Title of Pre-application: Hierarchal Extreme Scale Knowledge Management

Principal Investigator: Scott Klasky, Group Leader, ORNL, 854-241-9980, klasky@ornl.gov

Funding Opportunity Announcement Number: DE-FOA-0001338

List of all co-PIs and Key/Senior Personnel

Hasan Abbasi, Oak Ridge National Laboratory
Mark Ainsworth, Oak Ridge National Laboratory JFA, Brown University
Matthew Curry, Sandia National Laboratory
Jay Lofstead, Sandia National Laboratory
Carlos Malzahn, U. Cal. Santa Cruz
Manish Parashar, Rutgers University, Oak Ridge National Laboratory
Lee Ward, Sandia National Laboratory

Objective: Exascale scientific discovery will introduce more complex hardware, and many simulations will be bottlenecked without sufficient new research into manging and storing the large data which will be produced during the simulation, and analzyed for months after the simulations.

Our goal is to explore and address the multi-tier challenges that are faced by scientists in creating and managing and storing their data to expedite insights into mission critical scientific processes in exascale computing. We propose a research program that aims to explore a hiearchical organization and storage infstraturure which will facilitiate how we data can be written and read efficiently. In order to help us explore our many research issues, we will build upon the sucess of our middleware system, ADIOS, our multi-tier storage system, Sirocco, and our distributed storage system, Cepth.

We will also study ?? Finally, we will, jointly with application partners, study the behvior of how data gets written and read from the storage layers, and what key inishgts can be used in the next generation LCF systems, along with future exascale systems.

Key Technical Approach: The primary contribution of our research are insights into how to build a hiearchical organization and storage of massive scientific data sets generated from exascale computations along with the necessary information needed for advanced data validation and exploration from experimental and observational data. We are fundamentally trying to address the fundmantal questions : 1) How can we place and mangage massive scienfici data across all of the tiers of the storage and memory hiearchy? 2) What are the proper semantics needed in order to help bring knowledge from the applications to the middleware layer. 3) What information from the storage layer can be exposed to the middleweare such that the placement of information can be agreed upon from what the application requires and what the system resources are available.

Our approach will allow users to "plug-in" techniques to classify data, not as bytes but as motifs, where we can understand relationships between objects (into data models) and relationships of data in variables. This will allow us to adapt data from a variable into the various storage tiers. Rather than focus on simple data compression, we will incoroarte a multitude of techniques to reduce data on the faster storage tiers, and keep the less reduced information on the lower tiers.

We wish to support additional data access modes as well. First, within a given accuracy bounds, offer a mode for recomputation from data stored in a "fast" tier to reduce data latencies. Second, exploratory analysis operations frequently entail an overall data set view followed by targeted data exploration based on identified features. We will offer support for a configurable data access mode to support these sorts of analysis accesses. An "overview" access mode that gives a quick, approximate within error bounds, data view that can guide feature selection offering rapid coarse-grained data exploration without requiring loading the complete, detailed data set from storage. Based on the granularity requested, the accuracy and size of the data returned can be adjusted. At the most extreme setting, the original data can be retrieved at the time cost of moving the

potentially huge data quantity. Third, to ensure available storage for subsequent operations, we will offer automatic data migration based on user annotations for required data lifetimes using monitoring and learning techniques. Unlike existing approaches, this will be tempered both by the user annotations and through learned access patterns. While past access patterns may not indicate future access because the simulation run purpose may have changed, we are focused on scalability where runs are subsequently larger as the simulation prepares for a capability run. By learning from the output and access patterns during this run sequence, we can accurate decide how to place and organize data for the critical capability runs. Fourth, we anticipiate storing mutliple data copies, each compressed in different ways according to the underlying media, some of these copies will disappear based on storage pressures, but data persistence will be maintained according to user specifications. Assuming a relatively low latency cache layer before a tape system, we can offer exporatory data access reserving pulling data from tape to just the data required. This will save scientists time and make data stored on tape usable without long delays.

Our research efforts will be heavily focused on the need to, in a coordinated manner, adapt data and metadata retention policies to the dynamic resource balancing that will need to take place between the application, OS/R, and hardware.

The success of this project will provide insights into how to build middleware and storage layers which can interact well with each other, and take user-provided hints. Today data is reduced by application scientist who have limited information on what the storage layer can provide. They often make comprimises based on this limited knowledge and either tune their output for writing or reading performance. This data then gets moved to other locations, and much of the tuning is lost when the data is read back during their post processing. Furthermore, there is a limited set of operations which users will be able to stage to other staging nodes for real-tiime-reduction and visualization.

We will identify, through concrete application evaluation, the requirements for highly usable and scalable middleware and storage layers