RFC: Support external link caching feature for HDF5 tools

Jonathan Kim

This RFC is about using the new HDF5 APIs which were added to improve performance for frequent accessing same objects through external links. Those APIs can be used for HDF5 command line tools. New option will be discussed.

# Introduction

New HDF5 APIs were added to improve performance when accessing same objects repeatedly through external links. HDF5 tools’ performance can be improved by utilizing the APIs for users who need to access objects repeatedly through external links with HDF5 tools.

This RFC will present a new option for enabling the feature among necessary HDF5 command-line tools.

The API “H5Pset\_elink\_file\_cache\_size (fapl, Num)” will be used. This routine takes a file access property list ID and an unsigned value to set the number of opened child files to cache.

Refer to the “RFC: Caching Files Opened through External Links” for details of the HDF5 APIs.

# Target command line tools

Currently there are 4 tools that are support accessing object through external link.

Those are ***h5diff***, ***h5dump***, ***h5ls*** and ***h5copy***.

In near future ***h5repack*** will support external link as well.

# Use cases

Assumption:

There are HDF5 files with several objects and there are other HDF5 files with lots of external links to the objects in the other HDF5 files. This means that accessing the objects through external links will cause frequent and repeated access to the same objects when user runs HDF5 tools on the external link side HDF5 files.

Following symbolic links option is turned on with the HDF5 tools listed.

1. User wants to use ***h5diff*** tool on the above HDF5 files containing lots of external links for comparing the files or groups.
2. User wants to use ***h5dump*** tool on the above HDF5 files containing lots of external links for displaying HDF5 files’ contents.
3. User wants to use ***h5ls*** tool on the above HDF5 files containing lots of external links for listing objects or viewing object data with h5ls.
4. User wants to use ***h5copy*** tool on the above HDF5 files containing lots of external links for copying HDF5 files or groups.

# New option for HDF5 tools

This option will trigger for a HDF5 tool to use the new APIs to utilize external link file caching feature.

This option will be valid only when used with ‘follow symbolic/external link’ option.

# Two suggestions for the new option

## --enable-elink-cache[=N]

With this option, the ‘N’ is optional.

For example, a user can specify “—enable-elink-cache” or “—enable-elink-cache=N”.

The ‘N’ is positive integer number bigger than 0, which sets the maximum number of retainable open files.

Pros

* User doesn’t need to be stressed about what number to specify for trying out this feature up front.

Cons

* The default number may not as effective as the effectiveness can be various depend on user’s cases.

Issue

* What would be default N, if not specified?  Decision need to be made from feedback.

## --set-elink-cache=N

With this option, the ‘N’ is not optional. A user must specify N.

The ‘N’ is positive integer number bigger than 0, which sets the maximum number of retainable open files.

Pros

* As this is for performance improvement but not a functional feature, likely advanced user would look for this option. In such assumption, it’s rather clear by making the user specify the ‘N’ number according to user’s external links status.

Cons

* As user needs to enter a number, the user needs to figure out a number. It may keep help desk busy from user’s question.

***>> Tool team would appreciate for your feedback on these options.***

# Acknowledgements

This work was supported by a customer of The HDF Group.

# Revision History

|  |  |
| --- | --- |
| *April 1, 2011:* | Version 1 circulated for comment within The HDF Group. |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |