P2.1 Accessing HDF5 data via OPeNDAP

MuQun Yang*[1], Hyo-Kyung Lee[1], James Gallagher[2]
[1]The HDF Group, [2]OPeNDAP, Inc.

I. Introduction

HDF[1] is a set of data formats and software libraries for storing scientific data with an emphasis on standards, storage, and I/O efficiency. HDF software is open-source and available without charge. HDF4 is based on the original 1988 version of HDF and is backward compatible with all earlier versions. HDF4 files cannot be larger than 2GB, and the number of objects in an HDF4 file must be less than 20,000. HDF4's data model is rigid and there is no support for parallel I/O.

HDF5 is a newer data format that was first released in 1998. HDF5 limits neither the size of files nor the size or number of objects in a file. HDF5 is based on a more general data model and offers flexible, high-performance I/O. HDF5 can store almost any kind of scientific data structure, including images, arrays of vectors, structured and unstructured grids, and text. HDF5 was awarded the "R&D 100" award in 2002, as a "technologically significant" product that can change people's lives for the better [2].

HDF5 has been embraced as an important data format for Earth science and computational science, as exemplified by these adopters: HDF-EOS5, which is built on top of HDF5, is the primary data format for data from the Aura satellite [3]. HDF5 is being used as the data format for the real time data products produced from the tri-agency National Polar Orbiting Environmental Satellite System (NPOESS) [4]. Finally, the next generation of netCDF, netCDF-4, is built on top of HDF5 [5].

The Data Access Protocol (DAP) and related client-server software have emerged as important standards for Earth science data system infrastructure. The deployment of different DAP-enabled servers has improved access to data in many formats from many sources. The DAP protocol is widely used to access Earth science data.

*Corresponding author address:
MuQun Yang, The HDF Group,
1901 S. First St., Suite C-2
Champaign, IL 61820
E-mail: ymuqun@hdfgroup.org

Clearly, to provide interoperability and to support data flow between distributed data sources, it is important that DAP servers exist to serve HDF data. A DAP server for HDF4 data has been available for many years [6], and is used extensively to access data from current NASA missions and projects.

With the growing use of HDF5, it is increasingly important that a DAP server for data in the HDF5 format, as well as other formats based on HDF5 (e.g., HDF-EOS5, netCDF-4), should be available to the community. In an earlier collaboration between OPeNDAP and The HDF Group, a prototype OPeNDAP HDF5 server was developed. This prototype effort laid the foundation for the expanded production-level server described in this paper.

We introduce this production-level OPeNDAP HDF5 server, called *HDF5-OPeNDAP server*, and discuss its potential use by researchers, scientists, educators, and the general public to visualize and analyze HDF-EOS5 data via existing DAP clients. Section II gives an overview of the server. Section III describes how to use the server to access general HDF5 data. In section IV, we demonstrate the use of existing OPeNDAP clients to visualize NASA HDF-EOS5 Grid data.

Details on how to use the HDF5-OPeNDAP server to visualize and analyze Aura data are documented in the Appendix A.

II. Overview of HDF5-OPeNDAP Server

1. Features

The prototype server supported the following basic features:

- Mapping of atomic HDF5 data types to simple DAP data types.
- Translation of HDF5 datasets into multidimensional array of DAP data types.
- Translation of HDF5 attributes into DAP attributes.

With the additional features discussed below, the production-level HDF5-OPeNDAP server can access complex HDF5 data and work with existing DAP clients to visualize and analyze HDF-EOS5 Grid data.

a. Mapping an HDF5 Compound Datatype to DAP Structure

The HDF5 compound datatype can support very complex data structures. It is challenging to map the full range of supported complex datatypes to DAP. The addition of this mapping support allows the HDF5-OPeNDAP server to handle complex HDF5 data.

b. Mapping an HDF5 group to DAP attribute

In HDF5, a *group* object is used to show relationships among objects. Defining an appropriate mapping from the HDF5 hierarchical structure, implemented via group objects, to DAP is not trivial. DAP does not have a concept that corresponds to HDF5's group. There are several ways to map an HDF5 group to DAP. We chose to map all HDF5 datasets to DAP variables, with information about the group structure mapped as DAP attributes. This will be natural for most DAP applications, as clients that do not need to use the group information can simply ignore it.

c. Mapping HDF5 references to DAP URLs

Object references and *region references* are HDF5 datatypes that can be used to access HDF5 objects. Primarily, object references are used to access HDF5 groups and datasets. Region references are used to access regions of HDF5 datasets. NPOESS HDF5 data relies heavily on object and region references to organize data aggregations and granules.

URL is a DAP datatype. It is a string containing an OPeNDAP Uniform Resource Locator (URL). A user can specify an OPeNDAP URL to identify a particular data file on a remote host machine. The data can then be accessed over the network via an OPeNDAP server. The URL is not restricted exclusively to OPeNDAP URLs, and therefore can serve as a reference for any DAP object. We use this capability to map HDF5 object references and region references to OPeNDAP.

d. Mapping EOS Grid data to DAP Grid

The ability to map EOS Grid data to DAP Grid data was added to allow users to visualize and analyze Aura EOS Grid files with existing OPeNDAP clients and the HDF5-OPeNDAP server. Although one cannot use this feature to access general HDF5 data, it does help researchers access HDF-EOS5 grid data efficiently and easily.

As mentioned earlier, the NASA Aura mission uses the HDF-EOS5 format to store Earth science data. The HDF5-OPeNDAP server follows the Aura File Format Guidelines [7] to map EOS Grid data to DAP Grid data. Since most OpeNDAP existing clients assume the data served by DAP adheres to CF Conventions [8], the HDF5-OPeNDAP server also adds the attributes required by the CF conventions to DAP on-the-fly by leveraging the information provided by the Aura file guidelines. Except under unusual conditions when the CF conventions cannot be satisfied (e.g., "the length of variable name must be less than 15 characters"), the addition of these attributes ensures that OPeNDAP clients following the CF conventions can successfully access HDF-EOS5 file via the HDF5-OPeNDAP server. The server provides configuration option that may satisfy the CF conventions [9].

We have successfully tested NASA OMI L3G (not yet publicly available) with the existing HDF5 OPeNDAP clients IDV[10], ncBrowse[11], Ferret[12], GrADS[13], NCL[14], and ODC[15]. The detailed demonstrations can be found in Section IV. Note that the feature mapping EOS Grid to DAP Grid following CF conventions is optional. It aims to easily and effectively visualize and analyze HDF-EOS5 Aura data through the HDF5-OPeNDAP server and existing OPeNDAP clients.

2. Other work

A comprehensive test suite has been added using DejaGNU[16]. In addition, the server's source code is checked into OPeNDAP's subversion repository so that the HDF5-OPeNDAP server can be automatically tested every night. These actions help us find and fix bugs quickly.

Several on-line sample Level 2 HDF-EOS5 data files from all four instruments in Aura: OMI, HIRDLS, MLS, and TES [17], as well as several Level 3G HDF-EOS5 sample files obtained from Aura developers were used for testing. An internal test suite was created to test these data files with the HDF5-OPeNDAP server.

With the exception of the Aura EOS data that included a special character in the dataset name, all Aura EOS data can be successfully accessed by the HDF5-OPeNDAP server. An internal test suite has been implemented to periodically test the Aura data accessibility.

3. Limitations

Due to the restrictions of the OPeNDAP data model, the HDF5-OPeNDAP server has the following limitations:

- For HDF5 variable length datatype, the server only supports the mapping of one

dimensional array of HDF5 variable length string to DAP.

- The server won't map HDF5 ENUM, BITFIELD and OPAQUE types to DAP.
- HDF5 64 bit integers (and arrays) are not supported.
- HDF5 signed and unsigned char datatypes are both mapped to Byte in DAP.

III. Accessing general HDF5 data via the HDF5-OPeNDAP Server

The main task of the server is to retrieve all or part of arrays of numbers. In DAP, this is an either an *Array* or a *Grid*. The former is a general multidimensional array of numbers; the latter has named vectors to label the dimensions.

For instance, a *Grid* may have dimensions latitude, longitude, and time. The server must map HDF5 datasets into the corresponding *Array* and *Grid*. This can be done for any HDF5 dataset with HDF5 predefined atomic datatypes. We can also support hyperslab selection for particular subsets of array data. If an HDF5 dataset also has "dimension scales" of the appropriate kind, it will be mapped as a *Grid*.

An OPeNDAP client requests information or data from a dataset on a remote host. The request is sent as an OPeNDAP Uniform Resource Locator (URL). A typical DAP URL is shown below:

http://hdfdap.hdfgroup.uiuc.edu:8088/cgi-bin/nph-dods/aura.h5.*xxx*

In the above URL, *nph-dods* is the name of the HDF5-OPeNDAP server CGI program and *aura.h5* is the file name. The *xxx* suffix indicates the type of DAP information requested. The URL suffix can either be *dds*, *dods,* or *das*, to request one of the three types of DAP information. The server CGI program will run the HDF5 handler program called *dap_h5_handler* with an appropriate option: *-das*, *-dds* or *-dods*.

a. Dataset Descriptor Structure(DDS)

To obtain a description of the dataset, the client sends a URL similar to the following:

http://hdfdap.hdfgroup..uiuc.edu:8088/cgi-bin/nph-dods/aura.h5.dds

The suffix requests that *nph-dods* run *dap_h5_handler* with the *-dds* option to generate the DDS for the corresponding HDF5 file and data structure (Fig.1).

The *dap_h5_handler* program reads the HDF5 file and calls OPeNDAP libraries to construct a DDS table with appropriate DAP data types. The server returns the DDS to the OPeNDAP client.

b. DataDDS

To retrieve data, the DAP client sends a URL such as:

http://hdfdap.hdfgroup.uiuc.edu:8088/cgi-bin/nph-dods/aura.h5.dods

The *dods* suffix requests that *nph-dods* run *dap_h5_handler* with *-dods* to obtain HDF5 data and transport them via Internet in MIME type 'application/octet'. If you replace *dods* with *ascii* you can view the values of dataset in text as shown in Fig. 2.

In DAP, *DataDDS* uses *DDS* to determine how individual HDF5 datasets should map into the corresponding DAP objects.

c. Data Attribute Structure(DAS)

The *dap_h5_handler* program with *-das* option returns a list of all the attributes of the objects in an HDF5 dataset in DAS(Fig. 3).

This includes all the HDF5 group and dataset attributes and also the information such as:

- Attributes of the HDF5 root group are returned as global attributes in DAS
- HDF5 group structure stored as an attribute called HDF5_ROOT_GROUP

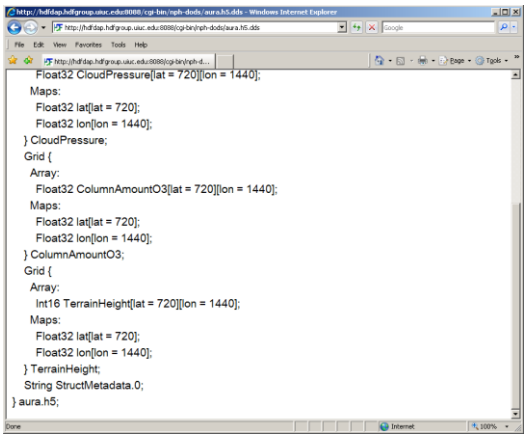

Fig. 1. DDS output of Aura HDF5 file through HDF5-OpeNDAP server

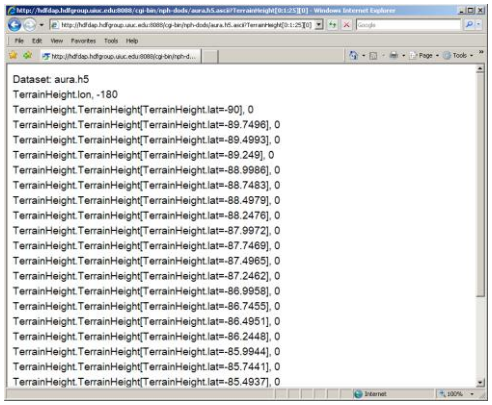

Fig. 2. DataDDS output of Aura HDF5 file through HDF5-OPeNDAP server in ASCII mode

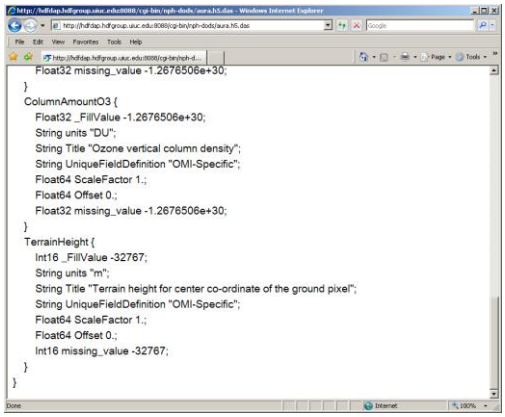

Fig. 3. DAS output of Aura HDF5 file through HDF5-OPeNDAP server

IV. Accessing NASA Aura HDF-EOS5 Grid data via HDF5-OPeNDAP handler and OPeNDAP clients

A breakthrough in the usage of the HDF5-OPeNDAP server is that we can successfully use several existing OPeNDAP clients to demonstrate NASA Aura HDF-EOS5 OMI L3G data served via the HDF5-OPeNDAP server.

The OPeNDAP clients we used were Ferret, ODC, IDV, ncBrowse, NCL, GrADS, and Matlab. Matlab, however, will require some modifications to read certain HDF5 data sources. Fig 4. through Fig 9. are snapshots of the clients that we tested.

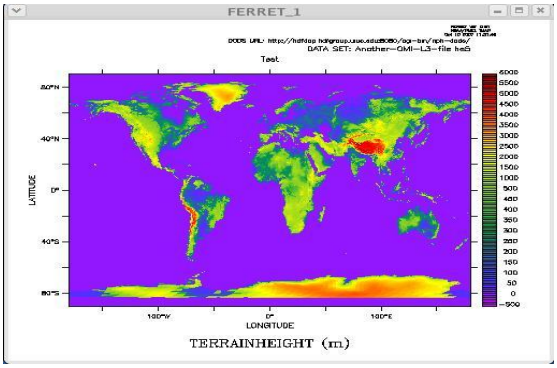

Fig. 4. Displaying NASA EOS Aura Grid Map of terrain height through the Ferret OPeNDAP client. The unit is in meters.

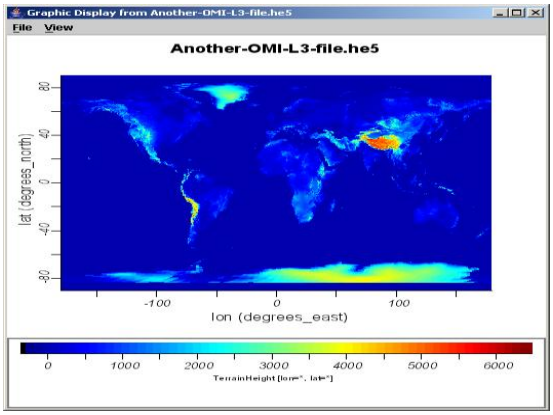

Fig. 5. Displaying NASA EOS Aura Grid Map of terrain height through the ncBrowse OPeNDAP client. The unit is in meters.

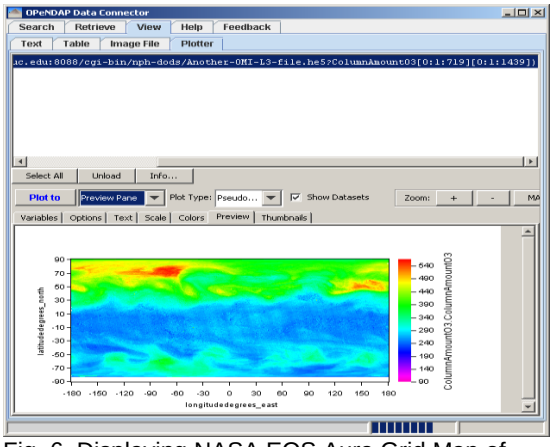

Fig. 6. Displaying NASA EOS Aura Grid Map of ozone through the ODC OPeNDAP client. The unit is DU.

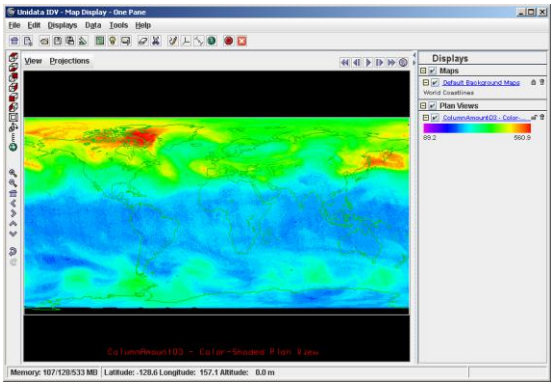

Fig. 7. Displaying NASA EOS Aura Grid Map of ozone through the IDV OPeNDAP client. The unit is DU.

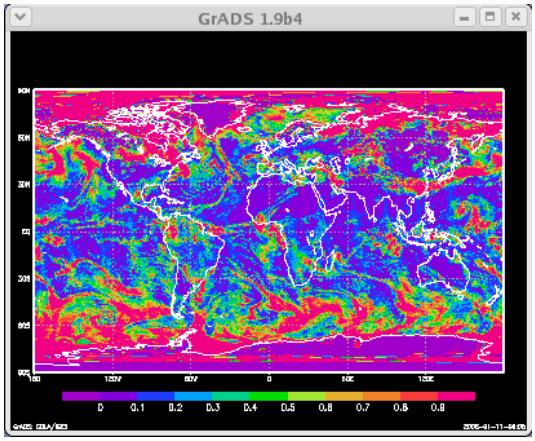

Fig. 8. Displaying NASA EOS Aura Grid Map of cloud fraction through the GrADS OPeNDAP client.

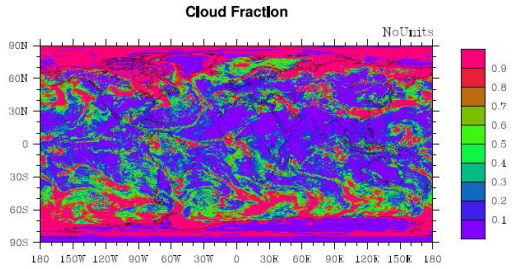

Fig. 9. Displaying NASA EOS Aura Grid Map of cloud fraction through the NCL OPeNDAP client.

V. Conclusions

In this paper we demonstrate how to use the HDF5-OPeNDAP server to access HDF5 data, especially HDF-EOS5 data. We believe that the server can help researchers, scientists, educators, and the general public visualize and analyze HDF-EOS5 data via existing DAP clients.

The official HDF5-OPeNDAP server will be released soon. Visit http://hdfdap.hdfgroup.uiuc.edu:8080/install.html to learn how to build the HDF5-OPeNDAP server now.

References

1. HDF home page: http://hdfgroup.org/

2. The HDF Group: “HDF5 Nomination for the R&D 100 Award 2002”, http://hdfgroup.org/HDF5/RD100-2002/All_About_HDF5.pdf

3. Schoeberl, M.R.; Douglass, A.R.; Hilsenrath, E.; Luce, M.; Barnett, J.; Beer, R.; Waters, J.; Gille, J.; Levelt, P.F.; DeCola, P. “The EOS Aura Mission,” International Geoscience and Remote Sensing Symposium (IGARSS), Volume 1, 2001, pp. 227-232.

4. Murphy, R.E.; Henegar, J.; Wharton, S.; Guenther, B.; 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, pp. 1332 - 1334.

5. Rew, Russ and Hartnett, Ed. “Merging netCDF and HDF5,” Bulletin of the American Meteorological Society, Combined Preprints: 84th American Meteorological Society (AMS) Annual Meeting, 2004, pp. 1457-1460.

6. HDF4 server, http://www.OPeNDAP.org/download/hdf4_server.html

7. A File Format for Satellite Atmospheric Chemistry Data Based On Aura File Format Guidelines: http://www.eos.ucar.edu/hirdls/HDFEOS_Aura_File_Format_Guidelines.pdf

8. CF conventions: http://www.cfconventions.org/

9. Visualizing EOS5 Grid data via HDF5-OPeNDAP handler and OPeNDAP clients following CF conventions: http://hdfdap.hdfgroup.uiuc.edu:8080/cf.html

10. IDV – Integrated Data Viewer: http://www.unidata.ucar.edu/software/IDV/index.html

11. ncBrowse – A Graphical netCDF File Browser: http://www.epic.noaa.gov/java/ncBrowse/

12. Ferret /OPeNDAP:
http://ferret.wrc.noaa.gov/Ferret/DODS/ferret_dods.html

13. GrADS – Grid Analysis and Display System:
http://grads.iges.org/grads/grads.html

14. NCL - NCAR Command Language:
http://www.ncl.ucar.edu/

15. ODC – OPeNDAP Data Connector:
http://www.opendap.org/ODC/

16. DejaGnu:
http://www.gnu.org/software/dejagnu/

17: NASA Aura Instruments:
http://aura.gsfc.nasa.gov/instruments/index.html

Acknowledgements

The authors would especially like to thank Dr. Mike Folk at The HDF Group who provided insightful suggestions for this work. We would also like to acknowledge Dr. Robert McGrath at the National Center for Supercomputing Applications at University of Illinois, who originally was the PI of this project and who laid the foundation for the current work. We also acknowledge Dr. Peter Leonard, Daniel Kahn, and Marghi Hopkins at ADNET Systems who kindly provided NASA Aura HDF-EOS5 OMI L3G data for us. Without their gracious help, we could not successfully demonstrate the use of existing OPeNDAP clients to access HDF-EOS5 data. We also thank the valuable suggestions provided by Dr. Christopher Lynnes, Dr. James Johnson, and Denis Nadeau. Thanks to Jennifer Adams from GrADS and Dave Brown at UCAR for providing timely help when we tested the server to access HDF-EOS5 data via GrADS and NCL. Finally, we owe great appreciation to Ruth Aydt at The HDF Group for her extraordinary help in editing the abstract. Her detailed comments on various parts in this abstract also make us clarify some contents that will be missed otherwise.

Appendix A: Detailed procedure for displaying HDF-EOS5 grid data using IDV

This appendix provides detailed instructions on how to use IDV to visualize HDF-EOS5 data via the HDF5-OPeNDAP server.

1. Set up the web server, OPeNDAP server, and HDF5-OPeNDAP server.

a. Set up an Apache Web Server:
http://httpd.apache.org/

b. Set up an OPeNDAP Server:
http://www.opendap.org/download/CGI_server.html

c. Obtain the latest HDF5-OPenDAP server source via subversion or from
http://www.opendap.org/download/data_handlers.html

Following the readme included with the server to configure and build the HDF5-OPeNDAP server.

The configure option will be similar to:

./configure --enable-cf --enable-short-path

d. Obtain test data and put it under Apache web server's document root:

For example,

http://your-server.org/aura.h5

e. Check the server and the handler.

http://your-server.org/cgi-bin/nph-dods/aura.h5.dds

2. Set up IDV

a. Obtain and install IDV from
http://www.unidata.ucar.edu/software/idv/

b. Go to the "Dashboard" window and click on the "Data Chooser" tab.

c. Select "URLs" from the tabs on the left side and enter the URL of your server. Select "Grid files(netCDF/GRIB/OPeNDAP)" for "Data Source Type". Next, Press the "Add Source" button.

d. Expand "2/3D grid" and click on one of the fields in the "Fields" column.

e. Select "Color-Shaded Plan View" from the "Displays" column.

f. Press the "Create Display" button.