NetCDF-4 Performance Improvements Opening Complex Data Files

Edward Hartnett 12/21/18
High Performance Computing Section
NOAA Earth System Research Laboratory

Abstract

The netCDF-4/HDF5 format offers many advantages for science data users, including built-in compression, a rich data model, and support for parallel I/O. However, several performance bottlenecks have been identified which impact complex files with many variables and attributes defined. These files are especially prevalent in the domain of High Performance Computing (HPC).

The performance issues appear when files include many variables, attributes, or dimensions, and include slow file opens.

A focus on netCDF-4 performance issues has resulted in the elimination of several bottlenecks, including time to open a file, looking up objects in the file, and reading strided arrays. Significant improvement is demonstrated.

Performance trials are conducted on NOAA’s Thea platform. Results are presented demonstrating the improved performance with real-world example files.

NetCDF Background

- NetCDF is a set of software libraries and self-describing, machine-independent data formats that support the creation, access, and sharing of array-oriented scientific data.
- Several binary formats are supported, including (since netCDF-4.0 was released in 2008) HDF5.
- Variables - multi-dimensional arrays of data.
- Attributes - scalar or “small” 1-dimensional array that applies to a file or variable.

Opening a File

- In classic format, all metadata are located in a block at the beginning of the file.
- Classic formats read all metadata, and it’s fast.
- With HDF5 it is scattered throughout the file.
- NetCDF-4 (until version 4.6.2) reads all metadata, and that slows it down.
- HDF5 does not read any metadata on start up, so opens files fast.

Performance Bottleneck in NetCDF-4/HDF5

- Opening a complex netCDF-4/HDF5 file can be slow - two orders of magnitude slower than opening a classic format file.
- Files with lots of global attributes are particularly slow to open.
- Use of gprof profiler identified the dimension mapping as a performance bottleneck.

Performance Testing Code

- nc_test4/tst_attsperf.c
  - The test creates files with varying number of global or variable attributes, or variables.
- nc_test4/tst_wrf_reads.c
  - The test creates files based on three complex real-world netCDF uses (two from WRF, one from a NASA GDMO model).
- Only built and run if --enable-benchmarks is used during configure.
- Timer used to time a netcdf open/close of the file.
- The open/close cycle is run 5 times, and the results averaged.
- A new file is used for each cycle, to defeat buffering.

Performance Testing Platforms

- mikado - dedicated 6-core Intel i7 system, with a solid state drive.
- theia - NOAA 760 Tflop Cray Compute Cluster high performance computing system.
- Results were quite similar.

NetCDF Versions Tested

- 4.6.0 released January 25, 2018.
- 4.6.1 released March 19, 2018.
- 4.6.2 released November 19, 2018.
- lazy read of some variable metadata - https://github.com/Unidata/netcdf-c/pull/1260.
- use of hidden coordinate attribute for dimension mapping - https://github.com/Unidata/netcdf-c/pull/1262

Performance Improvements

- Four performance improvements have been developed:
  - Lazy attribute reads.
  - Fast global attribute reads.
  - Lazy read of some variable metadata.
  - Use of hidden coordinates attribute for dimension mapping.
- Attribute improvements have been merged and were released as part of netCDF version 4.6.2.
- Variable improvements are currently pending review and merge to netCDF master branch.
- If accepted and merged, they will be part of the next netCDF release: 4.6.3(?).

Lazy Attribute Read

- When opening a file, netCDF (until 4.6.2) reads all available metadata, including all attributes.
- Starting with 4.6.2 attributes are not read on file open.
- Attributes are read for a variable or the group when the user:
    - Does an nc_get_att_*() for any attribute of the variable (or group).
    - Does an nc_inq_var() to get the number of attributes for a variable (or nc_inq() for the group attributes).
    - Does an nc_del_att()/nc_rename_att()/nc_copy_att() for any attribute of the variable (or group).
- If no attributes are queried for a variable or group, the attributes for that variable or group are never read.

Fast Global Attribute Reads

- Global atts were being read differently than variable atts.
- Variable atts used H5Aiterate2() to iterate through atts.
- Global atts used H5Aget_num_attrs(), and then iterated through them, using H5Aopen_idx() to open each attribute.
- H5Aiterate2() is much faster!!
- Code was converted to use H5Aiterate2() for global atts as well as variable atts.
- This results in speedup when atts are read. (Due to lazy atts, they are not read until user checks something, like their number.)

Lazy Read of Some Variable Metadata

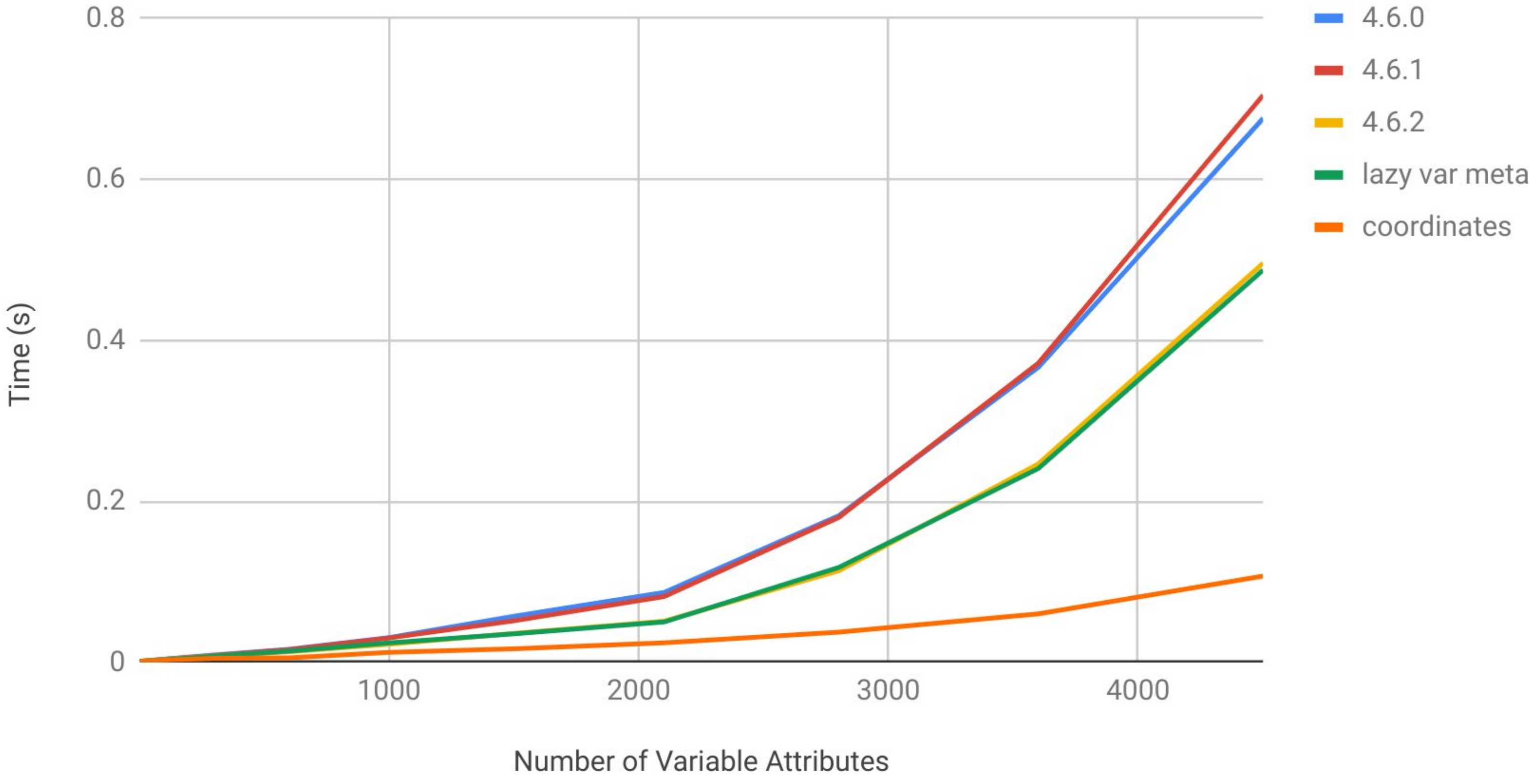
- Instead of reading all variable metadata at file startup, just read enough to map variables and dimensions.
- Read rest of metadata when needed for other variable operations, like getting/putting, renaming, or inqing.
- At file open netCDF must read name, size, some dimensional information, type.
- NetCDF can lazily read cache setting, chunking setting, fill setting and value, filter settings.

Use of Hidden Coordinate Attribute

- NetCDF-4 has always included several hidden attributes. One is **`_NetCDF4Coordinates`**. When present, it cannot be seen through the netCDF APIs.
- This hidden coordinates attribute is used for multi-dimensional coordinate variables, a case that can’t be handled by HDF5 dimension scales.
- Using this hidden attribute for every variable allows for much faster dimension mapping.
- When these hidden attributes are not present, then old-school dimension scale mapping continues to be used.
- This performance improvement will benefit all newly-created files.

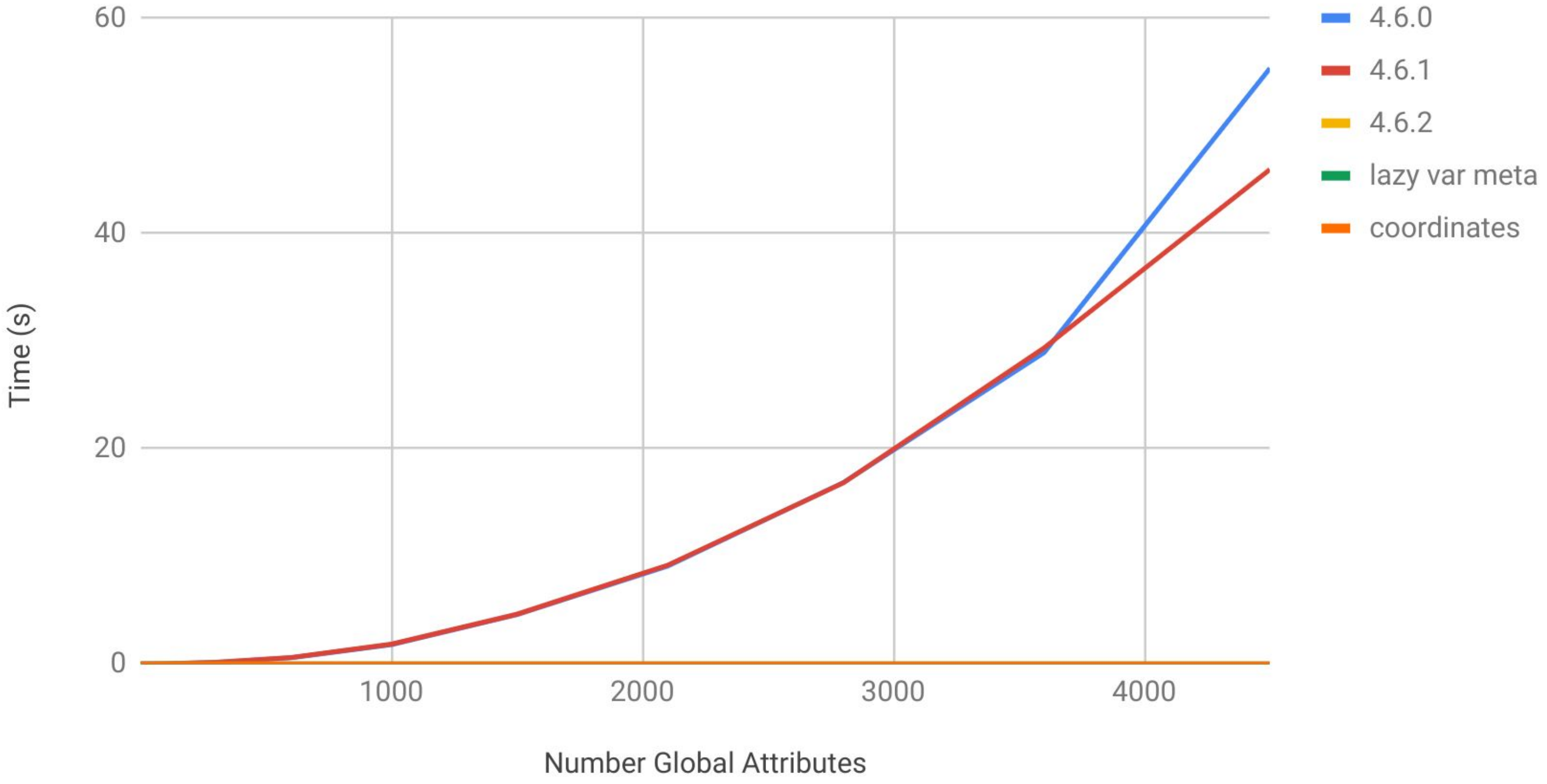
File Open Times with Many Variable Attributes

NetCD-4 File Open Times for Different NetCDF Versions on theia



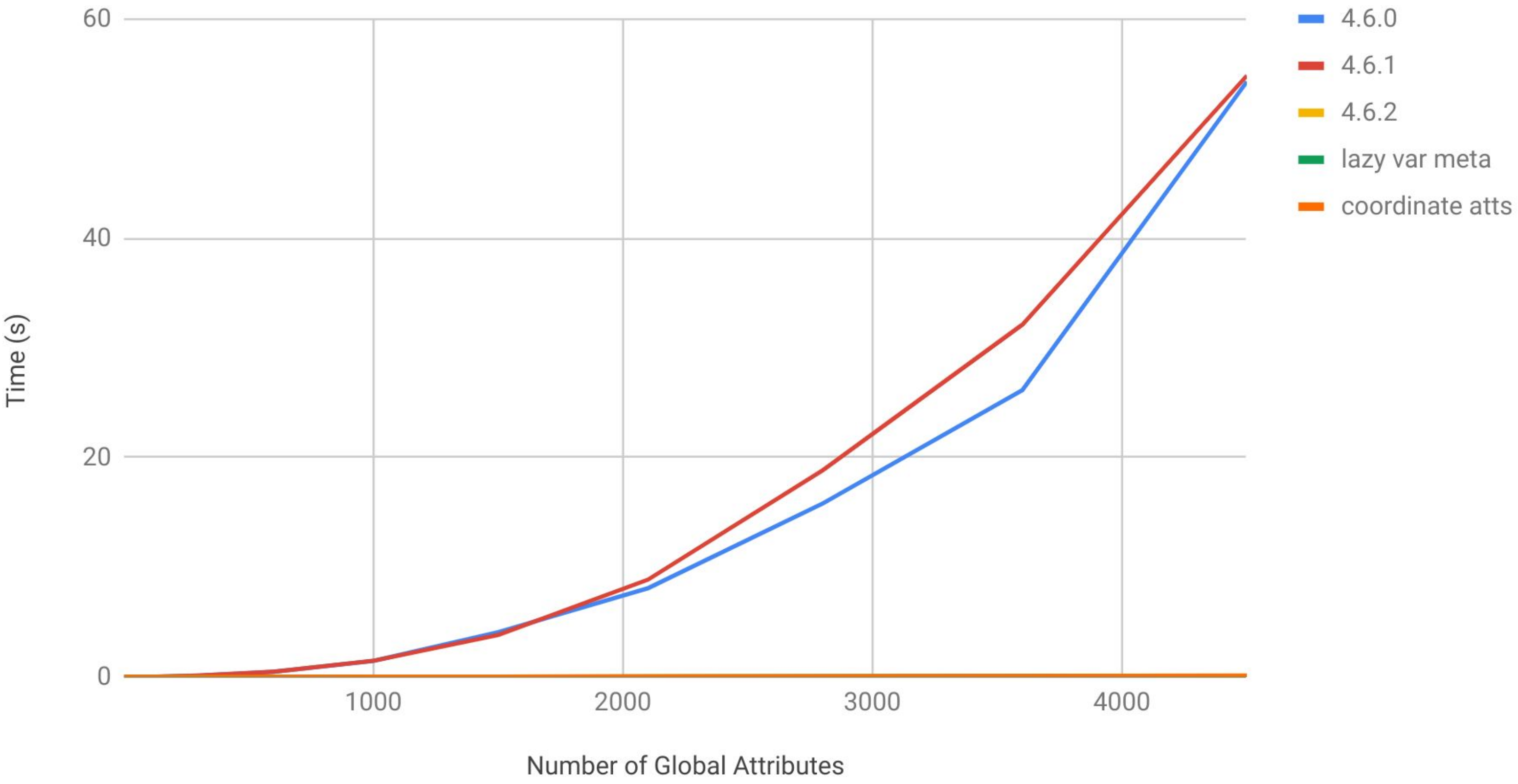
File Open Times with Many Global Attributes

NetCD-4 File Open Times for Different NetCDF Versions on theia



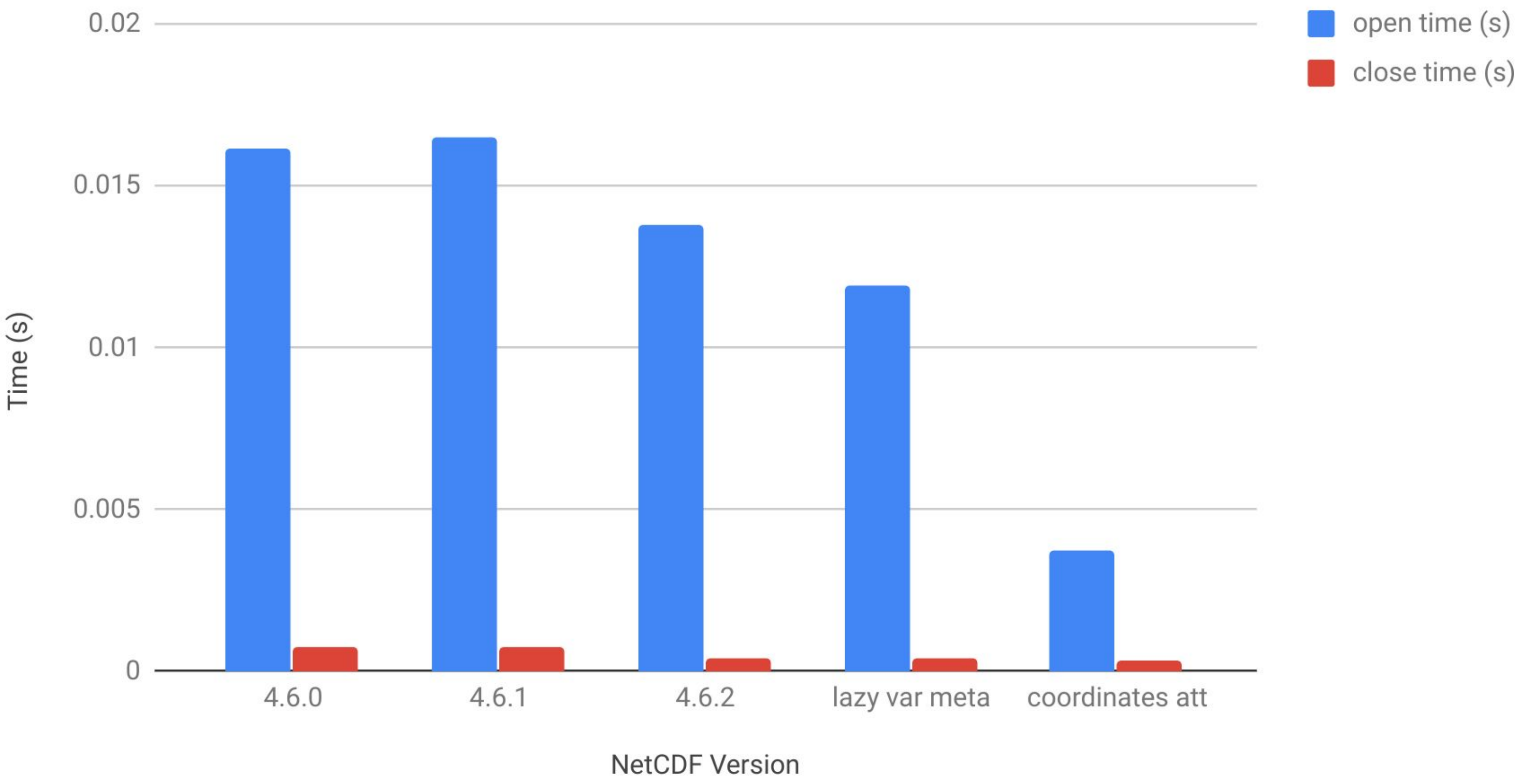
NetCDF-4 File Open Times with Inq of Many Global Attributes

Global Attributes are Subjected to Inq to Trigger Lazy Reads



NetCDF-4 Open/Close Times for WRFBDY File

Time to Open and Close a NetCDF-4 wrfbdy File from WRF Model



Summary

- Some improvements have been made to netCDF-4 file open performance.
- Complex files will open in about ~¼ the current time when all performance improvements are in place. (For newly created files.)
- Some improvements were part of netCDF 4.6.2 release, some coming in the next release.
- See extended abstract for more detail: https://ams.confex.com/ams/2019Annual/meetingapp.cgi/Paper/350021