Making H5CX Multi-Thread Safe:
A Sketch Design

John Mainzer
Lifeboat, LLC

12/03/22

Introduction

H5CX was initially developed to support the Asynchronous VOL connector development and performance issues in 1.10.0-1 releases.

To oversimplify somewhat, it creates a per thread cache of the property list IDs provided to the call (or their defaults). Values in these property lists are cached when they are referenced to minimize property list lookup overhead. Further, since contexts (i.e., instances of H5CX_t) are global to their threads, H5CX greatly simplifies routing of property list IDs to the sections of code where they are used.

H5CX also contains code for the manipulations of contexts – presumably to support asynchronous VOL, and possibly others.

To date, H5CX has been put to one side, as for purposes of making the initial target packages multi-thread safe, the per thread context makes it unlikely that H5CX will introduce any issues. As multi-thread support in the VOL layer (H5VL) moves to the top of the to-do list, a careful examination of H5CX will be essential, as H5VL makes a number of H5CX calls with obvious potential multi-thread issues. However, until then, this module has been largely bypassed, and should be viewed as work in progress.