ExaHDF5: An I/O Platform for Exascale Data Models, Analysis and Performance

Prabhat[1], Quincey Koziol[2], Karen Schuchardt[3], E. Wes Bethel[1], Jerry Chuo[1], Mark Howison[4], Mike McGreevy[2], Bruce Palmer[3], Oliver Ruebel[1], and Kesheng Wu[1]

[1]Computational Research Division, Lawrence Berkeley National Laboratory, Berkeley, CA 94720, USA

[2]The HDF Group, Champaign, IL 61820, USA

[3]Fundamental and Computational Sciences Directorate, Pacific Northwest National Laboratory, Richland, WA 99352, USA

[4]Center for Computation and Visualization, Brown University, Providence, RI 02906, USA

Email: prabhat@lbl.gov

Abstract: Modern computational science faces serious bottlenecks in processing large datasets. Major advances in storing, reading, finding, analyzing, and sharing data are required for facilitating scientific discovery at the exascale. Our project is focused on optimizing the HDF5 library to run on efficiently on current petascale and future exascale machines. In collaboration with several domain scientists, we are developing easy-to-use data models to hide the complexity of parallel I/O. We are also extending FastBit, a state-of-the-art indexing and querying system, to work on distributed-memory, multicore platforms. We present our research agenda in this paper and conclude with a brief case study in interactive analysis of a massive 50 TB particle accelerator dataset.

1. Introduction

It is well accepted that one of the primary bottlenecks in modern computational and experimental sciences is coping with the sheer volume and complexity of data. Storing, reading, finding, analyzing, and sharing data are tasks common across virtually all areas of science; yet advances in data management infrastructure, particularly I/O, have not kept pace with our ability to collect and produce scientific data. This “impedance mismatch” between our ability to produce and store/analyze data continues to grow and could, if not addressed, lead to situations where science experiments are simply not conducted or scientific data not analyzed for want of the ability to overcome data-related challenges.

Our ExaHDF5 project consists of three thrust areas that address the challenges of data size and complexity on current and future computational platforms:

- We are extending the scalability of I/O middleware to make effective use of current and future computational platforms.
- We are incorporating advanced index/query technology to accelerate operations common to scientific data analysis.

- We are building on our existing work on data model APIs that simplify simulation and analysis code development by encapsulating the complexity of parallel I/O.

We are conducting these activities in close collaboration with specific DOE science code teams to ensure the new capabilities are responsive to scientists' needs and are usable in production environments.

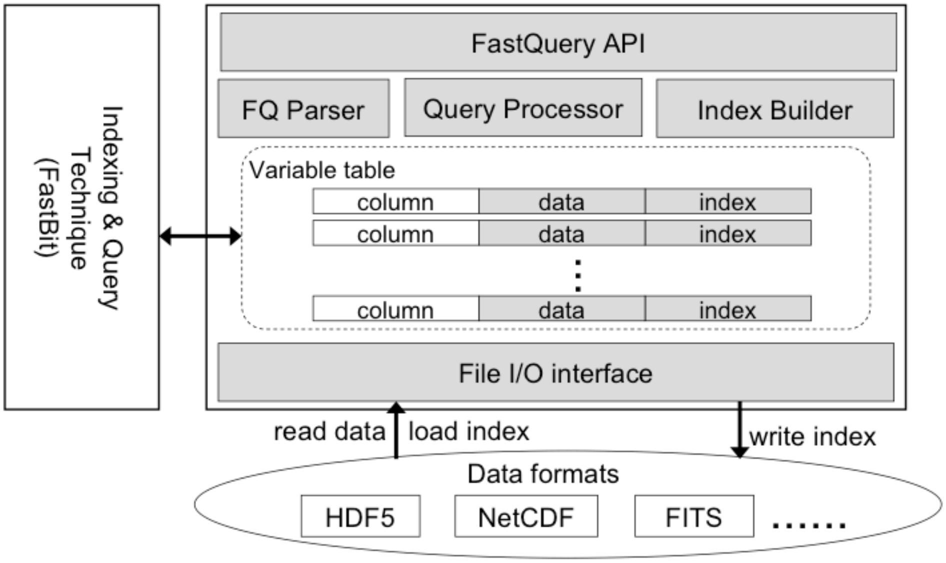

Figure 1: FastQuery Architecture

2. Our Approach

HDF5 Scalability: We are implementing a number of significant optimizations and improvements to the HDF5 library [4] to enable exascale I/O performance. These include the following:

- Streamlining HDF5 metadata modifications by removing collective I/O operations.
- Improved speed and scalability of append-only modifications to HDF5 files through file format and algorithmic changes.
- Supporting asynchronous I/O operations.
- Autotuning HDF5's behavior for the underlying file system.
- Incorporating state-of-the-art indexing solutions for data stored in HDF5 files.
- Adding fault tolerance to HDF5 by supporting new MPI extensions, as well as incorporating a new file update mechanism that makes the library robust to application failure.

Extreme scale analysis: As dataset sizes increase, it becomes infeasible for scientists to load and examine entire datasets. Typically, they expect an interactive system that allows them to specify, and refine, complex criteria that defines regions or elements of interest in the dataset. Conventional approaches based on loading entire datasets at each step of the refinement/display loop perform poorly in such a context. Building on our success with FastBit [5][6] and HDF5-FastQuery [2], we have developed a generic system called FastQuery [1], which applies state-of-the-art indexing and querying capabilities to generic scientific datasets. The current implementation can process datasets stored in popular file formats such as HDF5 and NetCDF. The FastQuery system architecture is shown in Figure 1. Our design is extensible and can accommodate other file formats in the future. Our current FastQuery implementation can work in serial on a single core, in a multi-threaded mode on multiple cores, and in a hybrid-parallel fashion to utilize distributed memory multi-core machines. We present a case study in applying FastQuery to a large scientific dataset in Section 4.

Exascale Data Models: We are developing data models in the context of three applications areas that are of strategic importance to DOE's mission: climate modeling, groundwater modeling, and accelerator modeling. For groundwater and accelerator modeling, we are leveraging our expertise with the H5Part and H5Block data models for handling particle and block structured data[3]. For climate modeling, we are building upon our existing work for handling geodesic grids for global cloud resolving models. We expect this work to play a critical role in terms of presenting real scientific applications with a high-performance as well as high-productivity interface for doing parallel I/O.

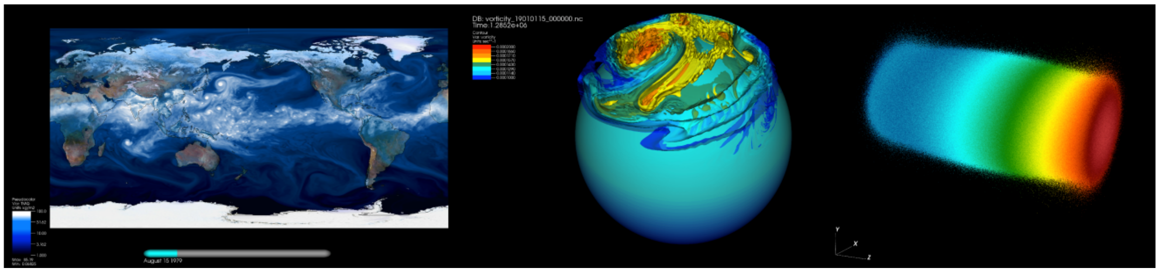

Figure 2: VisIt screenshots of CCSM, GCRM, and IMPACT-Z data.

3. Science Collaborators

Global Cloud Resolving Models (*Dave Randall, Colorado State University*). The largest source of uncertainty in climate models can be attributed to modeling of clouds. To simulate climate at high resolutions (<10 km), and scale to hundreds of thousands of processors, climate models are moving to new grid representations such as the geodesic and cubed-sphere grids. We are addressing a number of challenges resulting from the high-resolution application of the geodesic grid: developing a parallel I/O infrastructure to efficiently write large volumes of data (0.5 TB per snapshot), developing APIs to simplify integration with GCRM code, and efficiently reading and analyzing the data.

Community Climate System Model (*Mariana Vertenstein, UCAR*). CCSM is comprised of models for the atmosphere, ocean, land, and sea ice. The next generation of CCSM simulations will run at 25 km spatial resolution and a 10-minute temporal resolution. The total integration period can span anywhere from centuries to millennia. We are working with the CCSM team to tackle I/O challenges resulting from these massive simulation runs. We are optimizing the NetCDF-4/HDF5 PIO layer in the CCSM code to enable large-scale runs on systems such as Hopper and Jaguar.

Groundwater Modeling (*Tim Scheibe, PNNL*). Simulations of subsurface flows are moving to higher resolutions and incorporating more sophisticated physics and chemistry. Because of the large uncertainties associated with parameterizing the subsurface, there is also increasing interest in running ensembles of simulations that can be used either to estimate the distribution of outcomes or to perform inverse modeling that can provide estimates of parameters that best match measured behavior. All these trends are increasing the volume of data produced by these calculations. Particle-based simulations using smoothed particle hydrodynamics (SPH) will soon use billions of particles. In this project, we are working on incorporating HDF5-based data models into the SPH code. We are also profiling and optimizing parallel I/O for these codes to enable them to run at massive concurrency.

Accelerator Physics (*Robert Ryne and Ji Qiang, LBNL*). Particle accelerators are extremely important instruments for scientific research and discovery. Previously accelerators were seen mainly as tools for research in medium- and high-energy physics. Now, in addition, the nation's light sources and neutron sources are seen as essential tools for research in materials science, chemistry, and the biosciences. In this project, we are working with the IMPACT/MaryLie suite of codes. We are working on integrating the H5hut data model, optimizing parallel I/O on large-scale runs, and streamlining visualization and analysis capabilities for particle physics researchers

4. Accomplishments

The ExaHDF5 project is relatively new, and much of the proposed research activities are currently underway. Nevertheless, we present the latest developments in our project in the form of an end-to-end case study applied to the accelerator physics collaboration. We have been successful in incorporating H5Part calls into the IMPACT-T and IMPACT-Z simulation codes. We were able to obtain a collective write performance of ~5 GB/s with IMPACT-Z running on 10,000 cores on hopper, a Cray XE6 platform at NERSC. This problem configuration used 1 B particles for each timestep, and ran for 750 timesteps, thereby generating 50 TB of data. We then applied the parallel FastQuery infrastructure to build indices for the dataset on 1,920 cores in 12 minutes. After the indices were created, we ran strong scaling tests on a number of cores; the results are shown in Figure 3. In general, we see good scaling performance and are able to resolve queries on 2,880 cores in 35 seconds. In comparison, a technique based on examining the entire dataset would take nearly 7 minutes.

The queries in this case study were designed by our science collaborators: they were interested in examining the data for *halo particles*; these particles exhibit large transverse amplitude and result in component damage. Our quantitative analysis results provided scientists with a novel capability for interactive beam diagnostics. In this case, they had suspected that the latter half of the beam accelerator design was suboptimal; the large number of halo particles shown in the query results confirmed this suspicion.

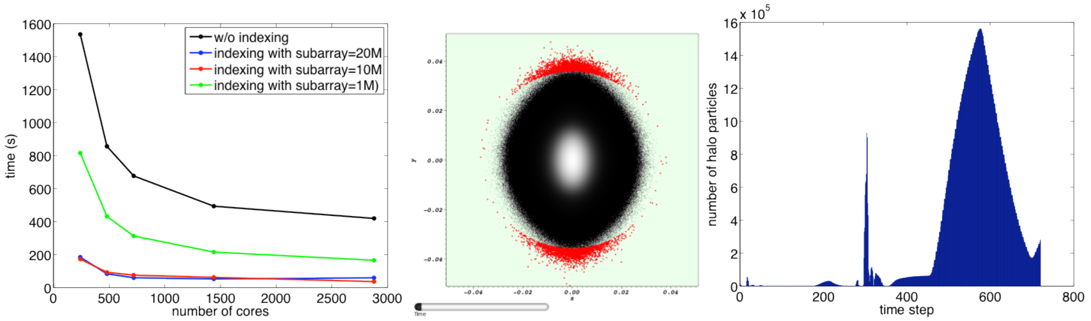

Figure 3: Strong scaling results for halo query resolution (left). Halo particles are shown in red in the center. Plot of halo particles across the complete timeseries is shown on the right.

Acknowledgments

We thank John Shalf, Katie Antypas and Francesca Verdier for assistance with I/O issues at NERSC. We also thank Brian Austin, Ji Qiang, and Rob Ryne for providing us with the particle accelerator dataset. The work conducted here used computational facilities provided by the National Energy Research Scientific Computing Center. This work was supported by the Director, Office of Advanced Scientific Computing Research, Office of Science, U.S. Department of Energy, under Contract No. DE-AC02-05CH11231.

Bibliography

[1] Chou, J., Wu, K., & Prabhat. (2011, July). FastQuery: A general indexing and querying system for scientific data. *SSDBM Poster* .

[2] Gosink, L., Shalf, J., Stockinger, K., Wu, K., & Bethel, E. (2006). HDF5-FastQuery: Accelerating complex queries on HDF5 datasets using fast bitmap indices. *SSDBM*, (pp. 149-158).

[3] Howison, M., Adelmann, A., Bethel, E., Gsell, A., Oswald, B., & Prabhat. (2010). H5hut: A High-Performance I/O Library for Particle-Based simulations. *IASDS.*

[4] The HDF5 Group. (n.d.). *HDF5*. Retrieved from http://www.hdfgroup.org/HDF5/

[5] Wu, K., Otoo, E., & Shoshani, A. (2006). Optimizing Bitmap indices with efficient compression. *Transactions on Database Systems , 31*, 1-38.

[6] Wu, K., Shoshani, A., & Stockinger, K. (2010). Analysis of multi-level and multi-component compressed bitmap indexes. *Transactions on Database Systems* , 1-52.