowledgements

This work is supported by a customer of The HDF Group, Dectris.

# Revision History

|  |  |
| --- | --- |
| *January 10, 2014:* | Version 1 circulated for comment within The HDF Group SWMR team. |
|  |  |
|  |  |