GA–A24600

THE NATIONAL FUSION COLLABORATORY PROJECT: APPLYING GRID TECHNOLOGY FOR MAGNETIC FUSION RESEARCH

by
D.P. SCHISSEL, J.R. BURRUSS, E. FEIBUSH, S.M. FLANAGAN,
T.W. FREDIAN, M.J. GREENWALD, K. KEAHEY, S.A. KLASKY, K. LI,
D.C. McCUNE, M. PAPKA, Q. PENG, L. RANDERSON, A. SANDERSON,
J. STILLERMAN, M.R. THOMPSON, and G. WALLACE

FEBRUARY 2004

DISCLAIMER

GA–A24600

THE NATIONAL FUSION COLLABORATORY PROJECT: APPLYING GRID TECHNOLOGY FOR MAGNETIC FUSION RESEARCH

by
D.P. SCHISSEL, J.R. BURRUSS, E. FEIBUSH,* S.M. FLANAGAN,
T.W. FREDIAN,† M.J. GREENWALD,† K. KEAHEY,‡ S.A. KLASKY,* K. LI,*
D.C. McCUNE,* M. PAPKA,‡ Q. PENG, L. RANDERSON,* A. SANDERSON,#
J. STILLERMAN,† M.R. THOMPSON,±Δ and G. WALLACE*

This is a preprint of a paper to be presented at the Workshop on Case Studies on Grid Applications, March 13, 2004, Berlin, Germany, in conjunction with GGF10 and to be published in *The Proceedings.*

*Princeton Plasma Physics Laboratory, Princeton, New Jersey.
†Massachusetts Institute of Technology, Cambridge, Massachusetts.
‡Argonne National Laboratory, Argonne, Illinois.
#University of Utah, Salt Lake City, Utah.
±Lawrence Berkeley National Laboratory, Berkeley, California.

Work supported by the U.S. Department of Energy under Cooperative Agreement No. DE-FG02-90ER54084

GENERAL ATOMICS PROJECT 30106
FEBRUARY 2004

THE NATIONAL FUSION COLLABORATORY PROJECT: APPLYING GRID TECHNOLOGY FOR MAGNETIC FUSION RESEARCH

D.P. Schissel,[1] K. Keahey,[2] T. Araki,[3] J.R. Burruss, [1] E. Feibush,[4] S.M. Flanagan,[1]
I. Foster,[2] T.W. Fredian,[5] M.J. Greenwald,[5] S.A. Klasky,[4] T. Leggett,[2] K. Li,[6]
D.C. McCune,[4] P. Lane,[2] M.E. Papka,[2] Q. Peng,[1] L. Randerson,[4] A. Sanderson,[7]
J. Stillerman,[5] M.R. Thompson,[8] and G. Wallace[6]

[1]General Atomics, P.O. Box 85608, San Diego, California 92186-5608
email: schissel@fusion.gat.com, Phone: (858) 455-3387, Fax: (858) 455- 4156
[2]Argonne National Laboratory, Argonne, Illinois 60439-4844
[3]NEC Internet Systems Research Laboratories, Kanagawa 216-8555, Japan
[4]Princeton Plasma Physics Laboratory, Princeton, New Jersey 08543
[5]Massachusetts Institute of Technology, Cambridge, Massachusetts 02139-4307
[6]Princeton University, Princeton, New Jersey 08543
[7]University of Utah, Salt Lake City, Utah 84112
[8]Lawrence Berkeley National Laboratory, Berkeley, California 04720

The overall goal of the DOE SciDAC funded U.S. National Fusion Collaboratory Project is to improve the productivity of fusion sciences research via the development and deployment of advanced software tools that reduce technical barriers to collaboration and sharing on a national scale. Our vision is to make resources — data, computers along with analysis, simulation and visualization codes – widely and transparently available as network accessible services, enabling real-time multi-institutional collaboration on fusion experiments, and improving comparisons between experiments and theory. The project unites fusion and computer science researchers to develop and deploy a national Fusion Energy Sciences "Grid" (FusionGrid) that is a system for secure sharing of computation, visualization, and data resources over the Internet. Some early prototype FusionGrid services proved so successful they are now used in a production environment for everyday fusion research. The success of other prototype services has resulted in fusion research funds being used to purchase necessary hardware to create production services. Future work is planned to augment the available FusionGrid services, transition prototype services to production capability, and put in place the infrastructure to support a larger user base.

1. Introduction

Developing a reliable energy system that is economically and environmentally sustainable is the long-term goal of Fusion Energy Science (FES) research. As fusion experiments have increased in size and complexity, there has been a concurrent growth in the number and importance of collaborations among large groups at the experimental sites and smaller groups located nationwide. Teaming with the experimental community is a theoretical and simulation community whose efforts range from applied analysis of experimental data to fundamental theory (e.g., realistic nonlinear 3D plasma models). As a result of the highly collaborative nature of FES research, the community is facing new and unique challenges.

Deployment of the prototype and production FusionGrid infrastructure [1,2] has been made possible through work supported by the DOE SciDAC program and from base funding for FES research. Substantial progress has been made in deploying FusionGrid. The TRANSP code [3], used for time dependent analysis and simulation of tokamak plasmas, has been released as a remotely accessible FusionGrid computational service along with supporting infrastructure development (data storage, monitoring, user GUI) [4]. This FusionGrid service has been so successful that it has become the production system for TRANSP usage in the United States and is starting to be adopted internationally. The Access Grid (AG) [5] has been deployed and used at major fusion experiments in support of remote participation. Progress has also been made in shared applications and displays, with testing and user feedback helping sharpen requirements for a truly collaborative control room for fusion experiments.

To achieve a dramatic impact on the efficiency of experimental FES it is necessary to combine Grid computing with unique collaboration technologies such as AG and application sharing. The combination of these technologies into a unified scientific research environment is a demanding and unique challenge and is outlined in the paper that follows.

2. FusionGrid: Requirements and Challenges

The vision for FusionGrid is a service oriented architecture in which experimental and simulation data, computer codes, analysis routines, visualization tools, and remote collaboration tools are accessible as network services. In this model, an application service provider (ASP) provides and maintains codes as well as the resources on which those codes execute. This mode of operation frees clients from maintaining and updating software, and providers from porting and supporting it on a wide-range of platforms. In this environment, access to services is stressed rather than data or software portability. Consequently, FusionGrid is not focused on desktop computing (e.g. SETI@home) or distributed supercomputing scenarios that are sometimes used to motivate Grid computing [6], but simply on making the ASP paradigm effective for Grids.

The ultimate goal of FusionGrid is to allow scientists at geographically distributed sites to participate fully in experimental and computational activities, whether by code execution or personal participation, as if they were working at a common site. We call this vision the collaborative tokamak control room and expect that it will dramatically increase the productivity of experimental science.

The collaborative control room is centered on the cyclical nature of experimental fusion research (Fig. 1). In any given day, 25–35 plasma pulses are taken with approximately 10 to 20 minutes in between each ~10 second pulse. For every plasma pulse, up to 10,000 separate measurements versus time are acquired at sample rates from kHz to MHz, representing about a gigabyte of data. Throughout the experimental session, hardware/software plasma control adjustments are made as required by the experimental science. These adjustments are debated and discussed amongst the experimental team. Decisions for changes to the next pulse are informed by data analysis conducted within the roughly 20-minute between-pulse interval. This mode of operation places a large premium on rapid data analysis that can be assimilated in near-realtime by a geographically dispersed research team. To be fully functional, the collaborative control room requires (1) secured computational services that can be scheduled as required, (2) the ability to rapidly compare experimental data with simulation results, (3) a means to share individual results easily with the group by moving application windows to a shared display, and (4) the ability for remote scientists to be fully engaged in experimental operations through shared audio, video, and applications.

Data analysis to support experimental operations includes between pulse analysis of raw acquired data as well as the merging of numerous data sources for whole-device simulation of the experimental plasma. Results of more detailed, computationally demanding predictive simulations, carried out during the planning phase prior to the experiment are made available for comparison to the actual experimental results in real-time. FusionGrid based computational services in support of experimental operations have the potential to greatly enhance the quantity of data analysis and the quality of experimental science. In addition, this same FusionGrid architecture can support other more computational demanding services that are not candidates for between pulse data analysis. Ultimately, fusion scientists can use the Grid for all computational requirements.

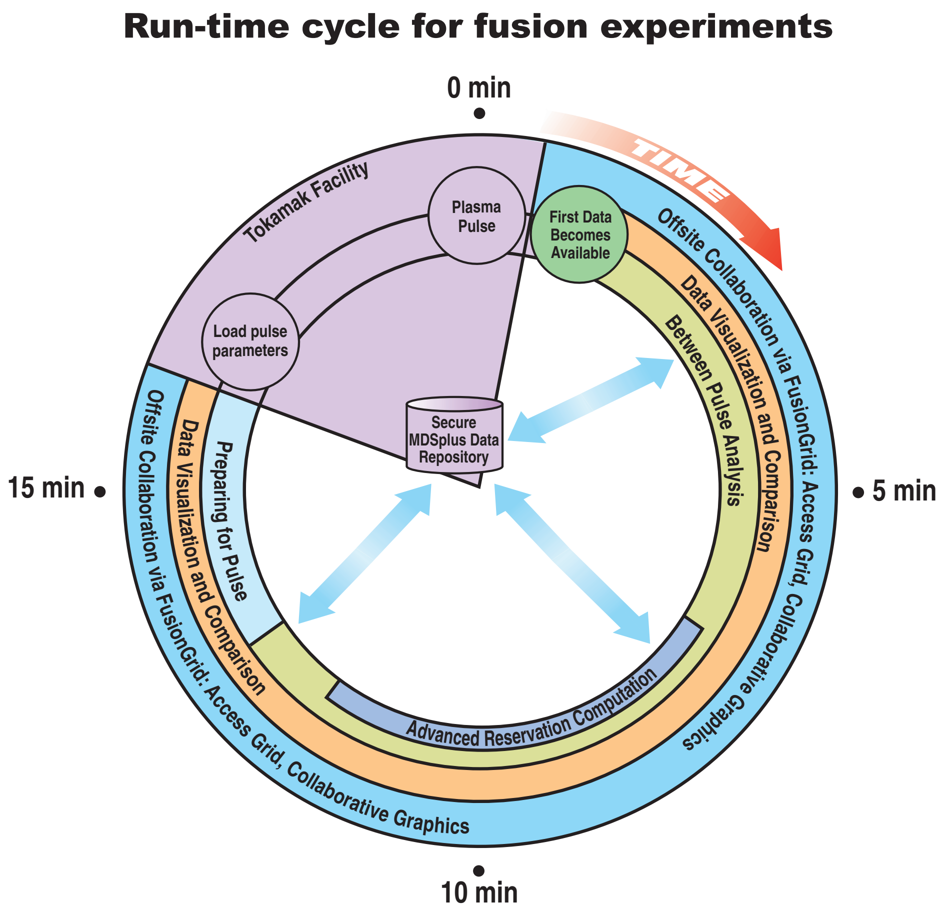

Fig. 1. Experimental magnetic fusion is cyclical in nature with fusion plasmas produced approximately every 20 minutes. Machine control is physically located onsite, but other work can be done off-site as appropriate. The collaborative control room will allow the integration of off-site tasks into experimental operations.

In addition to technical challenges, the most serious challenges to implementing FusionGrid on a large scale are issues of usability and deployment. In deployment, firewalls are currently our most serious issue. Firewalls and related security measures are typically not a matter of choice for the involved institutions. The introduction of firewalls in the fusion labs was mandated, following security violations, by DOE order 205.1. Utilizing GT-based services on a firewall-protected site requires the opening of a number of static ports, and, more significantly, an unpredictable number of dynamically chosen ephemeral ports created only for a specific session. Although some of these problems are resolved in GT3, some problems persist. In addition, users have found that the additional authentication required by firewalls with SecurID obviates the single sign-on advantage to Grid computing.

3. Grid Technologies in FusionGrid

To be successfully deployed, FusionGrid requires a broad range of components including underlying Grid tools and human-computer interfaces that makes its usage straightforward for fusion scientists. Grid tools needed include those supporting user and resource authentication; distributed authorization, resource management and enforcement tools (to give stakeholders control of real and intellectual property); secured shared computational services; shared applications and displays; and high-quality, flexible audio-visual applications.

3.1. Tools

FusionGrid creates a virtual organization comprising members of the U.S. fusion community, whose resources are protected by a shared security infrastructure based on authentication as well as authorization to allow fine-grain control of codes and resources. Authentication is based on X.509 identity certificates issued by the DOE Grids Certificate Authority (CA) as the global identifier for FusionGrid members. The Globus Toolkit (GT) [7] version 2 is employed by FusionGrid to provide secure authentication, remote job execution, and communication over the open Internet. Of particular benefit to the user community is the single sign-on capability since they need only log-on once, no matter how many services they use. GT is also used for starting remote computation and secure data transfer between fusion codes and databases.

To enable controlled resource sharing, an authorization callout was implemented in GT that enforces job creation as well as management decisions [8]. After evaluating a job creation request specified in GT's Resource Specification Language (RSL) in the context of authorization policies, an authorized user is allowed to create a job which can then be managed by another authorized user according to agreed-upon polices. These policies are expressed in and evaluated by the Akenti authorization system [9] called from GT [10]. A production version of the GT callout is currently being deployed on FusionGrid.

In order to both provide a better form of enforcement for authorization policies and alleviate the administrative burden on the administrators of fusion sites, we have developed a prototype dynamic account management system [11]. It allows a user authorized by its Grid identity to create a local account on a resource for a limited time, and later manage that account. Although this infrastructure has not yet been integrated with an authorization system and used in production, the prototype code for dynamic accounts is currently available.

In order to enable the between-pulse execution of analysis and simulation codes an agreement-based system has been developed to provide end-to-end agreements on between pulse code execution. This system combines resource reservations with application runtime and data transfer prediction to provide advance estimates of execution times and the level to which its completion can be guaranteed. This work has contributed to the development of WS-Agreement at the Global Grid Forum (GGF) and has been demonstrated at SC03. While GT version 2 is used in the production FusionGrid, prototype dynamic accounts and between pulse code execution mechanisms are being developed in GT3.

Data access on FusionGrid has been made available using the MDSPlus data acquisition and data management system [12], by far the most widely used and successful data acquisition and management system in the international fusion community. MDSplus provides a convenient meta-data representation and layer facilitating fusion data management. MDSplus communication has been layered on top of the Globus I/O library to take advantage of the Grid Security Infrastructure (GSI) and thus achieve secure data access on FusionGrid using the standard MDSplus interface without any loss in speed or functionality. Presently, the three main

MDSplus experimental data repositories at Alcator C–Mod, DIII–D, and NSTX are available securely on FusionGrid. Data management by MDSplus of large datasets generated by simulation codes is presently being tested. All access to relational databases, which are widely used in the fusion program and typically run on Microsoft Windows platforms, is currently done insecurely through MDSplus since a production version of GT for Windows is currently unavailable.

With multiple applications distributed throughout a Grid infrastructure, it becomes a challenge to monitor the progress and state of each application. Users of a Grid environment need to know the specific state of code runs, when their data results are available, or if the requested application is even available. To track and monitor applications on the FusionGrid, the FusionGrid Monitor (FGM) has been developed as a Java Servlet which can accept and monitor information from remote and distributed applications [13]. Currently, FGM tracks TRANSP analysis runs on the National Fusion Grid, and provides updated information on each individual run, including: current state, cpu time, wall time, comments, and access to log files that have been produced by the analysis. The Fusion Grid Monitoring system has built to provide user output through HTML, utilizing both server push and client pull capabilities. This allows multiple users to connect to FGM, view their code runs by using a web browser, and obtain updated information without excessive user input or client software. Designed in the Java language, the monitoring system is portable, and with the inclusion of the Java Expert System Shell (JESS), the system is also expandable and customizable. Online access to log files is available through FGM, utilizing anonymous FTP.

FusionGrid uses AG to create a service that enables group-to-group interaction and collaboration that improves the user experience beyond teleconferencing. AG provides scientific research with a complex multi-site collaborative experience integrated with high-end visualization environments including tiled display walls [14]. To meet the needs of the FES research community, the Personal Interface to the Grid (PIG) was developed offering a low cost alternative to a full-scale conference room size AG node. Currently, PIGs are installed at C–Mod, DIII–D, and a full-scale AG node at PPPL. Most importantly, a PIG has also been installed in the DIII–D Control Room and was used in a joint experiment between C–Mod and DIII–D. The initial test proved valuable in terms of the potential for such collaboration, and we plan to make such collaboration standard.

To enhance collaboration, in addition to Grid technologies we are also using visualization technology. Tiled display walls are being used to enhance the collaborative work environment of the tokamak control room (Fig. 2). Software tools and techniques, such as work on improving data distribution and scalability, are being developed to increase their ease–of–use in these environments. For example, remote collaboration between two geographically separated tiled walls with networking software has been developed as a prototype FusionGrid service, thus providing a shared collaborative display. We are also using the SCIRun Problem Solving Environment (PSE) for advanced scientific visualization. Such a capability allows for the exploration of complex simulation results and the comparison of simulation and experimental data within the collaborative control room.

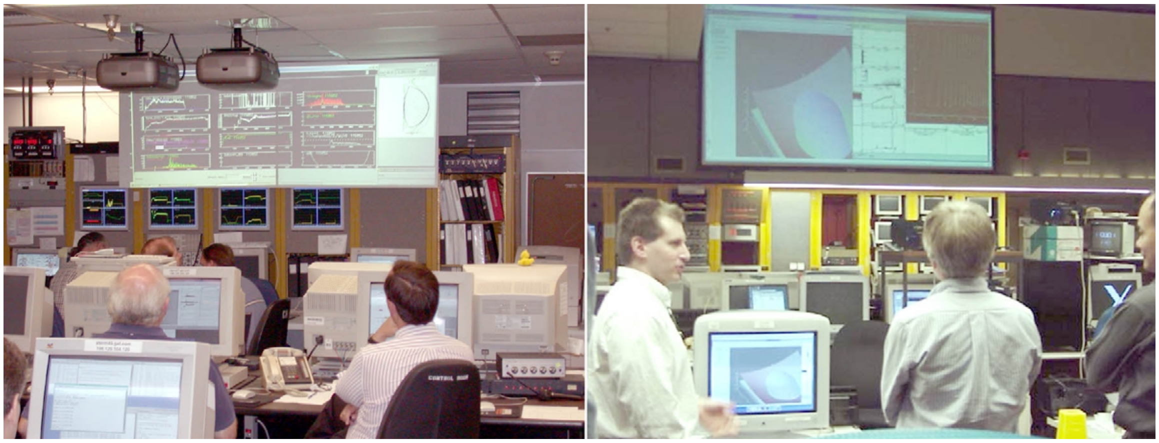

Fig. 2. The collaborative fusion control room has been tested during experimental operations. DIII–D (San Diego) is on the left and NSTX (Princeton) is on the right.

3.2. Interfaces

3.2.1. User Interfaces

Fusion scientists manually manage their identity certificates using the DOE Grids certificate management system. The NFC Project team and fusion scientists found the process to be very cumbersome especially for new users. Typically, the researcher requests a long-term certificate through their web browser, exports the PKCS#12 file, transfers the file from their desktop computer to a UNIX host, uses OpenSSL utilities to extract the private key and certificate from the PKCS#12 file, then installs those files in the expected directory under the correct permissions. This multi-step complex process is currently a serious usability issue and prevents widespread adoption of FusionGrid. Thus, we will explore the use of a myProxy server [15] to store the user's long-term credentials on a secure host. The standard grid-proxy-init command will be supplemented with a command that will retrieve a proxy certificate from the myProxy host. The long-term certificates will be password protected.

For ease of use, a GUI utility called PreTRANSP was created for input data preparation and TRANSP code innovation. PreTRANSP serves as the primary interface to Grid-enabled TRANSP and is written in the Interactive Data Language (IDL). PreTRANSP assigns unique run id numbers to new TRANSP runs, allows researchers to search through existing runs, loads inputs, sends messages to the monitoring system, and submits TRANSP jobs for remote execution using GT. All of these activities either happen behind the scenes or are controlled by the user through the PreTRANSP GUI. For example, the mechanism for submitting a TRANSP job for execution is a large, friendly start button.

TRANSP monitoring information is displayed through the Fusion Grid Monitor (FGM). The FGM uses server-push technology to update client web browsers when new monitoring information has been posted by TRANSP. TRANSP users can monitor the status of their code runs, view log files through an anonymous-FTP mechanism, and plot results as they are being calculated, all through the FGM. The log files are processed server-side. The plotting feature is handled through ElVis, a Java applet that runs client-side. We find this web-based interface to be very convenient.

Currently there is no user interface for editing and managing authorization policies governing access to TRANSP. FusionGrid has prototyped authorization using Akenti, but is currently not using this feature in production. Our current focus is on developing good policy management tools allowing a service administrator easy understanding of policies and quick reaction to policy changes. This work is of particular importance since we anticipate that authorization will become increasingly important as the number of services grows.

3.2.2. Software Interfaces

The PreTRANSP client described in the previous section relies exclusively on GT command-line tools such as globus-url-copy, and globus-job-submit to manage interactions in the Grid. Secure MDSplus, whose interactions are more tightly coupled, leverages the GT globus_io and globus_common modules. The result is that Secure MDSplus can securely transfer data with no loss of performance relative to normal MDSplus. We are awaiting the release of XIO features that will allow us to incorporate more advanced communication capabilities.

TRANSP calls secure MDSplus using the standard MDSplus C API routines such as mdsconnect and mdsvalue. TRANSP is linked against secure MDSplus, and by extension the GT shared libraries, at compile time. Again, this tight coupling results in efficient data transfer, a feature that is very important for large scientific codes.

Currently, SQL Server is accessed through MIT-written IDL procedures that utilize call_external. This connection is not encrypted and uses only the standard SQL Server security features. Since a production GT Windows release is not available, secure database connections are only possible through the use of an intermediate secure MDSplus server. This intermediary scheme was tested and works well enough, but performance is not as good because of the intermediate step. When the intermediate secure MDSplus server runs on the SQL Server host, memory allocations are greatly increased since — whether one uses client-side or server-side cursors — at least one cursor will be running on the server machine in the intermediary. If the intermediary runs elsewhere then the network hops are increased and the intermediary-SQL Server connections are left unencrypted.

3.3. Performance Considerations

Two performance considerations have become apparent with the initial deployment of FusionGrid. The first is the performance slowdown resulting from WAN latency of many data transactions per TRANSP computation. The input and output data for TRANSP is not large in total but there are on the order of 1000 different quantities to move over the WAN via MDSplus. For between pulse data processing and digesting of results by the experimental team, the slowdown was unacceptable. Instead, the individual quantities were packaged in one file for WAN transfer. The second is the performance slowdown associated with application sharing with remote users. Initial feedback from testing indicates that on-site personnel consider the present slowdown unacceptable. Work continues to quantify both of these performance issues and investigate methodologies for overcoming the limitations.

4. FusionGrid Applications

The code TRANSP, used for time-dependent analysis and simulation of tokamak plasmas, was released in late 2002 as a service on FusionGrid. Running on a Linux cluster at Princeton Plasma Physics Laboratory (PPPL), this FusionGrid service has performed over 1500 simulations taking over 10,000 CPU hours for 9 different experimental fusion devices (Fig. 3). Both the input data and output data of TRANSP is securely written and read from an MDSplus and SQL data repository. Monitoring and tracking of these runs has been accomplished through the deployment of the FusionGrid Monitoring System (FGM), a derivative of the Data Analysis Monitoring System used at DIII–D.

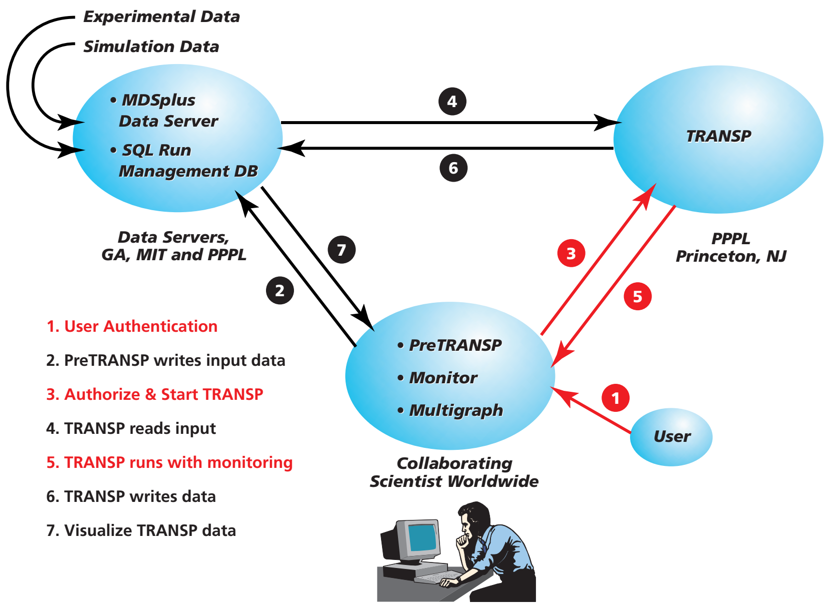

Fig. 3. The TRANSP FusionGrid service based at PPPL is securely available from anywhere in the world.

Deployment of TRANSP on FusionGrid frees scientists from the need to build and maintain local versions of a large and complex code, and also eases the burden on the code development team, which was previously required to provide support on a highly heterogeneous set of remote machines. Further, the Grid implementation has enabled users to perform significantly more calculations, resulting in more rapid progress than prior to the deployment of FusionGrid. Scientific results using FusionGrid were presented at the 2002 and 2003 American Physical Society Division of Plasma Physics (APS/DPP) meetings and the 2002 19th IAEA Fusion Energy Conference. The service proved extremely popular in our community with at present approximately 50 FusionGrid users with several based in Europe.

In a similar manner in which GT was combined with fusion science specific tools and interfaces to create the TRANSP service, AG has been combined with fusion specific displays and application sharing to create a service for remote experimental participation. To be effective, remote participation must include real time information to yield a sense of control room presence. This sense of presence is critical if off-site scientists are to participate effectively in an experiment and must include all real time data displayed to on-site staff. Sense of presence includes the ability to see and hear what people see and hear in the control room. Even more important to a remote collaborator is the information flow throughout the twenty-minute pulse cycle. This information includes the pulse cycle/status clock, pulse parameters, status of between-pulse analysis and announcements from the scientist in charge.

The FusionGrid requirement for researches to participate remotely in experimental operations has been tested with the desktop AG node combined with the ability to share applications. Additionally, real-time control room information is currently made available to remote participants through a fusion specific web interface. The tokamak control computer has the pulse cycle information and plasma control parameters. While it sends the information to the large

LED display in the control room, it also writes the same data directly to a web server that parses them into a reasonable format for web display. The web client checks with the web server periodically and updates the status accordingly. Integrated in the same display is a quick view of the data acquisition and analysis status. Whenever a group of data becomes available, the corresponding indicator changes color. The statuses are made available by the MDSplus event system that drives the analysis cycle. Therefore, the remote scientist can combine the AG node, shared application, and web display for a full sense of control room presence resulting in their complete integration into the decision making process of the experimental control room.

5. Lessons Learned

The real world usage and testing of FusionGrid services has highlighted several areas where additional work is required before large-scale adoption is undertaken in the fusion community. First and foremost, the management of certificates has to be made simpler and easier for both the service user and service provider. A researcher’s first experience with FusionGrid is obtaining and installing their X.509 certificate. Present difficulties must be resolved so scientists’ first impression will be positive.

The conflicting requirements of Grid computing and site security (firewalls) makes expansion of FusionGrid services a time consuming task and one that we cannot expect to scale well to the over 1000 U.S. based fusion scientists. In order to avoid excessive exposure by opening the firewall, site administrators will often open ports only to specific IP addresses (insufficient to support dynamic client access in the Grid), or use it in conjunction with other technologies, such as SecurID. The use of SecurID allows for more flexibility since a session is established based on a token and then the firewall is dynamically modified to open a set of ports for traffic coming from the IP that established the secure session. The ports that get opened are defined in the profile of the user (negotiated out of band) who established the secure session. This approach allows for more flexibility since the IP address does not have to be agreed on in advance. However, the SecurID solution has its own problems: the ports get opened for a fixed, preconfigured time only. After that time the secure session is terminated and can be reestablished only manually which results in termination of all existing connections and their state thereby terminating an existing Grid session. A better approach would be to allow for the user or application to manage the lifetime of the open ports, using for example a lease-based system. We hope to explore such an approach in future work.

A more serious problem is that different institutions may have different SecurID systems, which means that establishing connections to two institutions can require two SecurID cards, increasing complexity for the user. Clearly this approach of multiple independent SecurID systems does not scale for a national deployment of FusionGrid.

Efficient data management of large simulation datasets is required so that rapid comparison of simulation data to experimental data is possible in the tokamak control room. To accommodate large-scale simulations a new MDSplus server has been installed and configured at NERSC. Performance testing will start using the NIMROD code running on the Seaborg computer. These will include tests of MDSplus I/O performance comparing local LAN and WAN connections using a variety of transport levels. Comparisons with other data transport mechanisms, notably HDF5 with GridFTP will also be carried out.

Initial testing and usage of shared applications in the control room identified efficiency issues that must be resolved before the capability is adopted for the control room. Feedback from scientists in the control room indicated that when they began sharing with remote users, they themselves began to feel as if they were remotely located since their computer response became significantly slower. Giving a remote scientist a new capability at the expense of on-site staff is not acceptable.

Ambient audio in the fusion control room environment can reduce the effectiveness of the AG audio system. Isolating AG users with headsets is one way to overcome this problem. This approach has two advantages. First, if there is no ambient audio output, the echo cancellation hardware is not needed, which saves cost. Second, it provides each user with a dedicated audio channel delivered directly to them, removing distractions from ambient noise. The disadvantage to this approach is that it tethers the user to the desk and in the case of headphone-based systems inhibits the user's ability to interact with local collaborators. The possible of using Bluetooth headsets for providing individually delivered public audio channel has been proposed. This approach could form the basis for future work in private audio channels leading to the support of multiple audio channels within an AG environment.

The increasing user base of FusionGrid combined with the addition of new services will require better education, training, and documentation. Users, developers, and systems administrators will need access to crucial information in order to make best use of FusionGrid. This information has been provided mostly by word of mouth by project members but this solution does not scale for a membership on the order of 1000. To be successful, a User's Guide, a Developer's Guide, and an Administrator's Guide will need to be created for FusionGrid.

6. Conclusion

The main thrust of future work for FusionGrid is ease–of–use, including easier certificate management, easier and more uniform access to FusionGrid code services possibly via web portals, and better interplay between FusionGrid security and site security (firewalls). At the same time, new code services will be deployed as requested by the user community. For FusionGrid's advanced collaboration service the challenge is to develop a science of collaboration; to understand what works and why (human factors and social issues), what functionality is needed, and how should it be interfaced to the users as part of FusionGrid.

As Grid middleware evolves, there is a concern that trying to satisfy all possible Grid usage scenarios will result in exceedingly complex software. Simple software that quickly allows users and service providers to join FusionGrid will be seen as an advantage; the benefit of joining far outweighs the time required to join. Maintaining tightly coupled software, at all levels, that contains only what is required will be the best way to insure that FusionGrid evolves into a streamlined efficient system for the U.S. fusion community.

Acknowledgment

This work was supported by US Department of Energy SciDAC initiative and at General Atomics under DOE Cooperative Agreement No. DE-FG02-90ER54084 and at ANL by the Mathematical, Information and Computational Sciences Division Subprogram of the Office of Advanced Scientific Computing Research, SciDAC Program, Office of Science, U.S. Department of Energy, under Contract W-31-109-ENG-38.

References

[1] D.P. Schissel for the Collaboratory Team, "Building the U.S. National Fusion Grid: Results from the National Fusion Collaboratory Project," *4th IAEA Tech. Mtg. on Control, Data Acquisition, and Remote Participation for Fusion Research,* 2003, San Diego, California, to be published in Fusion Engineering and Design.

[2] K. Keahey, et al., *Computational Grids in Action: the National Fusion Collaboratory.* Future Generation Computing Systems (to appear), October 2002. **18**(8): p. 1005-1015.

[3] R.J. Hawryluk, "An Empirical Approach to Tokamak Transport," *Phys. Plasmas Close to Thermonuclear Conditions*, ed. by B. Coppi, et al., (CEC, Brussels, 1980), Vol. 1, p. 19.

[4] J. Burruss, et al., "Remote Computing on the National Fusion Grid," *4th IAEA Tech. Mtg. on Control, Data Acquisition, and Remote Participation for Fusion Research*, July 21-23, 2003, San Diego, California, to be published in Fusion Engineering and Design.

[5] R. Stevens, *Group-Oriented Collaboration: The Access Grid Collaboration System, in the Grid: Blueprint for a New Computing Infrastructure*, (2nd Edition), 2004, Morgan Kaufmann.

[6] I. Foster and C. Kesselman, "Globus: A Metacomputing Infrastructure Toolkit," International Journal of Supercomputing Applications, 1998. **11**(2) p. 115-129.

[7] I. Foster and C. Kesselman, *The Grid: Blueprint for a New Computing Infrastructure,* (2nd Edition), 2004, Morgan Kaufmann.

[8] K. Keahey, et al., "Fine-Grained Authorization for Job Execution in the Grid: Design and Implementatio," Concurrency and Computation: Practice and Experience (to appear), 2004.

[9] M.R. Thompson, et.al.,"Certificate-base Authorization in a PKI Environment," *ACM Transactions of Information and System Security (TISSEC),* (Nov. 2003) vol. 6.4 pp. 566-588.

[10] M.R. Thompson, et al., "Fine-grained Authorization for Job and Resource Management using Akenti and Globus," *2003 Conference on Computing in High Energy and Nuclear Physics*, 2003, La Jolla, California.

[11] K. Keahey, M. Ripeanu, and K. Doering, "Dynamic Creation and Management of Runtime Environments in the Grid," in *Workshop on Designing and Building Web Services (to appear)*. 2003. Chicago, IL.

[12] T. Fredian, J. Stillerman, "MDSplus Remote Collaboration Support–Internet and World Wide Web," Fusion Engineering and Design **43** (1999) 327; http://www.mdsplus.org/.

[13] S. Flanagan, et al., "A General Purpose Data Analysis Monitoring System with Case Studies from the National Fusion Grid and the DIII–D MDSplus between Pulse Analysis System*," 4th IAEA Tech. Mtg. on Control, Data Acquisition, and Remote Participation for Fusion Research*, 2003, San Diego, California, to be published in Fusion Engineering and Design.

[14] R. Stevens, M. E. Papka, and T. Disz, "Prototyping the Workspaces of the Future," *IEEE Internet Computing* **7**, pp. 51-58, 2003.

[15] J. Novotny, S. Tuecke, and V. Welch, "An Online Credential Repository for the Grid: MyProxy," in *10th IEEE International Symposium on High Performance Distributed Computing*, 2001, San Francisco: IEEE Computer Society Press.