HDF5 BoF
State of the Union

November 16, 2016
BoF Session Leaders

DAVID PEARAH, CEO @ HDF Group

JEROME SOUMAGNE, HPC Engineer @ HDF Group

Note: Quincey Koziol @ Lawrence Berkeley National Laboratory + Elena Pourmal @ HDF Group couldn’t attend but send their regards!
1. BACKGROUND: HDF GROUP + HDF5
2. FIRST 6 MONTHS
3. SUPPORT PACKAGES
4. HPC VENDOR PARTNER PROGRAM
5. HDF5 1.10: MARCH 2016
6. HDF5 1.10.1: JANUARY 2017
7. HDF5 ROADMAP: 2017 – 2018
8. COMMUNITY OUTREACH
Who is the HDF Group?

HDF Group has developed open source solutions for Big Data challenges for nearly 30 years.

Small company (40+ employees) with focus on High Performance Computing and Scientific Data.

Offices in Champaign, IL + Boulder, CO.

Our flagship platform – HDF5 – is at the heart of our open source ecosystem.

Tens of thousands use HDF5 every day, as well as build their own solutions (700+ projects in Github).

“De-facto standard for scientific computing” and integrated into every major analytics + visualization tool.
What does the HDF Group do?

**Products**
- **HDF Platform**
- Connectors: ODBC, Cloud
- Add-Ons: compression, encryption

**Support**
- Helpdesk
- Support for h5py + PyTables + pandas (NEW)
- Training

**Consulting**
- HDF: new functionality + performance tuning for specific platforms
- HPC software engineering with scientific domain expertise
- Metadata science and expert services
Our Industries

- Financial Services
- Oil and Gas
- Aerospace
- Automotive
- Medical & Biotech
- Silicon Manufacturing
- Electronics Instrument
- Government
- Defense & National Security
- Academic Research
A few of our users
Why Use HDF5?

I/O library optimized for scale + speed

Users who need both features

Self-documenting container optimized for scientific data + metadata
The HDF5 Platform

Marriage of data model + I/O software + binary container

HDF5 abstract data model

HDF5 library

HDF5 file format
Complex collisions of particles that light up the aurora borealis can fracture Earth’s magnetic shield and wreak havoc on electronics, power grids, and space satellites.

Visualization of trillion-particle datasets made possible with HDF5 are helping scientists decipher how.
EARTH OBSERVING SYSTEM
NASA

Deliver 6,700 Different Data Products to 12 Data Archive Centers

Nearly 16 terabytes per day are redistributed to more than 1.7 million end users worldwide
I/O library usage on leadership systems

**ALCF Procided Library Usage for Mira. Coverage is ~53% of jobs between 08/15 to 08/16**

<table>
<thead>
<tr>
<th>Library</th>
<th>Number of Jobs</th>
</tr>
</thead>
<tbody>
<tr>
<td>job</td>
<td>10000</td>
</tr>
<tr>
<td>icccomp</td>
<td>10000</td>
</tr>
<tr>
<td>darshan</td>
<td>10000</td>
</tr>
<tr>
<td>tacc</td>
<td>10000</td>
</tr>
<tr>
<td>tacc++</td>
<td>10000</td>
</tr>
<tr>
<td>scalapl</td>
<td>10000</td>
</tr>
<tr>
<td>hdfs</td>
<td>10000</td>
</tr>
<tr>
<td>few</td>
<td>10000</td>
</tr>
<tr>
<td>blue</td>
<td>10000</td>
</tr>
<tr>
<td>cash/fy</td>
<td>10000</td>
</tr>
</tbody>
</table>

Credit: Venkat Vishwanath (ANL)

**OLCF I/O Library Usage**

- HDF5 90%
- NetCDF – HDF5 Parallel 7%
- Other libraries

Credit: Scott Atchley (ORNL)
I/O library usage on leadership systems

Library usage on Edition by number of unique users 1/13/2015 to 1/12/2016

Credit: Zhengji Zhao (NERSC)
**First 6 Months: What’s New**

<table>
<thead>
<tr>
<th>WHAT’S ALREADY WORKING</th>
<th>WHAT’S CHANGING</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Users committed to success of HDF, e.g. community-driven growth from 600 to 700 projects on Github in just 6 months</td>
<td>• Launched Support Packages</td>
</tr>
<tr>
<td>• HDF5 core platform equals very high quality software</td>
<td>• Launched HPC Vendor Partner Program (to support vendors and end users and also develop optimized + advanced versions of HDF5)</td>
</tr>
<tr>
<td>• Reputation + track record of HDF Group: speaks for itself</td>
<td>• Increased focus on commercial clients, particularly Fintech</td>
</tr>
<tr>
<td></td>
<td>• Added expertise for Big Data (Spark) + Cloud (AWS) products and services</td>
</tr>
<tr>
<td></td>
<td>• Expanded engineering team to tackle general HPC + Scientific Data projects… not “just” HDF5</td>
</tr>
</tbody>
</table>
Support Packages (NEW)

- [https://www.hdfgroup.org/support/](https://www.hdfgroup.org/support/)
- **Bigger differentiation between free vs paid support**
  - Retain free HDF Help Desk but more limited in scope and increased reliance on HDF community itself
  - Emphasis on comprehensive support packages
  - Paid support directly funds team that maintains and extends the core platform
- **NEW: Adding official HDF Group support**
  - Python, including Pandas + PyTables + h5py
  - R
  - .NET
## Support Packages (NEW)

<table>
<thead>
<tr>
<th>Feature</th>
<th>Community</th>
<th>Basic</th>
<th>Pro</th>
<th>Premier</th>
</tr>
</thead>
<tbody>
<tr>
<td>Online knowledgebase + Community Forum</td>
<td>⭐</td>
<td>⭐</td>
<td>⭐</td>
<td>⭐</td>
</tr>
<tr>
<td>Training Videos</td>
<td></td>
<td>⭐</td>
<td>⭐</td>
<td>⭐</td>
</tr>
<tr>
<td>Flexible Assistance on HDF Group’s Technologies: annual hour for development, testing, support, documentation or training.</td>
<td></td>
<td>10</td>
<td>40</td>
<td>80</td>
</tr>
<tr>
<td>Onsite Customized Training</td>
<td></td>
<td></td>
<td>⭐</td>
<td>⭐</td>
</tr>
<tr>
<td>Email Support: initial response SLA</td>
<td>No SLA</td>
<td>&lt; 2 days</td>
<td>&lt; 1 day</td>
<td>&lt; 4 hours</td>
</tr>
<tr>
<td>Phone Support: initial response SLA</td>
<td></td>
<td></td>
<td>&lt; 1 day</td>
<td>&lt; 4 hours</td>
</tr>
<tr>
<td>Rapid Issue Response: best efforts for a fix or workaround for your confirmed bugs within 5 days</td>
<td></td>
<td></td>
<td>⭐</td>
<td></td>
</tr>
<tr>
<td>C, C++, Fortran, Java</td>
<td></td>
<td>⭐</td>
<td>⭐</td>
<td>⭐</td>
</tr>
<tr>
<td>.NET: C#, Visual Basic</td>
<td></td>
<td>⭐</td>
<td>⭐</td>
<td>⭐</td>
</tr>
<tr>
<td>Python: h5py, PyTables, pandas</td>
<td></td>
<td></td>
<td>⭐</td>
<td>⭐</td>
</tr>
<tr>
<td>R: rhdf5</td>
<td></td>
<td></td>
<td>⭐</td>
<td>⭐</td>
</tr>
</tbody>
</table>
HPC Vendor Partner Program (NEW)

- HDF5 works best when
  - **HPC Vendors** work with HDF Group to develop versions of HDF5 to showcase and take advantage of unique customizations
  - **HPC Users** (e.g. programmers, scientists) have access to HDF5 expertise, particularly when starting out or delving into the more advanced features

- **Examples:**
  - Intel: support for DAOS-M… allowing existing apps built on HDF5 to support their next-generation object store
  - NCSA Blue Waters… working directly with scientists to build and improve their apps
  - European Light Sources centers (DESY + ESRF + DLS)… delivering advanced functionality (compression plugins + VDS + SWMR)
HPC Vendor Partner Program (NEW)

• We’re interested in working with vendors to
  • Create optimized and custom versions of HDF5: create a competitive advantage through unique configurations
  • Build scientific applications, including working with HPC prospects evaluating your platform as part of the sales process
  • Support platform vendors and their end users (generating additional revenue for platform vendors)

• As part of this proactive stance, the HDF Group will no longer provide platform-specific support unless we have a partner agreement in place
  • It takes a lot of work to build and maintain HDF5 to target platforms, and that work needs to be supported.
  • HDF Group will not longer be certifying, testing, or providing releases for platforms outside our Partner Network
  • Questions around 1. building for specific platforms or 2. vendor-created binaries (i.e. not certified by HDF Group) should be directed to the vendor
HDF5 1.10: Released in March 2016

- Concurrent Read Access (SWMR)
- VDS
- Parallel I/O performance improvements
  Collective metadata read and write
- New internal structures to support SWMR
- 1.10.0 is compatible by default with 1.8 and only incompatible when new features (like SWMR or VDS) are used
  h5format_convert (rewrite just metadata in place) to have 1.8 file
Concurrent Read Access (SWMR)

New data elements are added to the dataset in the file, which can be read by a reader... With no IPC necessary.
Virtual Datasets (VDS)

Can aggregate multiple source datasets into a single virtual dataset

Supports unlimited dimensions in both source and virtual datasets
Virtual Datasets (VDS)

- Extension to the existing selection API
- Multiple files can be used to write in parallel
- Virtual view of single dataset

```
start[0] = 0; start[1] = 0; start[2] = 0;
stride[0] = DIM0; stride[1] = 1; stride[2] = 1;
block[0] = DIM0;
block[1] = DIM1;
block[2] = DIM2;

status = H5Sselect_hyperslab (vspace, H5S_SELECT_SET,
                           start, stride, count, block);
status = H5Pset_virtual (dcp1, vspace, "f-%b.h5", "/A",
                        src_space);
```
Cache image
Saves cache entries in the file for restart

Page aggregation and buffering
- I/O performance improvement
- Avoids small I/O operations
- Uses fixed-size blocks/pages when writing HDF5 file

Avoid truncate
Avoids expensive file truncate operation on file close

Evict on close feature
Keeps metadata cache small by evicting MD items when HDF5 object is closed
HDF5 Roadmap: 2017 – 2018 (already in motion)

- Sub-Filing
- Parallel compression
- Additional features
  Driven by research projects
- Productized and added to maintenance releases through
  Exascale Computing Project (ECP)
Sub-Filing

Translate single file I/O to multiple files
Reduce file locking and contention
Existing VDS feature internally used

File: test.h5

Comm1: Process 4 & 5
1-D Dataset: G1-G2-D1
File: sub-test-1.h5

Comm2: Process 2 & 3
D1 Dataset: G1-G2-D1
File: sub-test-2.h5

Comm3: Process 0 & 1
D1 Dataset: G1-G2-D1
File: sub-test-3.h5

VPIC I/O Write Time on BlueWaters System

Number of processes

- Single File
- SF - 32
- SF - 128
Parallel Compression

Split up filtering overhead between multiple processes / additional communication cost

Current architecture enables parallel support for all types of filters

Filter Pipeline
Each chunk compressed individually

P0 P1 P2 P3

Compressed dataset
H5Tuner

Uses XML description file

Dynamic library redirection through LD_PRELOAD

Tuning Phase

Application → Extract I/O Kernel An Pattern → Look up for Tuned Parameters → Pairs of Patterns and Tuned Parameters → Pattern Previously Tuned? → YES → Tuned Parameter set (XML File)

Auto Turning System

Start → H5Evolve → Parameter Space → Parameter Setting (XML File)

Adoption Phase

Tuned parameter set (XML File) → H5Tuner Dynamic Library → HPC System → Application

HPC System → HDF5 File → H5Tuner Dynamic Library

I/O Benchmark → Executable → HPC System → HDF5 File

Experiment Database: Parameter Setting & Performance Results
Research Projects

- Extreme Scale Storage and I/O (ESSIO)
- Started back in 2012 (FastForward)
  - Asynchronous I/O (transactional)
  - Query/Indexing
  - Analysis shipping
  - Map object support
  - Data Integrity
Research Projects

- Software Defined Service (SDS) = Define HDF5 as a service, definition?
- Started in 2016
- Develop prototype building blocks (user-space)
  - Use Mercury (RPC) + Argebots = Margo
  - Enable reusability → rapid development of specialized services
  - BAKE API (Key-value store)
  - Fault detection and group membership
- HDF5 VOL plugin
- Extension to VDS to ”data federation” concept
Exascale Computing Project (ECP)

- Collaborative effort of DOE-SC / NNSA-ASC
- Accelerate development of a capable exascale computing system
- Phase 1: 2016 – 2019 timeframe

Credit: Rajeev Thakur (ANL)
ECP — ExaHDF5 Proposal

- Collaboration with LBNL and ANL
- Virtual Object Layer (VOL)
  New VOL Plugins: Format adapters for ADIOS and netCDF
- Query and Indexing
- Asynchronous I/O
Virtual Object Layer (VOL)

Virtual object layer provides the user with the HDF5 data model and API, but allows different underlying storage mechanisms

- Native HDF5
- Metadata Server
- Remote plugins
- PLFS plugin (Raw)
- IOD/DAOS-M plugin (Raw)
Query / Indexing

Performance comparison on attr=2 query

- No index
- Dummy index
- DB index
- MDHIM index

View created in memory
Data Index (FastBit)
Metadata Index (Berkeley DB / MDHIM)
Asynchronous I/O

• Non-blocking I/O allows asynchronous I/O (i.e., overlapping compute with I/O)
• Current HDF5 I/O calls are synchronous or blocking
• I/O is initiated within the library after API call
• I/O operation completes in the background after API call has returned
  Beneficial for both raw data and HDF5 metadata I/O
• Modification of VOL to support non-blocking calls
• Support for POSIX AIO, non-blocking MPI I/O, etc
• Question of progress
HDF5 Roadmap: 2017 – 2018 (open for discussion)

• **Enhancements to data model**
  Add key-value object to HDF5: “Map” objects

• **Improve fault tolerance**
  • Metadata journaling
  • Transactions

• **More efficient storage and I/O of variable-length data, including compression**

• **Full C99 type support** (long double, complex, boolean types, etc)

• **Full UTF-8 support**

• **Thread-safety**
Beyond HDF5 Roadmap (open for discussion)

- Industrial-grade compression libraries
- Spark: e.g. H5Spark, RDD VOL, etc.
- Cloud
- “I/O kernels”
  - Remove HDF5 bottlenecks discovered
  - Publish repository of I/O kernels with verified results
- Etc.

https://goo.gl/4TfpZ3
We are using Git now!

https://git.hdfgroup.org/projects/hdf5

We are accepting patches

- Contact help@hdfgroup.org
- Sign Contributor agreement
- Go through our SE process (code review, regression testing, documentation, etc.)
THANK YOU!

Questions & Comments?