**P2.41**

# ATMOSPHERIC SCIENCES AND CLIMATE APPLICATIONS USING HDF AND HDF5

MuQun Yang
Robert E. McGrath
Mike Folk
National Center for Supercomputing Applications
University of Illinois, Urbana-Champaign

## 1. Introduction to HDF products: HDF4 and HDF5

The Hierarchical Data Format (HDF), developed at the National Center for Supercomputing Application (NCSA) at the University of Illinois at Urbana-Champaign, has been used by many atmospheric science applications since it was first released in 1988. HDF4 is based on the original 1988 version of HDF and is backwardly compatible with all earlier versions.

Since 1998, NCSA has developed a more general and robust data format, called HDF5, which was designed to support the future demands of Earth Science, such as large data storage, performance, and flexibility. The NCSA HDF group encourages applications to use HDF5 for better performance and maintenance. See extend abstract P2.42 for more information about HDF5. In summary, HDF5 has the following features that make its use attractive for many important atmospheric science applications.

1) Flexible data model
2) Open-source, free software
3) Portability
4) Emphasis on performance
    - MPI- IO support
    - Improved sub-setting
    - In-memory compression
    - Alternative storage models

In this paper, we will demonstrate how HDF4 and HDF5 provide flexibility and efficiency in six well-known earth science applications.

## 2. NASA EOSDIS project

The NASA Earth Observing System Data and Information System (EOSDIS) is a large archive of systematic measurements of the Earth's climate collected by a series of satellites [1]. The EOSDIS archives store and distribute data using HDF4 and HDF5. Data from these missions will be available for many decades, to facilitate the study of climate and climate change.

HDF4 and HDF5 provide rich data models that support most of the types of data required by scientists. To support the specific needs of remote sensed data, the EOSDIS project defined a storage profile, called HDF-EOS, and related metadata standards [2]. HDF-EOS defines a standard way to store geo-referenced data using HDF4 and HDF5.

The EOSDIS archives contain several petabytes of data stored in HDF files. From 1999, data from Terra, Aqua, Landsat 7, and other satellites has been distributed in HDF-EOS2 and HDF4. Starting in 2004, data from the Aura satellite will be delivered in HDF-EOS5 and HDF5 [3].

## 3. NPOESS project

The National Polar-orbiting Operational Environmental Satellite System (NPOESS) is a system of polar orbiting weather satellites and ground equipment for the collection and analysis of weather data and its distribution to government and civilian users. NPOESS will also provide long term climate records [4].

The NPOESS Interface Data Segment will deliver data in HDF5. HDF5 was chosen for NPOESS because it has the capability to operate well in high performance, data intensive environments and can store data in a variety of ways. NPOESS has chosen to standardize its organization of the HDF5 files so data can be easily and consistently accessed and shared amongst the community [4].

The NPOESS Preparatory Program (NPP) will be used as a bridge between the existing Earth Observing System (EOS) program and the NPOESS Program to assure a continuous record of the Earth's climate [5]. The NPP will provide an opportunity to utilize new instruments, algorithms, and data delivery packages prior to the full NPOESS deployment. Data from the NPP satellite will be delivered through the NPOESS Interface Data Segment using HDF5.

## 4. WRF-HDF5 IO modules

WRF (the Weather and Research Forecasting Model) [6] has been mainly developed and maintained at NCAR. It is a regional weather model that has come to be intensively used for both weather research and prediction. Due to the heavy computational volume, WRF is using a multi-layer domain decomposition method to run the model in parallel supercomputing environments.

4.1 Sequential IO issues to address

WRF officially supports the sequential NetCDF IO module, which has two limitations.

First, the IO module may become an IO bottleneck and may exceed memory capacity in some applications.

Secondly, as model resolution becomes higher, the huge volume of disk storage used by large WRF applications may require the use of in-memory data compression techniques to speed data IO.

4.2: How HDF5 can help

1) HDF5 fully supports MPI-IO, the standard parallel IO library, on many platforms. With parallel IO, large WRF applications can avoid or reduce the IO bottleneck and the potential problem of exceeding the memory capacity.
2) HDF5 supports two in-memory compression packages: deflate compression (gzip) and SZIP compression[7]. HDF5 also provides a shuffle algorithm that can amplify the effect of either compression method.

4.3: WRF-HDF5 IO modules

NCSA has implemented both sequential and parallel WRF-HDF5 IO modules.

Both modules should work with the current WRF 2.0 release. Applications can employ either SZIP or gzip compression within the sequential IO module. Applications can do parallel IO through parallel HDF5 IO module. Parallel HDF5 IO is incorporated into WRF model package in WRF 2.0 release [6]. You can also download the source code from HDF website [8].

4.4: Performance results

Figure 1 compares the wall clock time of the WRF-parallel HDF5 IO module and WRF-sequential netCDF IO module for a real WRF application. The maximum hyperslab size per timestep is 17 MB. The WRF-parallel HDF5 IO module outperforms WRF-netCDF IO module as the output file size increases. When the output file size reaches around 20GB, the model run takes nearly 80 minutes with the WRF-sequential netCDF IO module compared to 20 minutes with the WRF-Parallel HDF5 IO module.

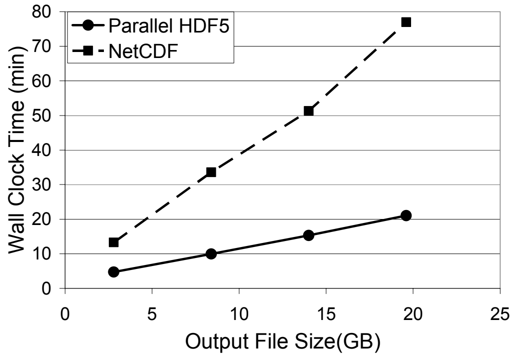

**Figure 1: Comparison of performance of parallel HDF5 to sequential NetCDF for various output file sizes on IBM Power 3 (256 Processors)**

Figure 2 illustrates the effect of compression on output file size in a different application, comparing two compression options with no compression. As expected, the overall HDF5 file size using either SZIP compression or shuffle with gzip compression is much less than the same data without compression. For example, for the data at timestep 70, the file size is about 2000 MB without compression, compared to less than 600 MB for either SZIP compression or gzip compression with shuffle. The total computing time with szip is almost the same as that with no compression[8].

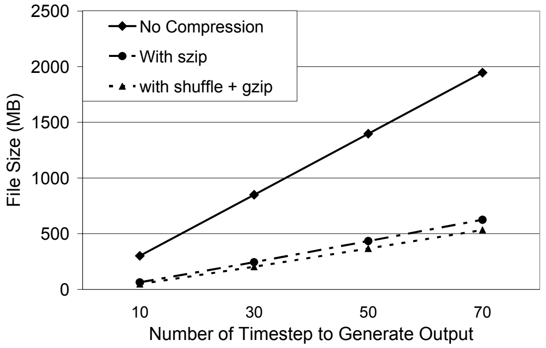

**Figure 2: Performance of sequential HDF5 with different compression methods on IBM Power 4 (16 processors)**

Compared with the sequential NetCDF IO module:

- The WRF-parallel HDF5 IO module can greatly improve WRF IO performance for some WRF applications.
- The sequential HDF5-WRF IO module, invoking available HDF5 compression methods, can greatly reduce the WRF model output data size.

## 5. OPENDAP/DODS HDF4 and HDF5 servers

The Distributed Oceanographic Data System (DODS) [9] is a distributed data management system that was designed so that scientists could use their applications to access their data remotely, over the internet, in a consistent way [10]. DODS uses a web client-server model. A client sends a data request to the server; the server checks the request and answers with the requested data if the request is legal. DODS is widely used by many atmospheric and climate applications.

DODS maintains HDF4 and HDF5 servers [9] so that Earth scientists can obtain data in either HDF format.

Many applications use DODS to do data subsetting, so efficient data access for subsetting across the internet is extremely important. Furthermore, data compression techniques may also be used to reduce the bandwidth required for data transmission across the internet.

An evaluation of the DODS-HDF4 server with AVHRR Oceans Pathfinder data found that using HDF4 chunking storage and compression techniques may greatly reduce data access time [11].

NCSA also implemented a prototype DODS-HDF5 server associated with a DODS-HDF5 white paper [12] and a DODS-HDF5 data mapping paper [13]. NCSA demonstrates how the DODS-HDF5 prototype can work with real Earth science data through the DODS Ferret client [14]. Since there were no real HDF5 datasets in 2000, we used the NCSA HDF4 to HDF5 conversion utility to convert real NASA HDF-EOS data from HDF4 to HDF5 and then used DODS Ferret client to demonstrate server functionality.

## 6. NetCDF4

NCSA and Unidata are collaborating to merge netCDF and HDF5. In version 4.0, the netCDF API will be extended and implemented on top of the HDF5 library and data format [15]. Users of netCDF in numerical models will benefit from support for packed data, larger datasets, data compression, and parallel I/O, all of which are available with HDF5. HDF5 users will benefit from the availability of a simpler high-level interface suitable for array-oriented scientific data; wider use of the HDF5 data format; the wealth of netCDF software for data management, analysis and visualization; and the body of experience that has evolved in the years since netCDF began.

## 7. **HL-HDF5**

National meteorological services in Scandinavian countries have used HDF5 to store their radar data in order to exchange radar data among Sweden, Norway, and Finland with quality-related information. HDF5 was selected because it is open-source, well-designed, and supports in-memory compression. Using HDF5 with deflate compression to store weather data performed better than compression through BUFR [16].

Their weather radar information model can be used to store individual scans, images and products. They also created a high-level interface to HDF5, called HL-HDF5, to facilitate data management [17].

## 8. **Conclusions**

We have briefly introduced six atmospheric and climate applications that use HDF4 and HDF5. An incomplete list of other applications using HDF4 and HDF5 can be found at [18] and more general information of HDF and HDF5 can be found at http://hdf.ncsa.uiuc.edu.

## References

1. Savtchenko A., Ouzounov D., Ahmad S., Acker A., Leptoukh G., Koziana J. and Nickless D.. Terra and Aqua MODIS products available from NASA GES DAAC. *Advances in Space Research, Volume 34, Issue 4, 2004,* 710-714
2. Leptoukh, G.; Ouzounov, D.; Savtchenko, A.; Ahmad, S.; Li Lu; Pollack, N.; Zhong Liu; Johnson, J.; Jianchun Qin; Sunmi Cho; Li, J.Y.; Kempler, S. and Bill Teng. HDF/HDF-EOS data access, visualization and processing tools at the GES DAAC, *Geoscience and Remote Sensing Symposium, 2003. IGARSS '03. Volume: 6* , 3571 - 3573
3. Schoeberl, M.R.; Douglass, A.R.; Hilsenrath, E.; Luce, M.; Barnett, J.; Beer, R.; Waters, J.; Gille, J.; Levelt, P.F. and DeCola, P. The EOS Aura Mission, *International Geoscience and Remote Sensing Symposium (IGARSS)*, *Volume 1, 2001*, 227-232.
4. Goldberg, A.M. Delivering Earth's shapes & colors in near-real time: NPOESS products and their characteristics for users, *Geoscience and Remote Sensing Symposium, 2003. IGARSS '03, Volume:* 5, 3035.
5. Murphy, R.E.; Henegar, J.; Wharton, S.; Guenther, B. and Kealy, P.M.*;* Extending climate data records from the EOS era into the NPOESS era, *Geoscience and Remote Sensing Symposium, 2003. IGARSS '03*, *Volume: 2*, 1332 - 1334.
6. WRF: http://wrf-model.org/
7. SZIP: http://hdf.ncsa.uiuc.edu/doc_resource/SZIP/
8. WRF-HDF5 IO module: http://hdf.ncsa.uiuc.edu/apps/WRF-ROMS

9. DODS: http://www.unidata.ucar.edu/packages/dods/index.html
10. Davis E. and J. Gallagher, Using DODS to Access and Deliver Remote Data, *The 15th International Conference on Interactive Information and Processing Systems(IIPS) for Meteorology, Oceanography, and Hydrology*, 1999.
11. Kapadia A. and N. Yeager. Performance Evaluation of HDF Version 4 Chunking Facility when used for Subsetting Pathfinder Data, 1999 http://hdf.ncsa.uiuc.edu/apps/dods/perfeval/report.html
12. Yang M. and R. E. McGrath. The HDF5-DODS Server Prototype, 2001 http://hdf.ncsa.uiuc.edu/apps/dods/DODS-White-paper.pdf
13. Yang M. and R. E. McGrath. HDF5-DODS Data Model and Mapping, 2001 http://hdf.ncsa.uiuc.edu/apps/dods/HDF5-DODS-Mapping.pdf
14. Yang M. and R. E. McGrath. Demonstration of HDF5-DODS Server Prototype with the DODS-Ferret Client, 2001 http://hdf.ncsa.uiuc.edu/apps/dods/DODS-Ferret-demo.pdf
15. Russ R. and E. Hartnett. “Merging netCDF and HDF5,” Bulletin *of the American Meteorological Society*, *Combined Preprints: 84th American Meteorological Society (AMS) Annual Meeting*, 2004, 1457-1460
16. Michelson D. B., Holleman I., Hohti H., and Salomonsen M., HDF5 information model and implementation specification for weather radar data, 2003, http://www.smhi.se/cost717/doc/WDF_02_200204_1.pdf
17. Henja A, and D. B. Michelson, A High Level Interface to the HDF5 File Format, ftp://ftp.ncsa.uiuc.edu/HDF/HDF5/contrib/hl-hdf5/README.html
18. HDF5 Users, http://hdf.ncsa.uiuc.edu/users5.html

**Acknowledgements:**

The authors want to thank Daniel Michelson for providing information about how HDF5 is used to store weather radar data.

The authors would also like to thank Frank Baker at the NCSA HDF group for his great help in editing the abstract.

The netCDF4 project is funded by NASA award AIST-02-0071.

The WRF IO Module was created as part of the NSF-funded Modeling Environment for Atmospheric Discovery (MEAD) Expedition (http://www.ncsa.uiuc.edu/AboutUs/FocusAreas/MEADExpedition.html)

This report is based upon work supported in part by a Cooperative Agreement with NASA under NASA grant NAG 5-2040. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Aeronautics and Space Administration.