Proposed changes to the sec2 driver

RFC (draft)

Elena Pourmal

March 17, 2006

1. Background

Last summer HDF Helpdesk received a request to add support for the O_DIRECT flag to the HDF5 sec2 driver.

A group wanted to write HDF5 files to the SAN storage device. Preliminary benchmarks showed that O_DIRECT flag with open call helped to improve I/O performance for their applications.

Modified copy of the 1.7.50 snapshot with O_DIRECT support was given to the group. Further benchmarks proved that adding the extra flags to the sec2 driver would be useful to all HDF5 users who would like to take an advantage of specific file systems and system I/O hints.

2. Proposed changes and new sec2 driver public APIs

We propose to add additional parameter to the current H5Pset_fapl_sec2 function to pass extra flags to the sec2 open call and to create a new function H5Pget_fapl_sec2 to retrieve the value of the flags.

The functions will have the following signatures:

herr_t H5Pset_fapl_sec2(*hid_t* fapl, *int* flags);

herr_t H5Pget_fapl_sec2(*hid_t* fapl, *int* *flags);

Parameters:
fapl – an access property list identifier
flags – extra flags (OR'ed) to the sec2 open call

3. Backward/forward compatibility issues

This not a forward compatible API change. Applications that use current H5Pset_fapl_sec2 have to be modified. Fortunately the impact should be very low since sec2 is a default driver and common applications do not use this call.

4. Changes to the HDF5 source

Several changes have to be made in the H5FDsec2.c file:

- Modifications to the H5FD_sec2_t structure
- Addition of two new internal functions H5FD_sec2_fapl_get and H5FD_sec2_fapl_copy
- Modifications to the H5FD_sec2_fapl_open function

More changes are needed (TBD) to

- preserve the current behavior (when H5P_DEFAULT is used for the access property list)
- all calls to H5Pset_fapl_sec2 used by HDF5 tests, tools and C++ and Fortran wrappers
- configure.in to support testing (see section 5)

5. Testing

We should add tests for O_DIRECT for those platforms where the flag exists. This can be done by adding additional tests to configure and defining an appropriate variable (e.g. H5_HAVE_ODIRECT_FLAG)