

Transition from HDF4 to HDF5: status and goals

Robert E. McGrath

NCSA

December 5, 2002

HDF4 vs HDF5

- HDF4 - Based on original 1988 version of HDF
 - Backwardly compatible with all earlier versions
 - Original HDF-EOS (terra, aqua)
- HDF5
 - New format & library - ***not compatible with HDF4***
 - HDF-EOS5 (aura)

Important Note

- **Both HDF4 and HDF5 are supported by the NCSA HDF group.**
- **We will continue to maintain HDF4, as long as we are funded to do so.**
- **We recommend using HDF5, and that you consider migrating from HDF4 to HDF5 to take advantage of the improved features and performance of HDF5.**

See: <http://hdf.ncsa.uiuc.edu/h4-h5.html>

Overview

- Discuss status of transition
- Suggestions for users and developers

Thesis:

Most environments will be using both HDF4 and HDF5 data and software for many years.

NASA ESE Data Centers and Users will be using both HDF4 and HDF5

- NASA ESE holdings are HDF4-based data and software
 - Terra, Aqua, Landsat 7, etc.
- Near future will include HDF5-based data and software
 - Aura, possibly others

Supporting Transition: Status and Discussion

Four Important Goals for NCSA (and NASA)

1. Support both formats and libraries
2. Interoperation of data and libraries
3. Conversion of data
4. Conversion of software

1. Support both formats and libraries

- NCSA is committed to support both HDF4 and HDF5 as long as needed by NASA
- But expertise with HDF4 will erode
- We need to get HDF4 software into a stable “safe mode”.
 - Analogous to satellite systems: a stable, dormant, well-known state, from which it can be awakened when needed.
- I assume these statements are true for HDF-EOS

2. Interoperation of data and libraries

- Must always be able to use HDF4 and HDF5 data and software in the same programs and environments
- This has largely been achieved at the data and library level (by virtue of separate name spaces)
- Many applications still need to work through this issue

2. Interoperation of data and libraries

- Likely to be increasingly difficult to ensure that HDF4 and HDF5 will work together on all platforms and compilers.
 - E.g., HDF4 has F77, HDF5 has F90: difficult for a single Fortran program to use both.

3. Conversion of Data

- One approach to heterogeneity is to convert data, especially from HDF4 to HDF5.
- NCSA documents, utilities, and toolkit support default conversions
 - <http://hdf.ncsa.uiuc.edu/h4-h5.html>
- *heconvert* utility for HDF-EOS files

Conversion of Data

- Generic conversion is not likely to be sufficient except in very simple cases
 - Default conversion may not produce desired result, or may be non-optimal use of HDF5
 - Always most important to preserve application and science meaning, not the details of HDF
- Generally will need product-by-product conversion

Conversion of Data

- Data can be converted as needed, or wholesale, e.g., as part of a refresh or regeneration process
- It is not clear what needs to be done to validate converted data
 - Preserve the numbers
 - Preserve structure (e.g., Swath)
 - How to cross-validate the “same dataset” is “same” when converted from HDF4 to HDF5?

4. Conversion of Software

- Adding HDF5 and HDF-EOS5 support to existing software will be a common and important task
 - Basically same as adding any new format
- Not difficult in any give case, unless the data is very complex
- But each case is different

Conversion of Software

- HDF-EOS metadata makes this much easier
 - Metadata is format independent
 - Tools that use the metadata don't have to change at all
 - The metadata makes it much easier to make the new format work the same as the old
- To the degree that these claims are true, *this is a strong validation* of the value of the effort (specification and implementation) that went into the HDF-EOS metadata.

Suggestions for Users

Suggestions for Users

- If you haven't started yet, use HDF5
- In most cases, you need to support with HDF4 and HDF5 (HDF-EOS4 and HDF-EOS5)

Convert Data or Multiple Readers?

- In an environment with both HDF4- and HDF5-based data (HDF-EOS4 and HDF-EOS5), how should programs deal with the data?
 - Convert data
 - Write software to read/write either format

Convert Data or Multiple Readers?

- Converting data (e.g., to HDF5) makes the software simpler
 - Only one reader/writer needed (data is converted to suit the reader)
 - Need conversion software
 - Conversion may be costly and surely is ‘extra work’
 - Multiple copies of the “same” data may exist

Convert Data or Multiple Readers?

- Reader/Writer for HDF4 and for HDF5
 - A lot of software already supports multiple formats
 - More software development
 - ...and maintenance (bugs have to be fixed twice, both libraries need to be upgraded, etc.)
 - Adding new data access methods is possible only if the code can be modified
 - Proprietary code
 - Design prohibits extension
 - No programmers available

Converting data or software from HDF4 to HDF5:

Three “Principles”

Three “Principles”

- 1. Do what makes sense*
- 2. Think of HDF5 as a completely new file format*
- 3. Anything you can do in HDF4, you can do in HDF5*

1. Do what makes sense

- The documentation and tools and talks are suggestions, not rules.
- Use HDF5 in ways that work best for your goals
 - Sometimes, it may not be best to exactly copy or emulate the way HDF4 was used
 - HDF5 has many new features to exploit

2. Think of HDF5 as a new Format

- Despite the name, the HDF5 Format and Library are new and different.
- Shouldn't expect things to work the same as HDF4 or earlier versions.

3. Anything you can do in HDF4, you can do in HDF5

- That said, HDF5 is conceptually compatible with HDF4
- It is reasonable to expect that whatever you did with HDF4 can be done with HDF5, usually better.

Suggested Goals for NCSA

- Get HDF4-based software into “safe mode”
- Help for software developers

“Safe Mode”

- Analogous to satellite systems: a stable, known, dormant state, from which it can be awakened when needed.
 - Document format and library*
 - Clean up and document source code
 - Porting guide for “temporal ports”

* Substantial progress in the last two years

“Temporal Port”

- Most “normal” maintenance brings software forward continuously, version by version
 - E.g., when OS is upgraded, software is made to work
- When software is “dormant”, may be called upon to make it work on something several years and many versions later than the last maintenance
- This is much more like porting to a new platform, because of the temporal gap: hence my term “*temporal port*”

Help for developers

- Given software that uses HDF4 (HDF-EOS4) add support for HDF5 (HDF-EOS5):
 - *Need to perform this task over and over*
- What might help:
 - Documents, training, and consulting for developers
 - Toolkits to assist software conversion?

Pointers

1. The HDF web site:
<http://hdf.ncsa.uiuc.edu>
2. The helpdesk: hdfhelp@ncsa.uiuc.edu
3. **HDF4 to HDF5 information:**
<http://hdf.ncsa.uiuc.edu/h4toh5/>

Acknowledgements

This report is based upon work supported in part by a Cooperative Agreement with NASA under NASA grant NAG 5-2040 and NAG NCCS-599. 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.

Other support provided by NCSA and other sponsors and agencies (<http://hdf.ncsa.uiuc.edu/acknowledge.html>).