The Journal of Open Source Software

h5fortran: object-oriented polymorphic Fortran interface for HDF5 file IO

Michael Hirsch[1]

1 Boston University

DOI: 10.21105/joss.02842

Software

- Review
- Repository
- Archive

Editor: Pierre de Buyl

Reviewers:

- @milancurcic
- @bast

Submitted: 06 November 2020
Published: 25 June 2021

Summary

h5fortran (Hirsch, 2018) is a Fortran interface to HDF5 that abstracts away most details of a frequently-used subset of HDF5 file read/write operations. h5fortran has object-oriented and functional interfaces that makes HDF5 use from Fortran as easy as in high-level languages. For example, using the official HDF5 Fortran interface to write an array to disk involves over twenty subroutine calls, plus the associated program logic to call those procedures with efficient parameters. The same array write operation with h5fortran is accomplished with a single subroutine call. A similar reduction in user-facing complexity is achieved by h5fortran for HDF5 array reads. h5fortran adheres to Fortran 2008 standard, working on Gfortran and Intel compilers for Linux, MacOS, Windows on Intel / AMD, ARM and IBM POWER systems. CMake is used to build h5fortran, detecting if the HDF5 library is present and working and building HDF5 from source if necessary. CPack can be used to generate distributable binary archives. Conan package manager may also be used to automatically install the HDF5 library and then install h5fortran. Meson build system is also supported by h5fortran.

h5fortran has general applicability to projects needing to do any of:

- writing variables to HDF5: scalar to 7-D, of type real32, real64 or integer
- reading variables from HDF5: scalar to 7-D, of type real32, real64 or integer
- read / write character variables to / from HDF5 file
- read / write variable attributes to / from HDF5 file
- get the shape of a disk variable to allocate a memory variable for reading that data

HDF5 does not have native support for complex numbers or booleans (`logical` Fortran datatype). h5fortran does not yet support complex numbers or booleans. High-level HDF5 interfaces in other code languages such as h5py have implemented these types using HDF5 struct and HDF5 enum respectively. If the HDF5 community continues to coalesce around these *de facto* data type implementations, we may consider implementing them in h5fortran in the future. h5fortran currently supports the serial HDF5 interface, and does not yet support the parallel MPI-based HDF5 interface. h5fortran does not yet support extensible datasets, although we would be open to investigating adding this support upon community request.

In addition to the object-oriented interface, h5fortran provides single-command read / write procedures. Array slicing allows reading or writing a portion of a large disk variable to/from RAM. If the user has HDF5 with SZIP or ZLIB compression enabled, h5fortran is capable of reading and writing compressed variables, which can save over 50% disk space depending on the data lossless compressibility. Data shuffling and Fletcher32 checksums provide better compression and a check of file integrity respectively. h5fortran was designed for use by individual users on their laptops or embedded devices, as well as for use in HPC applications where parallel tasks need read only part of a milestone or shared HDF5 variable.

h5fortran was originally developed for the GEMINI (M. Zettergren et al., 2017; M. D. Zettergren & Snively, 2019) ionospheric model, funded in part by NASA ROSES #80NSSC20K0176 and DARPA Cooperative Agreement HR00112120003. This work is approved for public release; distribution is unlimited. The information does not necessarily reflect the position or the policy of the Government.

Statement of need

Fortran has only raw file input-output (IO) built in to the language. To support reproducibility of work done in any programming language and long-term usefulness of the data generated or processed, it is beneficial to use self-describing data file formats like HDF5 (Folk et al., 2011). Many popular languages and libraries used for simulation and data science use the HDF5 (The HDF Group, 1997) file IO library. Most programs and libraries intended for use by practitioners such as modelers and data scientists themselves use an object-oriented HDF5 interface like h5py (Collette, 2012, 2013).

Other programs

While other HDF5 interfaces exist, h5fortran presents a broad set of commonly used features, comprehensive test coverage and robustness across compilers and computing systems. We have written a companion library for NetCDF4 called nc4fortran (Hirsch, 2019), which by design has a nearly identical user-facing API. Other Fortran HDF5 interfaces such as HDF5_Utils (Irwin, 2017) use a functional interface mimicking the HDF5 LT functions, which require the end user to keep track of extra variables versus the single object used by h5fortran. A package for C++ with similar priorities of using modern language features and simple commands is h5pp (Aceituno, 2019). A template-based C++ implementation is provided in h5xx (Höfling, 2020).

References

Aceituno, D. (2019). *h5pp: A c++17 wrapper for HDF5*. https://github.com/DavidAce/h5pp

Collette, A. (2012). *h5py: HDF5 python interface*. https://github.com/h5py/h5py

Collette, A. (2013). *Python and HDF5: Unlocking scientific data*. O'Reilly Media, Inc. ISBN: 9781449367831

Folk, M., Heber, G., Koziol, Q., Pourmal, E., & Robinson, D. (2011). An overview of the HDF5 technology suite and its applications. *Proceedings of the EDBT/ICDT 2011 Workshop on Array Databases*, 36–47. https://doi.org/10.1145/1966895.1966900

Hirsch, M. (2018, December). *h5fortran: Object-oriented fortran HDF5 interface*. https://doi.org/10.5281/zenodo.3733569

Hirsch, M. (2019). *nc4fortran: Object-oriented fortran NetCDF4 interface*. https://doi.org/10.5281/zenodo.3718418

Höfling, F. (2020). *h5xx — a template-based c++ wrapper for the HDF5 library*. https://github.com/fhoefling/h5xx

Irwin, J. (2017). *HDF5 utils*. https://github.com/tiasus/HDF5_utils

The HDF Group. (1997). *Hierarchical Data Format, version 5*. http://www.hdfgroup.org/HDF5/

Zettergren, M. D., & Snively, J. B. (2019). Latitude and longitude dependence of ionospheric TEC and magnetic perturbations from infrasonic-acoustic waves generated by strong seismic events. *Geophysical Research Letters*, *46*(3), 1132–1140. https://doi.org/10.1029/2018GL081569

Zettergren, M., Grubbs, G., & Hirsch, M. (2017). *GEMINI3D: Time-dependent ionospheric model*. https://doi.org/10.5281/zenodo.3731716