In-situ Scientific Data Processing for Extreme Scale Computing

S. Klasky, Q. Liu, J. Logan, N. Podhorszki, R. Tchoua (ORNL)
K. Schwan, M. Wolf (Georgia Tech)
M. Parashar (Rutgers)

The U.S. Department of Energy established leadership-computing facilities in 2004 to provide scientists with capability computing for high-profile science. Since inception, the system capacity has grown from 14 TF to 1.8 PF, an increase of more than a factor of 100, and it will increase by another factor of 100 in the next five years. This growth, along with computing policies that enable users to run at scale for long periods, have allowed scientists to write unprecedented amounts of data to the file system. At the same time, the effective speed of the I/O system (time to write full system memory to the file system) has decreased, going from 350 seconds on ASCI *Purple* (49 TB at 140 GB/s) to 1500 seconds on *Jaguar* (300 TB at 200 GB/s). As future systems will further intensify this imbalance, we need to extend the definition of I/O to include that of I/O pipelines, which blend together self-describing file formats, staging methods with visualization and analysis "plugins," and new techniques to multiplex outputs using a novel Service-Oriented Architecture (**SOA**) approach—all in **an easy-to-use** I/O componentization framework that can add resiliency to large-scale calculations. Our approach to this has been via the community-created **ADIOS** system, which is driven by many of the leading-edge application teams and the top I/O researchers.

Our approach addresses a conceptual shift in addressing I/O performance bottlenecks by addressing the needs of extreme-scale computing. To achieve this performance, we propose a multipoint approach to handling some of the biggest concerns and discuss a framework that has been developed with the following guidelines:

1. A Service Oriented Architecture, where the framework should be easy for new programmers to write individual "plugins" that can be compiled and either run independently from files, or *in-situ*, where users can introduce their services at runtime.
2. *In situ* processing, where we can place domain-specific services or plugins without incurring any additional cost of data movement.
3. Data Staging, which our group pioneered for HPC computing, using the fusion PIC codes as the first use cases, which allow the realization of the formation of I/O pipelines.
4. Data Management techniques, which can reduce the data being moved, and placed to the file system, using lossy and lossless compression techniques, data indexing, and multi-resolution techniques.
5. Monitoring techniques to transparently allow groups of scientist to monitor their simulation on web dashboards.
6. Usability of optimizations, which reduces the complexity to users, but allows for high performance on different platforms, without the need for experts.
7. Next generation file formats, to deal with a new generation of data requirements from application scientist.