**The Hierarchical Data Format (HDF): A Foundation for Sustainable Data and Software**

 Sustainable science depends on intricate relationships between data and software tools that access and analyze them. Software that cannot read data and data that can’t be read are both significant obstacles to sustainability. In the long-term, data must be preserved in well documented, self-describing formats that are accessible on multiple hardware platforms using a wide variety of programming languages. In addition, the formats must include mechanisms for including metadata that are required for using and understanding the data long after they are collected and processed.

 In the short-term, general commercial, open-source, and community tools must support data storage and analysis needs of multiple communities. In addition, sustainable formats must address specific data needs of multiple scientific communities in order to achieve the breadth of usage and support required for sustaining maintenance and development of new capabilities. In practice, this breadth is achieved through the development of conventions that map community science data types to data objects in the file (computer science data types). These conventions ensure interoperability by providing consistency in the data layer and isolating science users from the details of the underlying storage organization and format.

 The Hierarchical Data Format (HDF5) addresses the needs described above and forms a foundation for data storage and access in many scientific communities. The flexibility of the format is achieved through simplicity. HDF5 is a container format that includes three fundamental objects: datasets, attributes, and groups. Datasets are multi-dimensional arrays of atomic or compound datatypes (computer science datatypes). They provide the flexibility required for storage of diverse scientific data objects while attributes allow annotation of these objects with metadata required for use and understanding. Groups provide overall organization of datasets and attributes into associations that define scientific data objects.

 When combined with community specific conventions, HDF provides a robust and reliable foundation that scientific communities rely on for interoperability and high-performance storage of metadata and data. This combination naturally leads to a division of labor between the developers and maintainers of the format (The HDF Group) and the diverse communities that build conventional formats on top of HDF5. This paper focuses on practices and experiences of the second group that will help others understand their approach and build on their success.

**Scientific Facilities**

 Many important scientific problems require significant investments in equipment or facilities that must be amortized across entire communities. Scientists from all over the world either travel to the facilities to run experiments or use observations collected at/by these facilities to do science. Supporting these communities requires data formats and software that run in well-controlled situations at the facilities and in many environments on multiple platforms at the scientist’s home institutions. Multiple members of large and diverse science teams must also be able to freely share original observations and derived results without worrying about platforms, operating systems and other computer details. HDF5 is well suited to these demands and is being used successfully in several large facilities.

 The NASA flagship dust accelerator facility at the University of Colorado's Institute for Modeling Plasma, Atmospheres and Cosmic Dust ([impact.colorado.edu](http://impact.colorado.edu/)) is entirely standardized on HDF5.  The data product from this facility consists of an HDF5 container following a laboratory-determined convention, which represents particle impact events from the accelerator.  The "science data type" is a sequence of dust-particle events, each of which has an associated set of time series representing the experimental data, and metadata including the time of impact, grain mass and speed, and accelerator parameters. Crucially, the time series sampling rates and record lengths may vary from one dust event to another.

 The HDF5 standard data format easily accommodates these requirements. Dust particle events are stored as a series of HDF5 groups, tagged with the event metadata using HDF5 attributes.  The time series, stored as HDF5 datasets within the groups, are free to vary on an event-by-event basis.  HDF5 files produced according to this scheme may be read using facility-provided IDL scripts, but users are free to use any HDF5-aware analysis environment (IDL, Matlab, Python, etc.) to explore the files.  The use of standard HDF5 features like attributes and datasets means that "free-form" analysis in these environments is possible, without relying on the facility scripts or special-purpose reader programs.

 The Diamond Light Source (<http://www.diamond.ac.uk>) is the United Kingdom’s synchrotron, another large scientific facility. Diamond Light Source uses HDF5, specifically in the ‘Nexus’ format for most of their experimental data. They are in the process of migrating all beamlines to use this format and moving all our scientific data analysis algorithms to use it as a data source. HDF5 allows datasets much larger than will fit in machine memory to be handled elegantly – almost as if they were in memory.

 Diamond Light Source stores data in multi-dimensional arrays and has written a number of tools to work with the arrays and visualize data. Some examples are:

* HDF5 is built into the next generation of faster detectors.
* Visualization and slicing of data once it has been collected.
* Expressions involving data larger than will fit in memory can be written by users. HDF5 allows the data for evaluation of the expressions to be loaded as needed.
* Users can create a tool on one face of a stack of data and process the whole stack with that tool. For instance, image correction, peak and function fitting, azimuthal, radial and box integration.
* At any point using the analysis tools, users can save state or persist to a HDF5 file.
* We have developed a wide range of conversion tools which convert to and from HDF5 so that users with analysis codes not accepting the HDF5 format can run their algorithms as needed.

**Models and Tools**

 In other scientific communities, models are fundamental tools that increase understanding rather than experimental observations. HDF5 is used in many of these communities to support input and output from models as well as visualization.

 For example, in molecular chemistry and physics, molecular coordinates constitute the basic information for simulation software. In addition to spatial coordinates, many metadata fields are critical for complete understanding: velocities, masses, charges, etc. The scientific data items are very basic, scalars and vectors mostly, and are organized in arrays in which each element corresponds to a specific atom. Beyond the organization of the storage, an important challenge remains in the interpretation of the data in different contexts (biomolecular systems, simple fluids, etc.).

 H5MD <http://nongnu.org/h5md/>, has recently been proposed as an HDF5-based file format for molecular data (P. de Buyl et al, Comp. Phys. Comm. 2014). It solves the problem of data organization and definition within a HDF5 file and will serve as a common basis for further specialization. This process has started already, with K. Hinsen proposing a H5MD module for his MOlecular SimulAtion Interchange Conventions (MOSAIC).

 There are many similar examples of integrated simulation and visualization frameworks and formats built on top of HDF5:

* the F5 File Format (<http://www.fiberbundle.net>), an I/O library in C that implements a semantic layer inspired by the mathematics of Fiber Bundles on top of the HDF5 library
* the Silo framework (https://wci.llnl.gov/codes/silo/index.html) : A mesh and field I/O library and scientific database
* many simulation and visualization tools developed at TechX (http://www.txcorp.com).

**Sustainable Archives**

 Satellite remote-sensing archives are a special case of scientific facilities because they include a requirement to save the observations “forever” and support present and future users all over the world. The NASA Earth Observing System (<http://eospso.gsfc.nasa.gov>) selected HDF as the format for all their satellite data over twenty years ago and developed the HDF-EOS conventions for several common types of Earth observations and products (swaths, grids, points, and zonal averages). These conventions have been used for hundreds of NASA products and millions of data files.

 NASA’s EOS has lasted long enough that it has already encountered many real-world sustainability problems including 1) adoption of a standard format and related conventions across a wide variety of diverse science teams with their own requirements and priorities, 2) evolution of formats and conventions across major format versions (HDF4 vs. HDF5), 3) emergence of new programming languages, compilers, computer systems, and architectures, and 4) development of very high-resolution, multi-channel instruments with unanticipated observation geometries and significant long-term metadata needs.

 The sustainable archive challenges also face other long-term Earth observing systems such as the Joint Polar Satallite System (<http://www.jpss.noaa.gov>) and the Geostationary Operational Environmental Satellite R-Series (<http://www.goes-r.gov>).

**Conclusion**

 The proven ability of communities to store, access, and share modern scientific data and build data analysis and management systems they need on top of HDF5 is well demonstrated by the small set of examples described here (see <http://www.hdfgroup.org/HDF5/users5.html> for many other examples). Together these groups form a unique pool of knowledge and experience in sustainable science data and software. They have established that a single simple underlying format that supports flexible data organizations and access can address most of the needs of even the most demanding scientific disciplines.

 In addition, these groups have identified critical obstacles to sustainable scientific data and software. They are well summarized by the NeXus experience: The uptake of NeXus has been slow: at facilities or instruments that have working data acquisition and analysis pipelines the incentive to move towards NeXus is low. Especially as scientific software development is generally underfunded. What we see is that NeXus is the format of choice whenever new things are being built or a substantial upgrades are made. Another drive towards NeXus results from next generation x-ray detectors and neutron facilities being constructed. These generate data sets that are too large to be handled by the older existing schemes. Then NeXus and HDF-5 become the format of choice.

 The obstacles we are facing with NeXus are:

* NeXus is a voluntary effort; thus progress is very slow
* Lack of understanding for software issues in the scientific community and in management result of insufficient funding which in turn impedes progress.
* Many communities are not well enough organized to be able to negotiate, maintain and enforce a standard.

 These obstacles are really critical to understanding the problem of sustainable data and software: the social elements have overcome the technical elements in this equation. Actions required to overcome these obstacles are unfamiliar to the technical scientific community. Two important ideas seem worth pursing:

* Positive Deviance (http://www.positivedeviance.org): In every community there are certain individuals or groups who’s uncommon but successful behaviors and strategies enable them to find better solutions to problems than their peers - these are the positive deviants. We have Identifed those people who are headed in the right direction. We need to give them visibility and resources. Bring them together. Aggregate them.
* Collective Impact (<http://www.ssireview.org/articles/entry/collective_impact>): Technical problems are well defined: Their solutions are known and those with adequate expertise and organizational capacity can solve them... Adaptive problems are entirely different. They are not so well defined, the answers are not known in advance, and many different stakeholders are involved, each with their own perspectives. Adaptive problems require innovation and learning among the interested parties and, even when a solution is discovered, no single entity has the authority to impose it on the others. The stakeholders themselves must create and put the solution into effect since the problem is rooted in their attitudes, priorities, or behavior. And until the stakeholders change their outlook, a solution cannot emerge.