Framework for Integrating End to End SDM Technologies and Applications (FIESTA).

Roselyne Barreto, Scott Klasky
Oak Ridge National Laboratory
PO BOX 2008
barreto@ornl.gov, klasky@ornl.gov,

Chen Jin, Norbert Podhorszki
Oak Ridge National Laboratory
PO BOX 2008
cgj@onrnl.gov, pnorbert@ornl.gov

ABSTRACT
Petascale computing is now upon us with the release of Road Runner at LANL, and Jaguar at ORNL. These computers allow for new classes of problems which create unprecedented amount of data. These computers have also grown in complexity, allowing only the minority of scientist being able to take advantage of them. Not only is it difficult to run on these systems, but analyzing, visualizing, and managing the data has also grown even more complex. There is a new growing need for powerful collaborative tools to visualize and analyze the data these simulation, and easily manage everything in one end to end system. In this paper we focus on our new dashboard-workflow approach to presenting, analyzing and sharing data from physics simulations. We present the FIESTA approach for integrating a collaborative environment for petascale simulations and analysis.

Author Keywords
Visualization, collaboration, workflows, dashboard, provenance, analytics.

ACM Classification Keywords

INTRODUCTION
Leadership class computing has enabled a new age of computing where simulations can run for weeks on hundreds of thousands of processors. These simulations typically address multi-physics, and the complexity of coupling the different areas of physics is staggering. A key requirement for addressing this challenge is effective visualization and analytics support that can handle the scale and complexity of these simulations.

As the complexity of computers and simulations increase monitoring the information they generate becomes a major challenge. An essential part of undertaking this massive task is to have the visualization and analytics tools necessary to help extract meaningful knowledge from the substantial datasets simulations create. Furthermore making these tools collaborative becomes an implied requirement as in most cases they are not targeting one scientist alone but an entire team working on a specific project. They have to be easy to use for a variety of researchers with different expertise from pure application scientists to computational scientists for instance and establish easy communication between different team members.

Central to what we are creating is a "**data store**" - a database, which stores all of the information, and collects all the provenance information about the simulations and the entire lineage of the data. This allow researchers to look at their own piece (or other pieces they are given permissions to view), and be able to understand which codes they ran in a coupled simulation, what the performance of the code was, and what the data produced from the other pieces of code they coupled to was. If a researcher sees a problem he/she can send messages to the other collaborators, and as that is done the relevant information (codes used, monitors used, etc.) would automatically be shared. Because we need to automate all of these tasks, we believe that it is important to do this within a strong **workflow** framework.

Another key requirement is the need for highly annotated, fast and scalable I/O systems. We have designed the ADaptable I/O System (**ADIOS**)[1], to allow researchers easy ways to achieve high performance I/O while maintaining highly annotated I/O which can talk directly to our data store.

Another key requirement for the collaboration space is that it needs to be **easy to use** regardless of the end-user information technology expertise - from the theorists who commonly use their pencil and paper, to the experimentalists, applied mathematicians, computational scientists, and computer scientists. This requirement has been one of the central design themes of the infrastructure presented in this paper. By placing the burden of the system on the backend services, we reduced the complexity of software that one must install. One of our key goals is to create a unifying software infrastructure to make it easy for scientists to collaborate and use the data analysis tools that are common in their workplace. For instance, in the workflow we describe below, we use several visualization and analysis tools, but hide their complexity from the end users. Our overarching theme is simplicity in the interface of our software that we provide.

CHI 2009, April 4–9, 2009, Boston, MA, USA.

The rest of this paper is organized as follows. First we describe some scientific use cases, then in the subsequent two sections we describe our I/O, dashboard and monitoring workflow systems, followed by the analysis workflow section. Next, we discuss keeping track of provenance throughout the monitoring and analysis phases. We subsequently address the security and the collaborative aspects of the whole system before concluding in the last section.

SCIENTIFIC USE CASE, THE FUSION SIMULATION PROJECT

Our motivation for building the collaborative visualization space described in this paper is the ITER [2] project. ITER is an international research and development project that aims to demonstrate the scientific and technical feasibility of fusion power. In order to make this project successful, the United States is attempting to launch a Fusion Simulation Project (FSP) [3] which aims to leverage leadership class computing to develop a predictive capability for integrated modeling of magnetically confined burning plasmas. The knowledge that we gain from the FSP will ultimately enhance our understanding of data from burning plasma discharges and provide new opportunities for scientific discovery.

The FSP will be one of the largest collaborations ever attempted by the magnetic fusion program. This means that it will be necessary for the program to create collaborative infrastructure which will allow the scientists to work together to solve the complex problems. This is a daunting task. One of its most important aspects is creating a collaborative space that enables monitoring and documentation of each simulation, and sharing of this information with collaborators, and allowing for fast and scalable I/O for all of the data these simulations will produce. Given that this is a research program, we understand that the data will be fluid, changes in codes will be common, and researchers need to understand their necessary pieces and their roles in the project.

COMPONENTS INSIDE FIESTA

A picture of the FIESTA components is shown in Figure 1. We see that the foundation technologies are in the I/O system, workflow, and dashboard. The Analytics, code coupling, wide area data movement, and provenance and metadata capture are enabling technologies built on top of our foundation technologies. Next we describe several of the pieces of FIESTA.

ADIOS

The Adaptable I/O System, or ADIOS, is a componentization of the I/O layer. It provides an easy-to-use programming interface, which can be as simple as Fortran file I/O statements. ADIOS abstracts the I/O metadata information and data structures from the source code into an external XML file, which can reduce code pollution and create the connection between high-level APIs and the underlying I/O implementation details, as well as other technical descriptions, such as buffering and scheduling. By separating the detailed I/O implementation from the APIs, ADIOS also allows users to change how I/O is performed, i.e., by simply changing the declaration of the transport methods in the XML file without the need for source code modification.

One of the central themes of ADIOS is that it allows users to place multiple I/O implementations for different groups, such as writing restarts, or 3D diagnostics. By giving the users the ability to write data out with both high performance MPI-IO, and talking to our data store, to give the metadata to our backend datastore, we can simply in the design of our workflows, and allow the backend analysis programs a chance to fully understand the schema inside of the datasets.

Dashboard features

Our proposed user interface or "dashboard" is called eSimMon; it consists of a flash-based front-end combined with a PHP back end. Flash is a multimedia graphics program developed for the Web. It uses vector graphics and deploys consistently across all major browsers, desktops and operating systems. By using flash we reduce the software users have to install and we allow them to easily visualize data without a strong background in any specific graphics tools. The purpose of the eSimMon is to provide simple "one-stop shopping" access to the status of simulations, launch new simulations, and analyze/re-visit existing runs [4]. We make user experience a constant priority by placing the burden of complexity on the strong workflow back end. There are two main views/sections to the dashboard: the first gives the status of some DOE machines at a glance, the second focuses on monitoring a specific, running or past, simulation runs (called ***shots***). The first section called machine monitoring shows the activities on DOE supercomputers emphasizing the users' and their collaborators' running, eligible and old runs on the systems. Users can customize this view depending on their systems of interest and other preferences such as font and browser size. The simulation monitoring provides access to output from the simulation such as images, movies, input files and source code. Other capabilities include annotating movies, saving electronic notes and viewing provenance information. This is also the view in which users can download data or perform analysis on the simulation results.

Monitoring Workflows

The dashboard's function is to manage the large and complex scientific data generated from the simulations. An example of the dashboard monitoring feature is shown in Figure 2. We see that the dashboard display movies representing scientific variables and give access to post-processing and collaborative features on the data gathered from the simulations/experiments. The data from the simulations is delivered and processed by workflows in our environment [5]. A monitoring workflow transfers data from the running simulation on the fly, and performs all file conversion, data analysis and visualization of the scientific variables. Images of variables of individual simulation time steps and finally movies of variables spanning the whole simulation time are put into a place accessible from the dashboard. Datasets in standard format, like NetCDF or HDF5, are also made available for the dashboard as well as textual input files used by the simulation. Additionally, the workflow archives all data, images and movies automatically, so that users can access them from the dashboard later when disks are wiped off. The communication between the workflow and the dashboard is performed one way; all information necessary for the dashboard to find a workflow run, the movies and the data is recorded by the workflow in the dashboard's database.

Analysis Workflows

While the monitoring workflow delivers the data from the simulations and provides the pre-defined visualizations of their variables, other workflows can be used to post-process the gathered information. The goal of the analysis workflow on the eSimMon is to help the scientists extract knowledge from an exponentially increasing amount of simulation data. There are two approaches to integrating analysis on the dashboard. The first is to do a survey of commonly used analysis routines across applications and implement them. The second is to support "plug-in" capabilities and enable users to insert their own customized analysis routines into a pipeline provided by the dashboard. The initial version of the eSimMon "calculator" is a proof of concept for the first type analysis workflows. From the dashboard users have access to movies of different simulation variables. Without knowing the location of the data used for the visualization users can perform simple operations such as additions and subtractions on the corresponding variables by simply selecting movies and operators from the calculator. The complete progression requires tracking provenance information for the selected variables [6], generating an analysis routine for the operation chosen by the user, generating one image per simulation time step and creating a movie using these. The process is simplified for the users so that they do not need any knowledge/expertise in provenance nor in analytics to use this dashboard tool. Similarly, the eSimMon offers a list of statistical operations as a second illustration of predetermined analysis workflows. This particular example uses a package called parallel R. Given the massive quantities of data produced by simulations it is often useful if not critical to provide users with scalable high performance statistical data analysis tools. Parallel R lets scientists employ a wide range of statistical analysis routines on high performance shared and distributed memory architectures without having to deal with the intricacies of parallelizing these routines [7]. While these types of workflow are being integrated and continuously improved we should keep in mind that a number of scientists have their own adapted analysis scripts that they typically use on simulation output. Providing them with an easy way to use these routines from the dashboard should be a key specification of future analysis workflows. We must have an option to let users upload their own analysis routines and have workflow created to execute these on demand.

Provenance

The provenance information or metadata is the glue that links the dashboard and workflows together. As it progresses the workflow records the many operations and lineage information of data into a provenance database which enables the dashboard to follow the evolution of the processing. The provenance information establishes the link between user-selected movies on the dashboard and the location of the raw data that he/she wants to perform analysis on. Users also may want to download the file(s), which generated a particular movie. Another illustration of provenance use on the dashboard is the access to input files used for a particular simulation run, to the source code of the executed simulation, and to the information about the building/execution environment. In other words provenance is not only used throughout the dashboard as an add-on feature but also indispensable for most its functionalities. There are four categories of recorded information:

Process provenance which is about the steps performed in the workflow, the progress through the workflow control flow, etc.

Data provenance which records history and lineage of each data item associated with the actual simulation (inputs, outputs, intermediate states, etc.);

Workflow provenance which is about the history of the workflow evolution and structure;

System provenance which is about recording of all external (environment) information relevant to a complete run. This information includes compilation history of the codes, information about the shared libraries used by the codes, source code related information and run-time environment settings of each participating machine, etc.

The Provenance information we collect should allow complete end-to-end audit of any individual runs. The analysis workflow should also record detailed flow information and make it available to eSimMon so that users are able to reuse/share analysis results and routines.

Security

The analysis features of the dashboard requires an infrastructure behind the scene for executing the analysis workflows and parallel R programs on a high performance system, in the name – and on the account – of the user who is using the dashboard. The dashboard's server side is running on a dedicated web-server, which does not have user accounts. The web server has its specific authentication mechanism to identify the users and distinguish between them when they use the dashboard but all operations on that machine are performed under the web server's own user account. The Grid Security Infrastructure (GSI) [8] addresses the problems of authentication of users and single sign-on capability for a distributed system and Grid tools are available to enable a system to provide computational services for users using certificates. Users can upload so called proxy certificates to the dashboard, where the server can use them to authenticate in the name of the user to a remote system and execute jobs there.

COLLABORATIVE FEATURES

When using eSimMon a scientist has the capability of "tagging" other dashboard users as his/her collaborators allowing them to look at his/her shots. This collaboration can be extended to sharing not only shots but also movie annotations as well as electronics simulation notes for example. We make a distinction between passive and active collaborative work. Passive collaboration refers to work being done by a user independently of what his/her collaborators are working on at the same time. The ability to compare shots, to share notes, graphics annotations, analysis results and routines establishes a communication between members of a team working on the same project. Depending on the relationship between them the collaboration may be a peer-to-peer comparison or it could be a complimentary interaction such as one researcher requesting comments from a performance expert etc. Scenarios in which users share thoughts and ideas vary with the different use cases. The goal remains to share thoughts and ideas about the science. The concept of active collaboration simply adds the real-time element to this exchange. It is implemented as *sessions*. The session manager keeps track of the entire system. When multiple users are looking at a particular shot, the session manager keeps track of their actions and propagates changes back to all collaborators. This allows sharing of notes, comments, scripts and even animations with others in real-time. For example sessions allow a "master" user to *drive* the session and play a movie for all logged on collaborators.

CONCLUSIONS

By employing state of the art I/O, dashboard and workflow technology, scientist can now begin to easily manage the large amount of data being produced on leadership class computers. One of the keys to our system is that the workflow system gathers and processes simulation information and feeds it to the dashboard using the provenance database. By gluing together the I/O system, workflow automation for monitoring, and analysis, and our dashboard, users can easily begin to monitor and analyze data from petascale simulations. Our eSimMon dashboard provides scientists with a user-friendly, user-oriented interface for easy access, visualization and analysis of simulation data. The dashboard allows users to grant access to others on their simulation runs.

ACKNOWLEDGMENTS

We thank ORNL, the SDM center, Georgia Tech, and CPES.

REFERENCES

1. Lofstead, J. F., Klasky, S., Schwan, K., Podhorszki, N., and Jin, C. 2008. Flexible IO and integration for scientific codes through the adaptable IO system (ADIOS). In *Proceedings of the 6th international Workshop on Challenges of Large Applications in Distributed Environments* (Boston, MA, USA, June 23 - 23, 2008). CLADE '08. ACM, New York, NY, 15-24.
2. ITER Home Page: http://www.iter.org
3. Tang, W., "Scientific and computational challenges of the Fusion Simulation Project (FSP)", SciDAC'08, Journal of Physics: Conference Series 125 (2008) 012047
4. "Collaborative visualization spaces for petascale simulations", Klasky et al., Collaborative Technologies and Systems: International Symposium on Collaborative Technologies and Systems, pp. 203-211, May 2008.
5. J. Cummings, A. Pankin, N. Podhorszki, G. Park, S. Ku, R. Barreto, S. Klasky, C. S. Chang, H. Strauss, L. Sugiyama, P. Snyder, D. Pearlstein, B. Ludäscher, G. Bateman, A. Kritz, and the CPES Team, "Plasma edge kinetic-MHD modeling in tokamaks using Kepler workflow for code coupling, data management and visualization", *Communications in Computational Physics* special issue for the *20th International Conference on the Numerical Simulation of Plasma* (Austin, TX, Oct 2007).
6. Vouk, M et al. "Automation of Network-Based Scientific Workflows", "Grid-based Problem Solving Environments, IFIP, Volume 239, eds. Gaffney. PW, Pool JCT (Boston: Springer), pp 35-61, 2007
7. Samatova, N. et al, "High performance statistical computing with parallel R: applications to biology and climate modeling", Journal of Physics, 2006, 46, 505-509.
8. Foster, I., Kesselman, C., Tsudik, G., and Tuecke, S. 1998. A security architecture for computational grids. In Proceedings of the 5th ACM Conference on Computer and Communications Security (San Francisco, California, United States, November 02 - 05, 1998). CCS '98. ACM, New York, NY, 83-92.

Figures

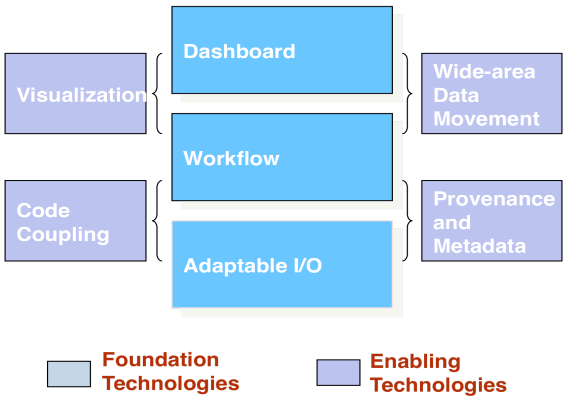

Figure 1. Components of FIESTA

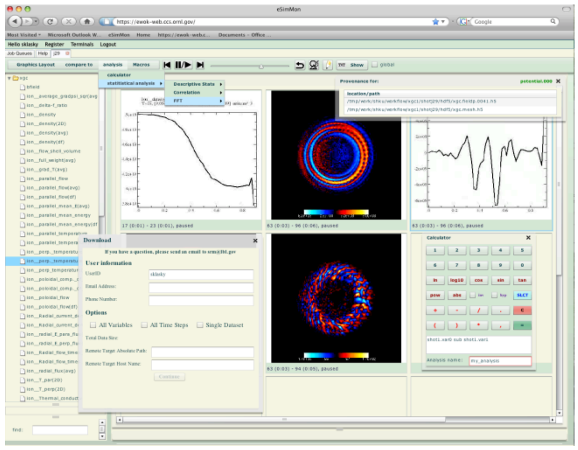

Figure 2. Simulation monitoring in the eSimMon dashboard