Collaborative Monitoring and Analysis for Simulation Scientists

Roselyne Tchoua, Scott Klasky, Norbert Podhorszki
Oak Ridge National Laboratory
{barreto, klasky, pnorbert}@ornl.gov

Brad Grimm, Ayla Khan, Emanuele Santos, Claudio Silva
University of Utah
{bgrimm, ayla, emanuele, csilva}@sci.utah.edu

Pierre Mouallem, Mladen Vouk
University of North Carolina
{pmouall, vouk}@ncsu.edu

ABSTRACT

Collaboratively monitoring and analyzing large scale simulations from petascale computers is an important area of research and development within the scientific community. This paper addresses these issues when teams of colleagues from different research areas work together to help understand the complex data generated from these simulations. In particular, we address the issues when geographically diverse teams of disparate researchers work together to understand the complex science being simulated on high performance computers. Most application scientists want to focus on the sciences and spend a minimum amount of time learning new tools or adopting new techniques to monitor and analyze their simulation data. The challenge of eSimMon, our web-based system is to decrease or eliminate some of the hurdles on the scientists' path to scientific discovery, and allow these collaborations to flourish..

KEYWORDS: Simulation, analytics, visualization, collaboration, dashboard, workflows, provenance.

1. INTRODUCTION

Leadership class computing offers an extraordinary amount of computing power to scientists at three different DOE centers: Argonne National Laboratory (ANL), Oak Ridge National Laboratory (ORNL), and the National Energy Research Scientific Computing Center (NERSC). For example, in 2010, The Leadership Computing Facilities at Argonne and Oak Ridge National Laboratories will provide over one billion processor hours to the INCITE program. As the complexity of the computers increases so does the complexity of the physics they simulate. For most of the simulations we are working with, this translates into an increase in the amount of data the simulations generate. For example, simulations produced by the GTC [1] code generated over 60 TB of data, which took months of analysis to gain insight into turbulence transport in tokamaks. Monitoring and analyzing results from large simulation runs often involves several research institutions and teams of scientists with diverse backgrounds including physics and computer science.

Our data management team's main goal is to accelerate the scientific discovery process by developing middleware and dashboards that allow users to generate fast metadata-rich I/O, and then perform the mundane tasks needed to present physicists with meaningful scientific results. To that end, we built the Framework for Integrated End to End Scientific Data Management Technologies for Applications (FIESTA) shown in Figure 1. The foundation components of FIESTA are: 1) fast adaptable I/O, 2) workflow management, and 3) a collaborative portal we refer to as the eSimMon [2] dashboard. FIESTA's auxiliary components are components for code coupling, provenance and metadata capture, wide-area data movement, and visualization.

The adaptable I/O System (ADIOS) [3] provides access to a variety of data transport mechanisms and data formats. The primary benefit of using ADIOS is that it provides portable metadata-rich output along with very fast, scalable I/O. The Kepler workflow system [4] orchestrates parallel tasks on multiple computer platforms and feeds a multitude of output to the dashboard including: images, movies, input files, performance data, and source code. We have integrated ADIOS and Kepler by providing a custom transport method in ADIOS that can talk to Kepler's provenance database. This information is accessible by eSimMon to be presented to users using provenance metadata to make data access fast and efficient.

In this paper, we focus on the front-end of the framework and describe how it is used to monitor jobs on the supercomputers to facilitate analysis by teams of scientists working in cooperation on a specific project. We describe our experience with a number of theoretical scientists, which is the driving force behind eSimMon's development. Understanding the way scientists work and where they most need our help is key in delivering a tool they will actually use. In Section 2, we describe in more details the motivation for this work; in Section 3 we cover related work. Section 4 emphasizes the relationship between user feedback and feature development. Section 5 describes the eSimMon dashboard and the supporting systems behind the Graphical User Interface (GUI). In Section 6, we describe the current infrastructure in place for analysis at ORNL followed by the different analysis methods currently available on the dashboard. Section 7 contains the summary of this work and conclusions.

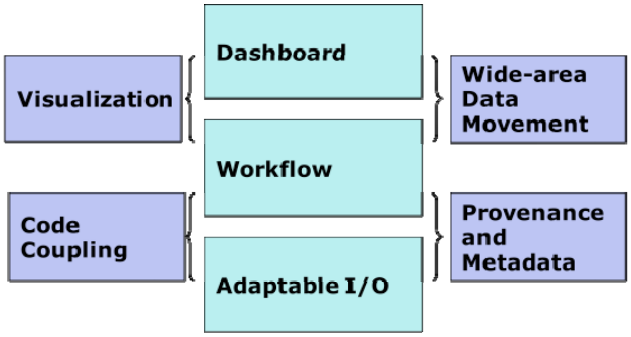

Figure 1. The Framework for Integrated End to End Scientific Data Management Technologies for Applications (FIESTA)

2. MOTIVATION

The supercomputers at ORNL, NERSC and ANL allow scientists to generate complex and massive output for extended periods of time; however, taking full advantage of the infrastructure is still a major challenge. The eSimMon dashboard, shown in Figure 2, is a user-centered tool developed specifically in an attempt to overcome obstacles encountered by simulation scientists. It's continued development requires an understanding of potential users and answering two essential questions: who are the users and how do they currently work? [5]

Typically, simulation scientists do not work alone: theoretical physicists need to work alongside computational scientists, I/O experts, performance experts, data management and workflow scientists, applied mathematicians and visualization experts in order to do their research. As computers increase in computational power, scientists are able to look at multi-physics, which requires codes to be coupled, since each code makes different assumptions and approximations. By coupling codes, scientists can explore larger ranges in the spatial and temporal domains, allowing for greater understanding of the science being examined. However, it also implies that experts in one code need to collaborate closely with experts in another code.

The eSimMon dashboard allows scientists to work together using one single tool for analysis, visualization, and data movement. Our key contribution is in using provenance information as our backbone, allowing researchers to talk about simulation variables, such as temperature, and not about files. Data movement tracking is a powerful feature of the dashboard. For example, we have seen cases where fusion scientists have seen performance graphs on the dashboard showing that their simulation was running poorly. The application scientist told a performance engineer to look at a particular run on the dashboard. By saving the entire system provenance, a code snapshot, and performance information, the engineer was able to compare that particular run to other runs, which led to finding the problem and fixing the code. This illustrates the point of effective collaboration: by saving enough information and having different people focusing on distinct aspects of the data, a group can work together, solving problems in different areas of expertise.

3. RELATED WORK

eSimMon allows for diverse international collaborations to occur easily, since we use the Adobe Flash framework for efficient Web 2.0 information exchange. For instance, the ITER [6] project is an international collaborative effort to establish fusion as an energy source for the future. Most frameworks related to the eSimMon and FIESTA at ORNL assume grid-based infrastructure which is a slightly more open environment than is used in DOE supercomputer centers. The ESMF [7] is similar framework for earth sciences applications. In our case, collaborators do not necessarily have direct access to the supercomputers, yet most of them need access to parts or all of colleagues' data to contribute to the project. Such access is granted through our web portal.

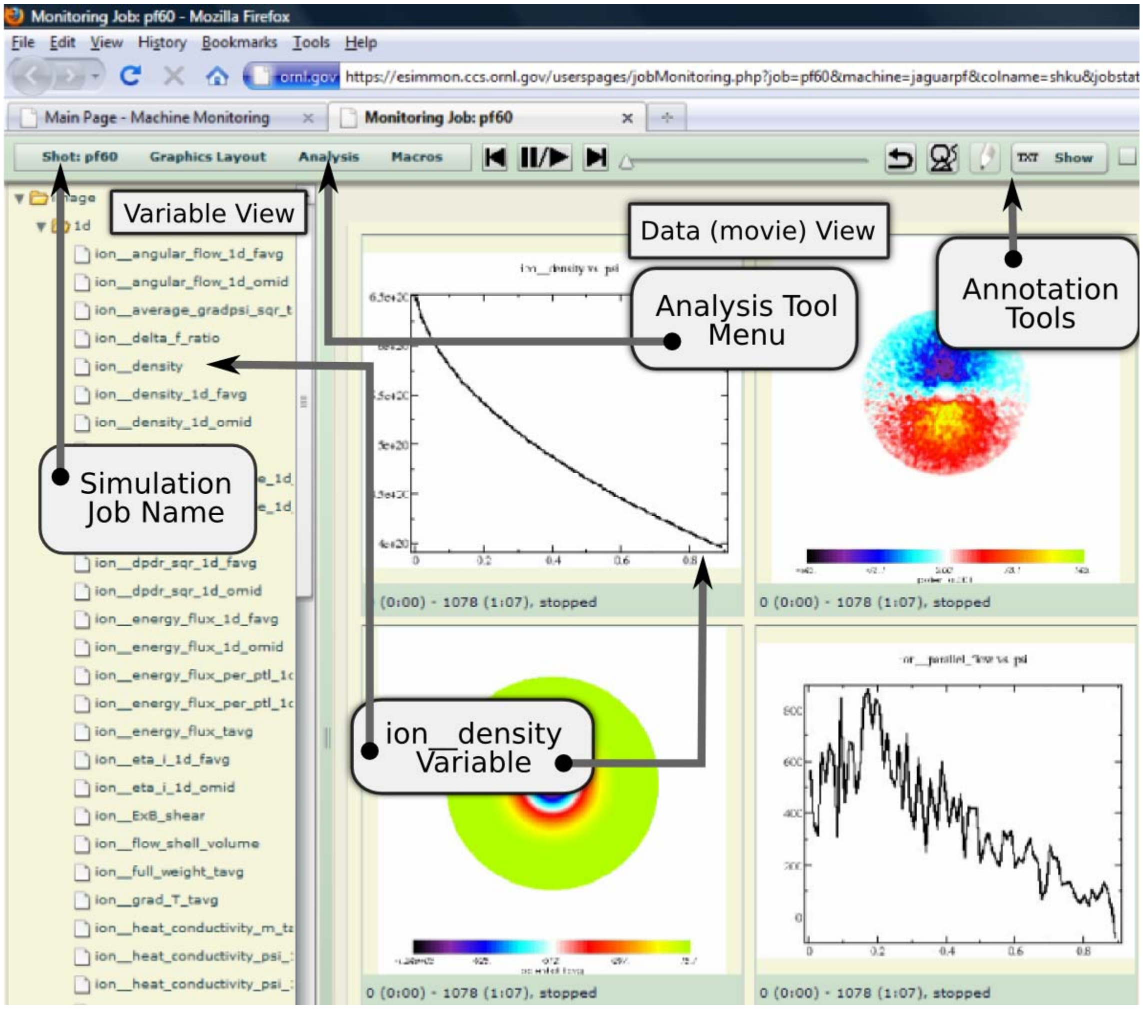
Monitoring Job: pf60 - Mozilla Firefox
File Edit View History Bookmarks Tools Help
ornl.gov https://esimmon.ccs.ornl.gov/userspages/jobMonitoring.php?job=pf60&machine=jaguarpf&colname=shku&jobstat
Main Page - Machine Monitoring
Monitoring Job: pf60
Shot: pf60
Graphics Layout
Analysis
Macros
TXT
Show
Variable View
Data (movie) View
Analysis Tool Menu
Annotation Tools
Simulation Job Name
ion__density Variable
ion__angular_flow_1d_favg
ion__angular_flow_1d_omid
ion__average_gradpsi_sqr_t
ion__delta_f_ratio
ion__density
ion__density_1d_favg
ion__density_1d_omid
ion__dpdr_sqr_1d_favg
ion__dpdr_sqr_1d_omid
ion__energy_flux_1d_favg
ion__energy_flux_1d_omid
ion__energy_flux_per_ptl_1c
ion__energy_flux_per_ptl_1c
ion__energy_flux_tavg
ion__eta_i_1d_favg
ion__eta_i_1d_omid
ion__ExB_shear
ion__flow_shell_volume
ion__full_weight_tavg
ion__grad_T_tavg
ion__heat_conductivity_m_t
ion__heat_conductivity_psi_
ion__heat_conductivity_psi_
(0:00) - 1078 (1:07), stopped
0 (0:00) - 1078 (1:07), stopped
0 (0:00) - 1078 (1:07), stopped
0 (0:00) - 1078 (1:07), stopped

Figure 2. The eSimMon Dashboard

Another framework for the dissemination of scientific data is the Earth System Grid (ESG) [8]. ESG was established in 2000 to advance climate science by developing computational resources for accessing and managing model data physically located in distributed multi-platform archives. Users are given a file-based (NetCDF) view of the data, and scientists around the world are allowed to download climate community datasets. A similar aspect of eSimMon is that scientists want to process data where the data lives, and then move the reduced data to local resources. ESG is working on capabilities within their web portal to provide more powerful analysis and visualization capabilities.

The Cactus framework, originally designed for relativity simulations, has also emerged as a modular, portable programming environment for High Performance Computing [9, 10]. The Cactus team provides scientists with a complete environment. Depending on the applications, users may or may not need to put some effort into adapting to a new work environment. Cactus accomplishes remote visualization by streaming data using HTTP connections. This is not always scalable, as we found out with eSimMon. We are moving towards using the Flash Media Server for better scalability and performance when streaming our Flash video format movies.

There are powerful sophisticated software tools available to simulation scientists for visualization and analysis, such as ParaView [11], VisTrails [12], VisIt [13], VTK [14] and AVS [15]. However, it's important to note that these require a level of expertise that is not trivial to acquire. Physicists usually learn one tool and the try to use it for all their work. This makes it difficult for them to collaborate given that other scientists may be used to another software package, and there is not currently a way to combine or integrate most of these analysis and visualization tools. A key feature of the eSimMon dashboard is that our users do not have to be experts in any one tool; the work is done for them at the back-end, using a collection of available software. All they have to do is to describe which analysis they want to apply to their data from our user-friendly interface.

The most prominent similar attempt to create a collaborative visualization portal is Many Eyes [16] by IBM. It shows the advantages in providing users with tools to share analysis methods and datasets on a simple, accessible Web environment. The main differences with the eSimMon are: the size of datasets and the types of complex analysis users apply. While we like the approach and believe that creating this type of scientific social network is the new, more flexible approach to collaborative visualization, our users would not be able to upload their large datasets, nor would they be able to upload their various, diverse customized analysis routines in Many Eyes. By having an analysis cluster close the users' data and enabling them to upload their analysis written in their favorite language, we intend to add a level of flexibility to the collaboration.

4. USER FEEDBACK

The composition of the team already gives clues as to how these scientists work. Members of the computer science team try to assist the computational scientists by providing powerful I/O, data management, data archiving, analytics and other tools. Performance engineers focus on the performance data: which modules were loaded, what were the inputs, how many processors did the simulation run on etc. In contrast, the visualization experts work on methods to show more details through graphics results. Meanwhile, analytics specialists may want to improve errors on these same results. Concurrently, experimentalists attempt to validate the data by comparing results to experimental data. From this insight and supporting scenarios, another key requirement of the dashboard beyond connecting users, is to provide a single point of access to all of available tools, codes, and data.

Simulation scientists perform the following task when working on large-scale high performance computers: 1) submit a job to a supercomputer, 2) monitor the status of the simulation, 3) list the output files, 4) visualize some of the initial results, 5) wait for the job to finish and start analyzing the output data. Sometimes users write their own customized analysis routines, other times they simply rely on the experts in visualization and analytics. They also may compare data from one simulation output to another. This sharing includes the comparison of the source code, the input parameters, the output variables, and derived products from the analysis of this data. Users might also want to compare the performance data from one simulation to another.

There are other implied requirements that emerge from profiles of eSimMon users. Because users come from different backgrounds and sometimes specialize into distinct domains within the same general science field the eSimMon dashboard should not be too detailed at its top level. Users should be able easily get to different tools or data sources and the dashboard should make these maneuvers seem effortless by doing the heavy lifting in the background. Ease of use is a must when dealing with scientists who routinely use a pencil and paper approach to solving problems and who are not necessarily computer savvy.

We have created a direct feedback system, where users send us notes/emails when there is a problem or request for the dashboard. We promptly answer the request and try to focus on working with the user community to aid the application scientist research. Our design method is to prototype a system, then get the initial feedback from the fusion community. We follow a continuous 360 design model, where the dashboard features evolve with submitted comments and requests. Some feedback reinforces our initial assumptions, while other comments provide new insights into the way scientists work and what they expect. One S3D researcher mentioned that his international collaborators found web access to his simulation runs very appealing and was anxious to be able to do more than simply viewing his results [17]. We also have the case of an XGC [18] user and his supervisor: the latter can easily check the former's results on the dashboard. A potential fusion scientist tells us he will use the software only if it incorporates a complete list of tools to run tests he routinely runs on his results. Some current dashboard features were directly requested from XGC users. For instance the ability to search through the occasionally lengthy list of variables was added upon such request. The feedback usually amounts to delivering simple and effective methods to monitor and analyze simulation runs. Indeed, we are dealing with the type of users who will often give a single try to new software but are eager to go back to their own ways of doing things even when these can be much less efficient. They want something that is easy to learn and that simply works!

Positive feedback and constructive criticism are necessary in order to produce good software. Choosing the right beta users is also important in order to avoid having some users abandon the system before it is in production. Our preliminary investigation and users comments are combined and summarized into two simple facts: they desire a collaborative tool for themselves and their team members as well as a tool that provides access to a range of tasks that they routinely execute. They want an easy way to do all of the above and stay as close as possible to their habitual ways of working which is what we undertake to provide in the eSimMon dashboard.

5. OVERVIEW OF FIESTA

As previously mentioned, eSimMon itself is a front-end to a more involved framework whose goal is to help scientists run, debug, monitor and analyze their simulations from any location through a web browser. The system we designed involves a tight integration of all of the components used in FIESTA.

Central to our work is the provenance capturing system. By saving all types of provenance information, including: 1) system, 2) workflow, 3) process, and 4) data provenance, we can allow scientists to interact more easily with a large simulation ran on a supercomputer. System information includes all of the information about where the simulation and analysis routines were compiled, and ran on; including all of the libraries they were built from. The workflow provenance information indicates which components ran at different times. This information can be used to debug workflows. Process provenance information is a vital tool for profiling simulation performance. Finally, data provenance tracks data history, starting from ADIOS [3], which captures provenance information about data being produced by the simulation, and continues to trace the data during analysis. While provenance tracking is hidden from users, it is the key to collaboration on the dashboard. It is by consistently recording and tracking data lineage and user activity that we can make accurate connections between groups of scientists and their data.

The workflow coordinates and executes all the mundane simulation tasks (moving files, converting formats, producing images and movies etc.), recording its every step in the “Data Store” accessible from the dashboard: it is the eSimMon “data feeder”.

The dashboard homepage is an overview of ORNL and NERSC supercomputer status. Users can quickly determine which systems are up or down, which are busy and where they would like to launch a job. Users can also view the status of their jobs and of their collaborators’ jobs. A significant feature is the consistent listing of past jobs and whether they are still on disk or on tape (HPSS archive); users can always go back and explore past jobs. From the home page they can select a particular job and view the data generated by this simulation. This view presents users with a tree view of variable names which they can drag-and-drop on the main canvas to visualize (see Figure 2). When a job is running, the user can view images as they are being generated. When a job is finished, these images, which represent physical variables at different time steps of the simulation, are encoded into a movie. Using the workflow provenance recording component of the workflow and its integrated provenance tracking system [19], the dashboard can straightforwardly trace images and movies back to original raw data files used to generate them. This functionality can be used for downloads or for analysis. More importantly from a user’s point of view, it can be used to avoid the intricacies of dealing with directories, files, file formats etc. When looking at lists of variable names and 1D, 2D or 3D graphics they can directly think of the sciences. In the simulation monitoring view, users can animate pre-processed movies; they can scribble notes on the simulation or directly on the graphics, then new questions arise: what else can they do? Can they interact more with the data and create their own analysis, their own graphics? To meet this need we have been integrating more analysis into the dashboard. Our analytics tools are either built-in or hook into server-side analysis/visualization software, including 3D visualization modules.

6. ANALYSIS

Our newest focus of eSimMon is to go beyond monitoring the viewing the initial simulation to emphasize visualization and analytics. Attempts to solve this problem include building in tools such as the eSimMon Calculator and interfaces to external tools, specifically the statistical analysis software package R, in order to help scientists perform basic operations on simulation variables.

The first questions users ask when they see outputs generated by the workflow on eSimMon are usually: “How do you generate these plots?” and “Can I generate my own?” Performing analysis on massive amount of data is not a small undertaking. Often times it involves moving the data across to an analysis cluster or moving the analysis closer to the High Performance Computing environment, converting file formats, extracting slices of data and plotting variables at different time steps etc. In other words, it involves series of cumbersome steps that the workflow routinely executes to output preprocessed dashboard movies. Therefore we developed a couple of tools to put our plotting utility at the disposition of the scientists to generate their own movies. The eSimMon Calculator is one way to do so. Users on the dashboard initially look at pre-generated movies from plots of variables over time (Figure 2); the Calculator allows users to generate their own plots and movies, for example by adding two variables. New movies are generated simply by selecting variables and operators and requesting a new movie. The plotting utility is called on the server to create new output. Similarly, if users want to adjust the axis ranges on the original plots, a rescaling widget is used to generate a slightly different movie of the same variable over time.

Our observations of scientists' work habits led us to the following conclusion: application scientists a priori do not care what tool is used; they may have learned one software and do not want to "waste" time learning another; they are not visualization experts. If a user wants to compute an FFT for instance, as long as they can specify input parameters and options, whether it's done in R or Matlab or another software package is not their primary concern. Therefore our approach is to provide access to a few software tools that perform a number of common analysis routines. We started with R: on the front-end, users select variables and statistical functions such as the Maximum, Minimum or Mean, and the back-end generates a new movie on demand.

Feedback from these two approaches has increased our understanding of scientists and has given more insight into how they operate. The latest emphasis on the dashboard is to address our findings about their work. First, it is not unusual for researchers not to know exactly what regions or sections of data they would like to focus on before the simulation runs. For example, the S3D team performs analysis and tests to narrow their area of focus. These steps vary from simulation to simulation and are not predictable. They cannot always be automated; neither do they need to be recorded. This leads to the first conclusion that the dashboard should provide a certain level of interactivity with the data before further analysis can be done. Second, when the time comes to do more in-depth data analysis, generic software tools are not always suitable for the needs of the scientists, who often have customized tools that they or someone in their team of collaborators use on a regular basis. Hence, there is a need to securely and conveniently exchange and run customized users' analysis scripts or programs. Third, we need to develop analysis workflows to allow users to better collectively explore their simulation results.

6.1. Interaction with the Data

Currently, there are two types of interactive GUI on the eSimMon dashboard: vector graphics popup windows for 2D variables and the contour tree widget for 3D variables.

6.1.1. Vector Graphics

Physicists, specifically fusion scientists in the following case examine a considerable number of 2D trends. In XGC, the most common plots show the evolution of one physical quantity over time. Before applying complex analysis on the data, users want to zoom in and out, pan, and vary x-y axes or time ranges. They may overlay or compare associated variables by placing them on the same graph. All of these intermediate tasks do not automatically need to be recorded. As one scientist put it, these maneuvers are futile unless the software is able to capture the thought process of the actor at the time; they simply guide scientists to the next task, to the next specific operation that leads to useful and/or publishable data. Only then is it useful to record activity provenance and users' notes for later use. Figure 3 shows a vector graphics cell on eSimMon. Another useful vector graphics feature the dashboard provides is the ability to place several variables side by side on a canvas and vary the axis in all sub-windows at the same time. An alternative vector graphics widget allows users to add rows and columns on demand, drag and drop more variables into newly added cells and overlay two variables within one cell. Users have the option to show or hide titles, legends and axis labels. Other features include formatting plot line colors, symbols and widths.

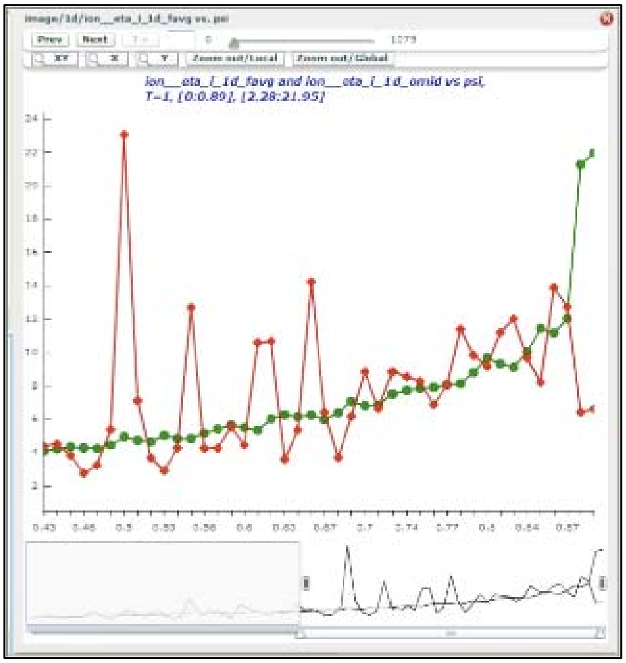

Figure 3. Single Cell Vector Graphics Window

Once again scientists are oblivious to what is done on the server. Flex (development framework for building Flash applications) native support for vector graphics and powerful back-end tools that extract XML data from large data files make the process on the front-end seem as simple as moving an x-axis slider or dragging and dropping items on a designated canvas. The data is moved to the client side which makes the GUI fast and responsive to users' request.

6.1.2. Contour Tree Widget

Demand for online, interactive tools to explore and identify useful regions in 3D data has been growing among eSimMon users. To meet this demand, we introduced the Contour Tree widget (Figure 4), which allows users to explore 3D simulation variables efficiently in simply connected volumes with scalar data.

The widget has three main components (Figure 4): the 3D View, the Contour Tree View and the Complexity Slider.

The 3D View shows a Flash movie of precomputed views from multiple camera angles of an isosurface set extracted from a simulation variable that is navigable by clicking and dragging the mouse. The Contour Tree View displays the contour tree (usually a simplified version of the complete contour tree) of the current dataset.

A contour is a connected component of the scalar field's level set. A contour tree is a Reeb graph without loops [20]. The contour tree summarizes the topological structure of a scalar field and can be used to identify useful features. To construct the contour tree data structure, we extract the isosurfaces, or level set of the dataset's scalar field. Each isosurface may contain multiple components. The contour tree is an index to the components in the scalar field the tree has a node corresponding to each contour, where the node's y coordinate corresponds to its value in the scalar field [20]. Branching in the tree corresponds to critical points in the scalar field. The simplified tree shows only the most significant features in the scalar field.

The three horizontal lines, colored blue, green and yellow in the Contour Tree View, correspond to default isovalues of the three isosurfaces displayed in the 3D View rendered in the same colors. Each line crosses every tree arc at most once. Selecting an arc in the tree highlights in magenta the component in the isosurface corresponding to the median value in the arc. The S3D turbulent flow temperature dataset, which was our initial case study and is shown in Figure 4 contains six components in the blue isosurface, three components in the green isosurface and thirteen components in the yellow isosurface in Figure 4; these are exactly the number of crossings of the colored horizontal lines with the contour tree in the Contour Tree View [20].

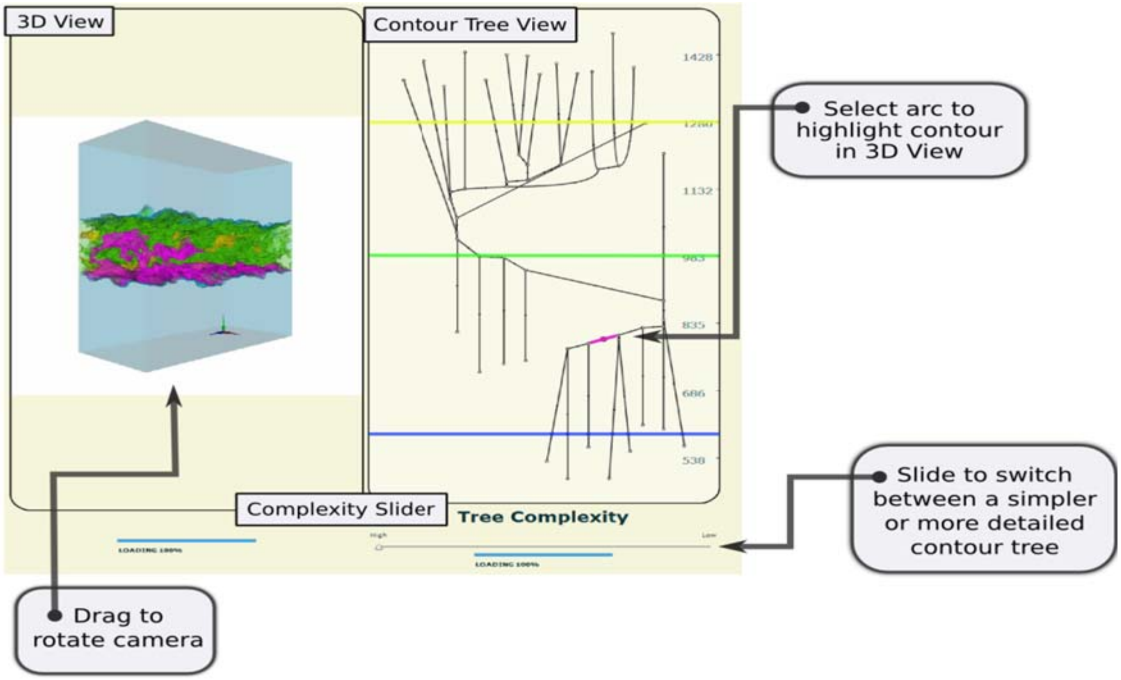

Figure 4. Detailed View of the Contour Tree Widget

Simple algorithms can be used to compute multi-scale representations of the contour tree allowing for the creation of a hierarchy of contour trees, where the small details are progressively removed and the major features are progressively highlighted. The Complexity Slider allows the user to choose the contour tree's level of complexity and therefore the size and number of features in the displayed isosurfaces [20].

6.2. Running User-Customized Analysis

6.2.1. GSI-SSH Infrastructure for Analysis

At ORNL, the security infrastructure requires authentication of the user with One Time Password (OTP) using an RSA SecurID each time the user accesses a resource at ORNL. This prohibits creating services that could submit jobs on behalf of a user. With help from security and system administration personnel, we designed and installed a secondary infrastructure based on the GSI-SSH and MyProxy servers that allows user authentication with grid certificates within ORNL, allowing eSimMon, a service inside ORNL, to access machines and run jobs on behalf of the user.

The key components of the infrastructure are shown in Figure 5: the GSI-SSH servers installed on Jaguar supercomputer and on the Ewok cluster, the modified MyProxy server on a dedicated machine, the Globus software installed on each machine and the user's grid certificates. The GSI-SSH server is a modified OpenSSH server that handles authentication with grid certificates and is part of the Globus package. These servers are running as secondary SSH servers on Ewok and Jaguar and are accessible only from selected machines inside ORNL (including the machine running eSimMon).

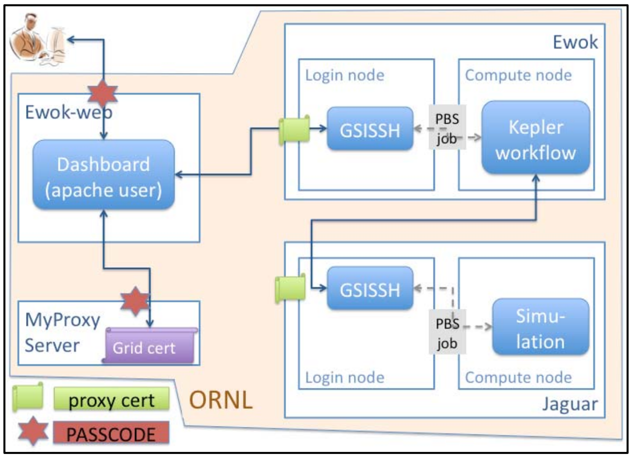

Figure 5. GSI Infrastructure at ORNL

A MyProxy server primarily stores proxy certificates created from user's certificates available from the server. The requester must authenticate with a pass phrase defined by the user when uploading the proxy certificate to the server. The MyProxy server at ORNL is modified to require the OTP authentication of the user when downloading the proxy certificate. This ensures that only the user can download it. Moreover, the MyProxy server extends the proxy certificate with special attributes; and the GSI-SSH servers accept only certificates with these attributes. This ensures that no other proxy certificates of undefined origin can be used for authentication.

The dashboard can submit an analysis job on the Ewok cluster on behalf of a user in three steps. First, the dashboard has to upload a primary proxy certificate to the MyProxy server. The grid certificate is encoded with a pass phrase that must be entered by the user, which should only be necessary once a year when a grid certificate is renewed. Second, the dashboard downloads a proxy certificate with the extended attributes from the MyProxy server which requires that the user authenticates again using the OTP. Finally, the dashboard can connect to Ewok and authenticate with the downloaded proxy, this time without the need for the presence of the user. Jobs can be submitted and checked on as long as the proxy certificate is valid (typically for a couple of hours).

6.2.2. Upload/Run Custom Analysis.

Using the infrastructure described in the previous section, it is possible to run a real, pre-prepared Matlab job from eSimMon. However a developer's assessment is not a user's evaluation. The difficulty lies in transforming a user-customized routine into a ready to use routine for collaborators. First we must get the user to conveniently and accurately describe his or her routine, the input parameters, and the expected output. Without these crucial steps, effective sharing analysis between collaborators is unlikely. The script upload interface should be easy to use to avoid discouraging users from sharing their code, yet the dashboard should ask enough questions to be able to later appropriately run the script for the owner or the owner's collaborators.

The first GUI prototype was developed using XGC Matlab analysis routines altered to eliminate hard coded input variables and automatic popup results. Instead, input parameters are passed to the script from the command line. Some constraints are put on the output as well. Results should consist of a single image, a series of images that can be encoded into a single movie for consistency or ASCII text. Based on these assumptions, the Upload Widget inquires about the number of inputs, and then accordingly creates fields to be filled by the user such as input names and types. The dashboard also encourages users to enter a description of the script they are uploading as guidance for their collaborators. This information can be retrieved anytime another user wishes to run an analysis routine. User names, analysis routines, input parameters and path to results are recorded on the back-end each time the widget is used. Since the imposed output types are already use throughout the dashboard, users can use the same methods to display data as they do for preprocessed workflow movies, images and text files.

6.2.3. Running Analysis Workflows

Running customized analysis routines from the dashboard can prove to be even more complex than running a single program and getting its output. Scientists often run their analytics in series of sub-tasks; reconstructing the steps that produce final post-processing simulation results is often equivalent to building an analysis and visualization pipeline. There are many research and implementation challenges involved in correctly building such pipelines. The Common Component Architecture (CCA) [21] promotes collaboration of independently developed software by moving component-based software engineering (CBSE) and creating a plug-and-play environment for high performance computing. Advanced Visual Systems (AVS) commercial software allows users to build a sequence of components (modules) to create a visualization pipeline [15]. VisTrails [12] is a scientific workflow management system with provenance support that also allows the creation of visualization and other, more general pipelines, in a similar manner. The Kepler [22] framework is another workflow system that captures the process from raw data to publishable simulation results. These examples require a considerable amount of work to carefully describe pipeline module or component inputs and outputs. We took a similar approach via web browser, which eliminates the need to set up a server-client environment and the need for running X windows or other setup for remote visualization. Furthermore, we propose to go further in this context by having a metadata-rich file format and data store in order to filter input parameters. We want to prevent processing corrupt data specifically for collaborators who are somewhat removed from the theoretical sciences. We can have the application scientists set temporal and special limits for variables, in other words, describe a reasonable and valid value. Using this information and the statistical data stored in the footer of our BP [3] file format, we can filter out invalid inputs for a particular module. This allows any user, including those who are not physics experts such as experimentalists and visualization specialists, to generate compelling results.

7. CONCLUSION

Our first version of the eSimMon dashboard had been used in concert with a workflow-provenance system to monitor simulations running on supercomputers. As scientists started using eSimMon, the need for more collaborative analytics tools became apparent and is now our main focus. Researchers can perform basic analysis on their data using built-in tools or external software while focusing solely on science as the dashboard hides IT overhead. In this paper, we have introduced new ways to provide users with swift access and efficient interactivity with their data in an environment that facilitates sharing and exchanges of analysis results and routines.

We currently have results from interactive front-end and computational back-end analyses. Further exploration of using one versus the other and the difference in performance will be an interesting challenge. We are also working toward a modularized, collaborative plug-and-play environment for running more complex analysis workflows from the dashboard. The opportunities for expansion are numerous. We have yet to add “live” collaboration and improve scalability, which will become more important as the number of users increases. Our main goal is to constantly investigate ways to improve the user experience on the dashboard and to facilitate collaboration between scientists, whose field of expertise, customized analysis scripts, preferred methods, and familiar tools prevent them from efficiently sharing visualization and analysis. Balancing enhancements and need for production ready software often leads to simplifications and custom-fit analysis. We need to keep exploring a more general and easily modifiable model to accommodate more codes.

ACKNOWLEDGEMENTS

The research and development of eSimMon has been supported in part by the National Center for Computational Sciences (NCCS) at the Oak Ridge National Laboratory, by DOE SCIDAC-2: Scientific Data Management Center for Enabling Technologies (CET) grants DE-FC02-01ER25486 and DE-FC02-ER25809, and by SciDAC Fusion Simulation Prototype (FSP) Center for Plasma Edge Simulation (CPES).
We wish to thank our colleagues in the SDM center, and in particular, Ilkay Altintas (SDSC), Paul Breimyer NCSU, Dan Crawl (SDSC), Ayla Khan (Utah), Guruprasad Kora (ORNL), Seung Hoe Ku (NYU), Mei Nagappan (NCSU), Nagiza Samatova (NCSU), Emanuele Santos (Utah), Arie Shoshani (LBL), Claudio Silva (Utah), and Alex Sim (LBL).

REFERENCES

[1] Z Lin, Y Xiao, I Holod, W Zhang, W Deng, S Klasky, J Lofstead, C Kamath, and N Wichmann, “Advanced simulation of electron heat transport in fusion plasmas,” *Journal of Physics: Conference Series*, Vol. 180, No. 1, pp.

012059-012070, 2009.

[2] R. Barreto, S. Klasky, N. Podhhorszki, P. Mouallem and M. Vouk, "Collaboration Portal for Petascale Simulations," 2009 International Symposium on Collaborative Technologies and Systems, (CTS 2009), pp. 384-393, Baltimore, Maryland, USA, May 2009.

[3] J. Lofstead, F. Zheng, S. Klasky, and K. Schwan, "Adaptable, metadata rich IO methods for portable high performance IO," Proceedings of the 2009 IEEE International Symposium on Parallel & Distributed Processing, Washington, DC, pp. 1-10, 2009.

[4] B. Ludascher, I. Altintas, S. Bowers, J. Cummings, T. Critchlow, E. Deelman, D. D. Roure, J. Freire, C. Goble, M. Jones, S. Klasky, T. McPhillips, N. Podhorszki, C. Silva, I. Taylor, and M. Vouk, "Scientific process automation and workflow management," SCIENTIFIC DATA MANAGEMENT: CHALLENGES, EXISTING TECHNOLOGY, AND DEPLOYMENT, A. Shoshani and D. Rotem, Eds., Chapman & Hall/CRC, Boca Raton, FL, 2009.

[5] A. Cooper, R. Reimann and D. Cronin, ABOUT FACE 3: THE ESSENTIALS OF INTERACTION DESIGN, John Wiley & Sons, Inc., New York, NY, USA, 2007.

[6] International Tokamak Research/Engineering project, "ITER - the way to new energy," ITER [website], February 17, 2009, Available: http://www.iter.org.

[7] "Earth Science Modeling Framework (ESMF)," ESMF [website], February 24, 2010, Available: http://www.earthsystemmodeling.org.

[8] Williams, D.N., R. Ananthakrishnan, D.E. Bernholdt, S. Bharathi, D. Brown, M. Chen, A.L. Chervenak, L. Cinquini, R. Drach, I.T. Foster, P. Fox, D. Fraser, J. Garcia, S. Hankin, P. Jones, D.E. Middleton, J. Schwidder, R. Schweitzer, R. Schuler, A. Shoshani, F. Siebenlist, A. Sim, W.G. Strand, M. Su, and N. Wilhelmi, "The Earth System Grid: Enabling Access to Multi-Model Climate Simulation Data," *Bulletin of the American Meteorological Society* (in press), 2009.

[9] "Cactus Code - Welcome," Cactus Code [website], November 7, 2009, Available: http://www.cactuscode.org.

[10] T. Goodale, G. Allen, G. Lanfermann, J. Masso, T. Radke, E. Seidel, and J. Shalf, "The Cactus framework and toolkit: Design and applications," HIGH PERFORMANCE COMPUTING FOR COMPUTATIONAL SCIENCE — VECPAR 2002, pp. 15-36, Springer, Berlin/Heidelberg, Germany, 2003.

[11] Kitware, "ParaView - Open Source Visualization," ParaView [website], Available: http://www.paraview.org

[12] J. Freire, C. T. Silva, S. P. Callahan, E. Santos, C. E. Scheidegger, and H. Vo. "Managing Rapidly-Evolving Scientific Workflows," in Proceedings of the International Provenance and Annotation Workshop (IPAW), LNCS 4145, pp. 10–18, Chicago, IL, 2006.

[13] "VisIt Visualization Tool," LLNL [website], January 29, 2010, Available: https://wci.llnl.gov/codes/visit.

[14] Kitware, "VTK - The Visualization Toolkit," VTK [website], Available: http://www.vtk.org.

[15] Advanced Visual Systems Inc., "AVS: Advanced Visual Systems: Data Visualization," AVS Advanced Visual Systems [website], January 20, 2010, Available: http://www.avs.com.

[16] IBM, "Many Eyes," Many Eyes beta [website], February 24, 2010, Available: http://manyeyes.alphaworks.ibm.com/manyeyes.

[17] E. Hawkes, R. Sankaran, J. Sutherland and J. Chen, "Direct numerical simulation of turbulent combustion: fundamental insights towards predictive models," *Journal of Physics: Conference Series*, Vol. 16, No. 1, pp. 65–79, 2005.

[18] S. Ku, C. Chang, M. Adams, J. Cummings, F. Hinton, D. Keyes, S. Klasky, W. Lee, Z. Lin, S. Parker and the CPES team, "Gyrokinetic particle simulation of neoclassical transport in the pedestal/scrape-off region of a tokamak plasma," *Institute of physics Publishing Journal of Physics: Conference Series*, Vol. 46, pp. 87–91, 2006.

[19] P. Mouallem, M. Vouk, S. Klasky, N. Podhorszki, and R. Barreto, "Tracking Files Using the Kepler Provenance Framework," *Proceedings of the 21st International Conference on Scientific and Statistical Database Management*, Vol. 5566, pp. 273–282, 2009.

[20] E. Santos, J. Tierny, A. Khan, B. Grimm, L. Lins, J. Freire, V. Pascucci, C. Silva, S. Klasky, R. Barreto, N. Podhorszki, "Enabling Advanced Visualization Tools in a Web-Based Simulation Monitoring System", in IEEE International Conference on eScience 2009, Oxford, UK, pp. 358-365, December 2009.

[21] B. A. Allan, R. Armstrong, D. E. Bernholdt, F. Bertrand, K. Chiu, T. L. Dahlgren, K. Damevski, W. R. Elwasif, T. G. W. Epperly, M. Govindaraju, D. S. Katz, J. A. Kohl,M. Krishnan, G. Kumfert, J. W. Larson, S. Lefantzi, M. J. Lewis, A. D. Malony, L. C. McInnes, J. Nieplocha, B. Norris, S. G. Parker, J. Ray, S. Shende, T. L. Windus, and S. Zhou, "A Component Architecture for High-Performance Scientific Computing," *International Journal for High Performance Computing Applications*, Vol. 20, No. 2, pp. 163–202, 2006.

[22] I. Altintas, C. Berkley, E. Jaeger, M. Jones, B. Ludascher and S. Mock, "Kepler: an extensible system for design and execution of scientific workflows," *Proceedings of the 16th International Conference on Scientific and Statistical Database Management*, pp. 423-424, 2004.