RFC: Support External Link Open File Cache in HDF5 Tools

Jonathan Kim

This document proposes a command-line tool enhancement using HDF5's external link open file cache feature to improve performance when accessing large numbers of objects through external links.

It further proposes a new option providing user control over the size of the cache to tune performance.

Introduction

External links in HDF5 files provide a transparent mechanism for accessing objects in other HDF5 files, without an application being aware that the object is not in the current file. However, when an object opened through an external link is closed, HDF5 currently closes the external file and releases all cached information about that file. This can cause performance problems if the same external link, or another external link into the same file, is accessed repeatedly.

The HDF5 API `H5Pset_elink_file_cache_size` was added to the HDF5 library to implement the external link open file cache, significantly improving performance when external links are used repeatedly. This feature will be applied to selected HDF5 command-line tools and will be the default behavior.

This RFC also proposes a new argument, `--set-elink-cache`, to allow the user to control the number of cached external link open files.

External link open file cache

Implement the feature in selected HDF5 command-line tools

Four command-line tools currently support accessing objects through external links: `h5diff`, `h5dump`, `h5ls`, and `h5copy`. External link support will be added to `h5repack` in the near future.

External link support will be extended in all of these tools to include support for the external link open file cache. Use of this will be the default behavior because the only change the user will see is improved performance; the output from each tool will remain unchanged. The default can be overridden with the new `--set-elink-cache` option described below.

Use cases

Some HDF5 files contain tens or hundreds of thousands external links. A user working with such files might use `h5diff` to compare two files or `h5dump` to display the contents of a group.

With such large number of external links, the user might experience poor performance with those tools. With the external link open file cache feature, users can expect better performance when handling HDF5 files such as those described below:

- An HDF5 file containing thousands of external links pointing to objects in a single target file

- An HDF5 file containing thousands of external links pointing to objects in multiple target files

HDF5 performance tests indicate that users can expect 7- to 9-times faster performance when processing such files with HDF5 command-line tools after the external link open file cache is enabled.

A new argument for selected HDF5 tools: --set-elink-cache=*N*

A new command-line tool option, `--set-elink-cache=`*N*, will set the number of files that can be held open in an external link open file cache. With this option, the number of cached files is set to *N*, which must be an integer equal to or greater than 0.

The maximum value of *N* will be limited by the system (available memory or OS or file system limitations, for example). Such limitations cannot be known outside the context of each computing system. If we assume no such limitations and if N is an unsigned, 4–byte integer, the maximum value of *N* can be up to 2^32 - 1 (about 4 billion). However this is a theoretical number and not realistic.

For HDF5 command-line tools, the default value of *N* will be set to 16 and its maximum value to 4096 as an initial trial. This is expected to provide enough range for users to work efficiently. These values may be adjusted based on user feedback.

To turn off the cache feature, set *N*=0. The HDF5 command-line tools will then perform as though the external link open file cache had not been implemented.

Note that this argument will affect performance only when used with the `--follow-symlinks` option.

The `--set-elink-cache=`*N* option is provided to allow power-users to tinker with the value of *N* for finer performance tuning.

HDF5 tools will use the external link open file cache feature as the default action with the default value of N set to 16, so a user will not need to take any action to get the performance benefit.

Test

Performance test files and cases:

Performance tests will be added in the same manner as the HDF5 library currently employs them. See http://hdfdap.hdfgroup.uiuc.edu/h5perf/index.html to view example test results for the HDF5 development trunk; click '1.9' and look for the "elink traversal with/without EFC" charts. These results are updated daily.

- Test files
    - An HDF5 file containing ten thousand external links to objects in a single target file
    - An HDF5 file containing ten thousand external links to objects in multiple target files, with the number of target files less than 16 (the default value of *N*)
    - An HDF5 file containing ten thousand external links to objects in multiple target files, with the number of target files greater than 16
- Test cases
    - Compare performance between '*N*=0' and '*N*=16' with each of the above test files. '*N*=0' is used to test performance with no caching support; 'N=16' is used to test default caching performance.

Regression test cases and results:

- Cases for invalid 'N' values.
    - Negative number
    - Number greater than the maximum
    - Non-number (for example, a string)
- Results
    - Output to indicate invalid value of *N*:
      "Error: invalid value of <N>. Try '<tool> -h' or '<tool> --help' for more information"
    - Exit code:
      A number indicates success, failure or error per tool.

1)

2)

Acknowledgements

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

Revision History

April 1, 2011:	Version 1 circulated for comment within The HDF Group.
June 3, 2011:	Version 2 circulated for comment
June 20, 2011:	Version 3 circulated for comment within The HDF Group.