13th World Conference on Earthquake Engineering
Vancouver, B.C., Canada
August 1-6, 2004
Paper No. 1674

NEESGRID: A DISTRIBUTED COLLABORATORY FOR ADVANCED EARTHQUAKE ENGINEERING EXPERIMENT AND SIMULATION

Billie SPENCER Jr.,[1] Thomas A. FINHOLT,[2] Ian FOSTER,[3] Carl KESSELMAN,[4] Cristina BELDICA,[5] Joe FUTRELLE,[6] Sridhar GULLAPALLI,[7] Paul HUBBARD,[8] Lee LIMING,[9] Doru MARCUSIU,[10] Laura PEARLMAN,[11] Charles SEVERANCE,[12] Guangqiang YANG[13]

---

[1] Nathan M. and Anne M. Newmark Endowed Chair in Civil Engineering and NCSA Senior Center Affiliate, Univ. of Illinois at Urbana-Champaign, Urbana, IL 61801, USA bfs@uiuc.edu

[2] Research Associate Professor, School of Information, Univ. of Michigan, Ann, MI 48109, USA finholt@umich.edu

[3] Associate Division Director and Senior Scientist, Mathematics and Computer Science Division, Argonne National Laboratory, Argonne, IL 60439, USA foster@mcs.anl.gov

[4] Information Sciences Institute Fellow, Information Sciences Institute, Univ. of Southern California, Marina del Way, CA 90292, USA carl@isi.edu

[5] Project Manager, National Center for Supercomputing Applications, Univ. of Illinois at Urbana-Champaign, Urbana, IL 61801, USA cbeldica@ncsa.uiuc.edu

[6] National Center for Supercomputing Applications, Univ. of Illinois at Urbana-Champaign, Urbana, IL 61801, USA, futrelle@ncsa.uiuc.edu

[7] Information Sciences Institute, University of Southern California, Marina del Way, CA 90292, USA, sridhar@isi.edu

[8] Mathematics and Computer Science Division, Argonne National Laboratory, Argonne, IL 60439, USA, hubbard@mcs.anl.gov

[9] Mathematics and Computer Science Division, Argonne National Laboratory, Argonne, IL 60439, USA, liming@mcs.anl.gov

[10] Assistant Director of Grid and Security Technologies, National Center for Supercomputing Applications, Univ. of Illinois at Urbana-Champaign, Urbana, IL 61801, USA marcusiu@ncsa.uiuc.edu

[11] Information Sciences Institute, Univ. of Southern California, Marina del Way, CA 90292, USA laura@isi.edu

[12] School of Information, Univ. of Michigan, Ann, MI 48109, USA csev@umich.edu

[13] Postdoctoral Research Associate, University of Illinois at Urbana-Champaign, Urbana, IL 61801, USA gyang2@uiuc.edu

SUMMARY

NEESgrid, the system integration component of the NEES project, links earthquake researchers across the U.S. with leading-edge computing resources and research equipment, allowing collaborative teams (including remote participants) to plan, perform, and publish their experiments. The integrated tools and components made available by NEESgrid enable earthquake engineering simulation – both physical and numerical – providing an environment for researchers to develop increasingly complex, comprehensive, and accurate models of how structures of all kinds respond to earthquake loadings. In this paper, various NEESgrid components are introduced and their roles in NEESgrid are identified. Subsequently, the Multi-Site Online Simulation Testbed (MOST) experiment, which was conducted on July 30, 2003 to showcase the NEESgrid software capabilities, will be discussed. The MOST experiment coupled two large-scale physical experiments in Illinois and Colorado with a computational simulation. This paper also describes the Mini-MOST experiment, which is a tool for NEESgrid education, training and outreach. The success of the MOST and Mini-MOST experiments demonstrates the significant potential of Grid technologies in engineering research and, more importantly, the experience gained offered us a better insight on how to build and deploy effective Grid applications in the future.

INTRODUCTION

NEESgrid, the system integration component of the NEES project, links earthquake researchers across the U.S. with leading-edge computing resources and research equipment, allowing collaborative teams (including remote participants) to plan, perform, and publish their experiments. The integrated tools made available by NEESgrid enable earthquake engineering simulation – both physically and numerically – providing an environment for researchers to develop increasingly complex, comprehensive, and accurate models of how structures of all kinds respond to earthquake loadings. This NEESgrid-based "collaboratory" allows researchers across the country gain remote, shared access to experimental equipment and data. It facilitates the communication of information efficiently from researcher to researcher, and it provides a powerful collaborative space for modeling and simulation. Through the NEESgrid, researchers will:

- perform teleobservation and teleoperation of experiments;
- publish to and make use of a curated data repository using standardized markup;
- access computational resources and open-source analytical tools;
- access collaborative tools for experiment planning, execution, analysis, and publication.

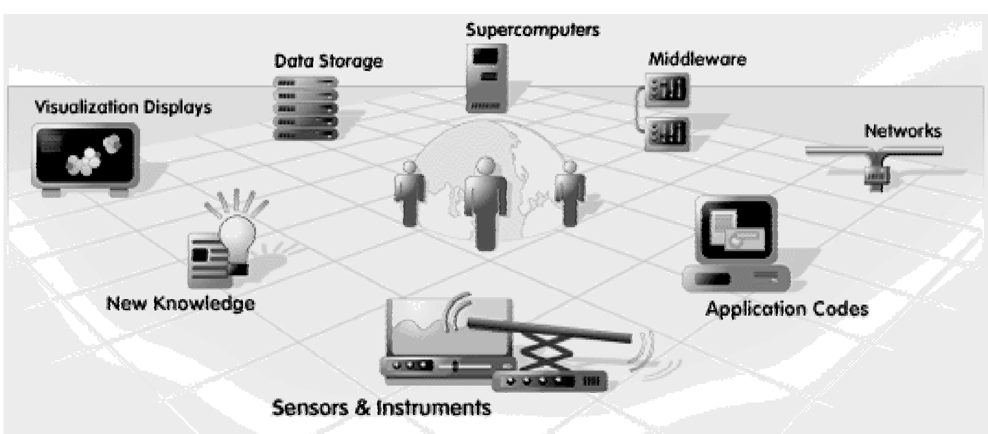

Figure 1: Framework of the NEESgrid.

NEESgrid is expected to shift the emphasis of earthquake engineering research from current reliance on physical testing to integrated experimentation, computation, theory, databases, and model-based simulation. Through NEES, the earthquake engineering community will be catalyzed to test and validate more complex and comprehensive analytical and computational numerical models, improving the seismic design and performance of our nation's civil infrastructure.

In this paper, the various NEESgrid components are introduced and their roles in NEESgrid are identified. Subsequently, the Multi-Site Online Simulation Testbed (MOST) experiment, which was conducted on July 30, 2003 to showcase the NEESgrid software capabilities, will be discussed. The MOST experiment coupled two large-scale physical experiments in Illinois and Colorado with a computational simulation. This paper also describes the Mini-MOST experiment, which is a tool for NEESgrid education, training and outreach. The success of the MOST and Mini-MOST experiments demonstrates the significant potential of Grid technologies in engineering research and, more importantly, the experience gained offered us a better insight on how to build and deploy effective Grid applications in the future.

AN INVENTORY OF NEESGRID COMPONENTS

The complete NEESgrid system comprises a variety of different components. At a high level, it is important to recognize that NEESgrid contains two classes of components.

1. Components that are primarily used to bring "nodes" (equipment sites or other resource providers) onto NEESgrid. These items will be of interest mainly to those who will be involved in equipment site operations and support and those who will be using equipment site resources in their work.

2. Components that are primarily used to operate the NEESgrid infrastructure itself in support of the entire NEES community. These items will be of interest mainly to those who will be responsible for operating the NEESgrid infrastructure and those who will be using centralized NEESgrid services (such as simulation capabilities) in their work.

Items Intended Primarily for NEESgrid Nodes	Items Intended Primarily for NEESgrid Operation
1. NTCP service and related software	1. OpenSEES
2. NSDS service and related software	2. FedeasLab
3. Data Repository services and related software	3. The NEESgrid Simulation Portal
4. Data Acquisition (DAQ) reference implementation	4. Website Materials
5. Information services	5. User Mailing Lists
6. E-notebook services	6. NEESgrid CVS Repository
7. Telepresence Mode (TPM) service	
8. The CHEF collaboration interface	
9. Administrative tools	

Items Primarily Used to Bring Nodes/Sites onto NEESgrid

NEESgrid software contains components that are used to bring a node/equipment site onto NEESgrid. These components – the main contents of the NEESgrid software distribution – are primarily used by equipment site operators and other resource providers to provide standard NEESgrid interfaces to their resources, and thus are used heavily in deployment activities and equipment site operations. The following sections detail each of these components and identify their role in NEESgrid.

NTCP Service

The NTCP (NEESgrid Teleoperations Control Protocol) service provides a common protocol that can be used by remote applications to control aspects of a physical experiment or simulation (Pearlman et al. 2003). The core NTCP service provides a NEESgrid-wide standard interface to local equipment and simulation capabilities. A programmer interface is provided in the Java and C programming languages for NTCP clients (programs that control experiments or simulations through the NTCP interface).

The NTCP service uses a plug-in architecture to integrate with local control systems and simulation code. Plug-ins written by equipment sites or simulation code developers manage the interface between the NTCP service and specific local equipment or software. The plug-in programmer interface is provided in both C and Java, and a number of example plug-ins (all of which have been used in experiments by equipment sites and/or software developers) are included in the NEESgrid software distribution.

NSDS Service and Related Software

The NEESgrid Streaming Data Service (NSDS) provides a uniform interface for subscribing to and accessing streaming data originating from experiments and simulations during their execution. Streaming data allows NEESgrid users to observe data remotely as a NEESgrid activity takes place. The core NSDS service provides a subscription interface to the data streams available at a given NEESgrid node. Data streaming itself is provided by DataTurbine. A programmer interface is provided in Java for interfacing a data source (e.g., DAQ system or simulation) to the NSDS/DataTurbine system. A sample Java implementation of a data provider is also provided, which allows any device that can support a simple network protocol to provide data to the system. Client interfaces to the NSDS and DataTurbine services are available in Java, and DataTurbine supports a wide variety of client interfaces for accessing streaming data.

DataTurbine, a high-performance multi-channel ring buffer from the Creare Corporation, will be used to store and stream data from DAQ systems, video, audio, and still cameras. This data will be viewable live or in a time-shifting Tivo-like manner. There will also be capabilities for permanently storing data automatically into the repository from DataTurbine. DataTurbine provides for precise time-synchronization of diverse channel types. DataTurbine acts in concert with the NEESgrid streaming data service (NSDS).

Data Repository Services

The distributed data repository allows users to manage data and metadata about NEES experiments and other activities. Architecturally, this component consists of local repositories, a central repository and archiving service, and a set of service protocols to link the repositories together and connect the repositories with NEESgrid users, applications, and portal components. The NEESgrid File Management Service (NFMS) provides clients with the ability to locate files independently of how and where they are stored, as well as the ability to negotiate transactions with storage systems. The NEESgrid Metadata Service (NMDS) provides a means for creating, managing, retrieving, and modifying metadata objects in a NEESgrid repository. All of the data repository services use NEESML, an XML format for defining NEES metadata schemas and uploading metadata objects into the NEES metadata repository. NEESML allows

NEESgrid users to describe real-world objects of interest to them while at the same time making it possible to browse and retrieve data and information gathered from all of the NEESgrid sites.

Data Acquisition (DAQ) Reference Implementation

To demonstrate the vision of how NEESgrid sites will interface local data acquisition (DAQ) systems to NEESgrid, a reference implementation is provided of a DAQ system, implemented in the popular LabVIEW framework. This reference implementation demonstrates interfaces to both streaming data services (NSDS) and the NEESgrid data repository services. This implementation also includes several test applications to simulate the NSDS, to stress test code, to measure bandwidth, to validate protocol implementation, and to simulate DAQ hardware.

Information Services

The NEESgrid distribution includes a number of information services for monitoring the status of NEESgrid sites and the NEESgrid system in general. These services are included with the software for NEESgrid sites because it must be installed at a site in order for the NEESgrid operations team to be able to monitor that site. Information services included with NEESgrid consist of Globus Toolkit's Monitoring and Discover Services (MDS), basic MDS information providers for monitoring system configuration and status plus a customized information provider for NSDS status and configuration, the BigBrother service for monitoring the entire NEESgrid system, and the Network Weather Service (NWS) for monitoring network status. Each of these services includes client interfaces for those who need to use them (primarily intended for NEESgrid operations personnel), and the CHEF service for each site includes basic viewers for much of this information.

E-notebook Services

E-notebooks are electronic (typically online) versions of laboratory notebooks. They allow lab notes to be entered by NEESgrid users, shared with other team members, and archived in the NEESgrid data repository for later use and review. NEESgrid e-notebooks allow users to store, review, search and share text, images, and documents.

NEESgrid offers two e-notebook options, one developed at Argonne National Laboratory which emphasizes easy-to-use data entry methods, and one developed at Pacific Northwest National Laboratory which emphasizes integration with the NEESgrid system. NEESgrid users may enter data into either notebook interface, transfer data from the Argonne to the PNL notebook, and submit data from the PNL notebook to the NEESgrid data repository for archiving.

Tepepresence Mode (TPM) Services

The Telepresence Mode (TPM) provides an interface and display that is intuitive to users using commodity WWW browsers such as Netscape, version 7.x and Microsoft Internet Explorer, version 5.x. Remote telerobotic video cameras with PZT (pan-zoom-tilt) capabilities enable users to remotely view the physical experiment. Fixed video cameras positioned by local collaborators also allow remote, site-specific observations. High-resolution, static images, uploaded by the local users, can also be viewed. TPM also enables synchronous and asynchronous monitoring of the preparation and construction of tests and test specimens.

CHEF Service

CHEF is a set of web-based collaborative tools that allow users to access computing facilities, communicate via discussion forums and text chat, and collaborate using scheduling and data sharing and exchange teamlets. The NEESgrid CHEF service also includes the NEESgrid DataBrowser and DataViewer, both of which allow users to browse and view repository data.

Administrative Services

The NEESPOP software distribution includes the GSI-OpenSSH tool, which provides a remote login service that uses Grid security credentials for authentication and authorization. This tool allows system administrators and other authorized users to login to the NEESpop using their NEESgrid credentials. The distribution also contains the MyProxy package, which includes the MyProxy client software used by the CHEF portal to obtain Grid credentials for users who login to the CHEF portal via standard web browsers. These Grid credentials are then used by CHEF to perform NEESgrid operations (like accessing the NEESgrid data repository, submitting simulation jobs, or subscribing to data streams) on behalf of the user. Note that the MyProxy software included here can also be used by the central NEESgrid operations team to establish a NEESgrid MyProxy server.

Other Software

The NEESgrid software distribution includes a number of other open source software packages that are either required by the NEESPOP software or that are likely to be needed in the future by NEESgrid operators. These include the Apache web server, various Web services and Web portal support packages, Grid programming libraries and commonly-used Grid services, and security tools.

Items Primarily Used for NEESgrid-wide Operations

The services and items described in this section will be operated by the NEESgrid operations/support team and made available for use by the entire NEES community. These items are typically installed in a central location rather than at each equipment site. The following sections will detail each of these components and identify their role in NEESgrid.

OpenSEES

The Open System for Earthquake Engineering Simulation, or OpenSees, is a software framework for computational simulation of structural and geotechnical systems. Simulation is an important part of NEES research, and OpenSees provides NEESgrid users a wide range of material and element models for nonlinear analysis (material and geometric), solution methods, and data processing procedures for research and problem-solving. OpenSees is open-source software, and it has an application program interface (API) for researchers to extend OpenSees for simulation applications. OpenSees is designed to support parallel processing, and researchers have been developing OpenSees applications for hybrid experimental methods.

FedeasLab

FedeasLab is a MATLAB toolbox developed at the University of California, Berkeley, for nonlinear analysis of frame structures. FedeasLab is useful for research in simulation methods because it allows rapid prototyping of models and computational methods within the MATLAB environment. FedeasLab is also ideal in education and development of courses on structural analysis, nonlinear analysis, and dynamic analysis. The toolbox contains a variety of beam-column elements and material models that can be used for nonlinear materials and P-Δ analysis. FedeasLab is designed hierarchically in a modular manner so that prototype developments can be re-implemented using a similar software architecture in OpenSees.

The NEESgrid Simulation Portal

The NEESgrid Simulation Portal supports remote access to the simulation capabilities of OpenSees. This Portal provides the following benefits to NEESgrid users:

- It eliminates the need for users to install and update the OpenSees software.

- It adds tools to help construct and validate OpenSees scripts and to visualize the results of scripts.
- It seamlessly integrates OpenSees simulation capabilities with NEESgrid's data repositories and computational resources.

Website materials

During the System Integration award period, the SI team has operated a website at http://www.neesgrid.org/. This website contains news, events, and a document repository for NEESgrid. A software section provides access to downloads, documentation, and help for all NEESgrid software releases. Additional information is available explaining the basic NEESgrid system and framework.

User mailing lists

During the System Integration award period, the SI team has operated a public email discussion list for use by early NEESgrid users. The list has been archived and the archive contains valuable answers to common questions that can be retrieved for subsequent use.

NEESgrid CVS Repository

The CVS repository is a generic CVS archive containing the NEESGrid code. This is used by the developers to manage the source code among multiple sites, and is the source for creating distributions.

Once the System Integration team completes its work in September 2004, the NEESgrid website, documentation, mailing lists, and CVS repository will be transferred to the NEES Consortium (http://www.nees.org).

CASE STUDY: THE MOST EXPERIMENT

To showcase the NEESgrid software capability for geographically distributed experiments, the Multi-Site Online Simulation Testbed (MOST) experiment was conducted on July 30, 2003 (The MOST Experiment, NEESgrid Report). This large-scale experiment linked physical experiments in the Newmark Civil Engineering Laboratory at the University of Illinois at Urbana-Champaign (UIUC) and at the Structures and Materials Testing Laboratory at the University of Colorado, Boulder (CU) with a numerical simulation at the National Center for Supercomputing Applications (NCSA), also in Urbana-Champaign.

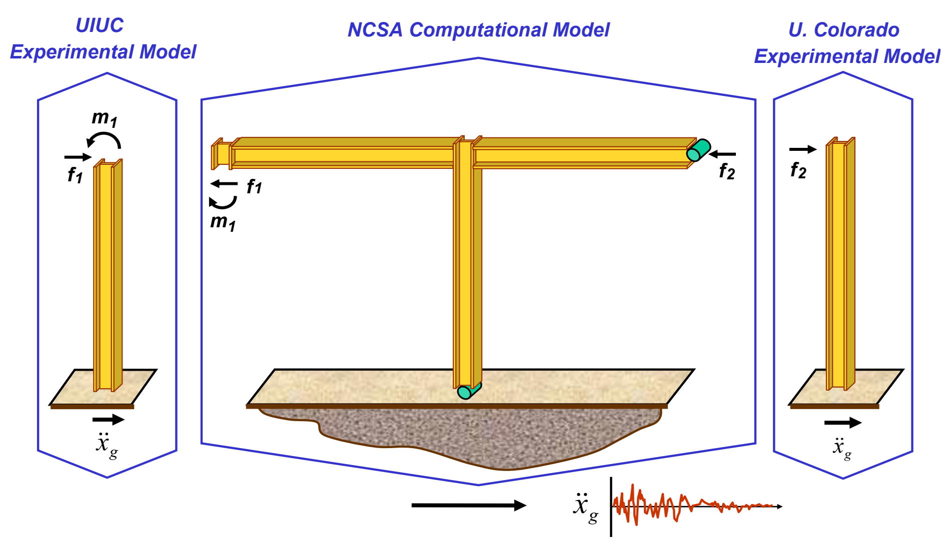

Figure 2: Experimental MOST Two-bay Single Story Steel Frame Structure.

The structure used in the experiment, shown in Figure 2, represents a two-bay single-story steel frame, like that of the interior of a multistory building. To test the structure, a method recently introduced in Japan (Nakashima et al. 1999; Watanabe et al. 1999) called Multi-Site Pseudo-Dynamic Substructure (MS-PSDS) was applied. In this method, the structure to be tested is divided into several substructures, and each of which is to be physically tested or numerically simulated at the same time at a different location in a coordinated manner. A simulation coordinator controls the overall experiment and communicates with the test sites and simulation computers. This experimental technique allows for testing a wide range of large structures that might otherwise be beyond the experimental capabilities of many laboratories.

Figure 3 illustrates how NEESgrid services are used to implement the MS-PSDS methodology. Specifically, NTCP was used to communicate between the various components of the experiment. A Simulation Coordinator provides overall management of the experiment. This component repeatedly issues a set of NTCP proposals based on current simulation state, collects information about the resulting state of all the substructures, and, based on that resulting state, computes the next set of NTCP commands to send. The coordinator also handles exceptions such as lost network connections or invalid responses. To help manage complexity, the MOST experiment was developed incrementally. First, a distributed simulation-only experiment was implemented and tested. Once the correctness of the simulation coordinator was verified, the two slave numerical simulations were replaced with physical substructures. The use of NTCP made this substitution transparent to the coordinator.

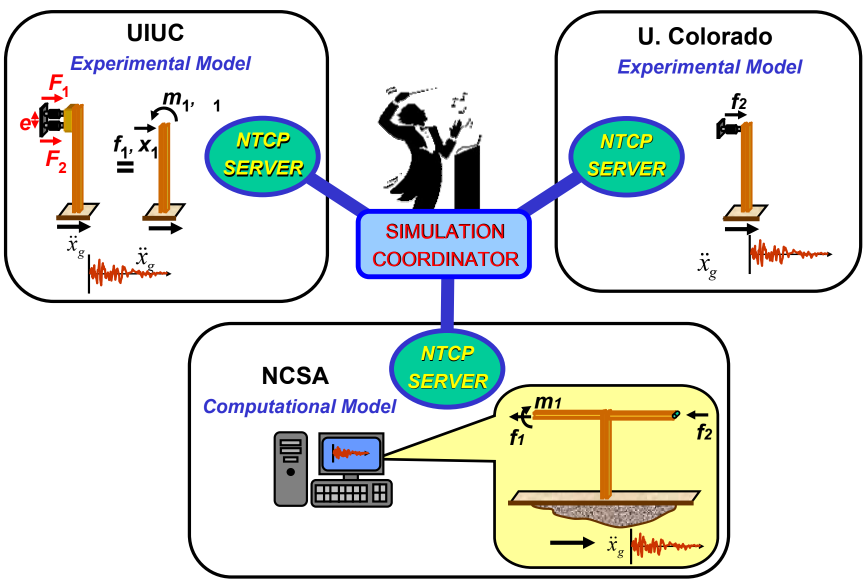

Figure 3: Modular Framework of the MOST Experiment.

The physical experiments are shown in Figure 4. The left column of the experimental frame was tested at UIUC and the right column at CU. Both columns were tested in a horizontal position. The UIUC column has a fixed connection between the column and beam, therefore the column acts in double-bending. Since moments at the top and the base of the tested column must be developed at the same time, two hydraulic actuators were used and the tested specimen was longer than the actual column. The CU column is a cantilever column because of the beam-column pin connection that connected it to the simulated horizontal beam. The central section of the frame was modeled by a simulation performed at NCSA on a Pentium 2.4 GHz Windows machine with 512 MB of memory.

For each time step simulated in the MOST experiment, force data was fed to the computational model at NCSA; the calculated displacements were sent to the Illinois and Colorado physical test sites; displacements were then applied to the physical models; and forces for the next iteration were measured and sent back to the computational model at NCSA. To emulate the structural response subject to a 15-second earthquake ground motion, this cycle was repeated 1,500 times during MOST experiment, which lasted approximately five hours.

During the experiment, the structural response was streamed to remote users and simultaneously stored in the main data repository for archiving. To observe the test and collaborate with others, users remotely accessed tools via a NEESgrid specific collaboration interface built using the CHEF collaboration framework (CHEF Project website). The CHEF interface used various NEESgrid protocols to authenticate to NEESgrid resources, access the metadata catalog, and download experimental data so that it could be viewed immediately by remote participants. CHEF also provided a range of useful collaboration tools such as a message board, access to an electronic notebook and an interactive chat.

Figure 5 shows some of the data viewers available via the CHEF interface. Arrangements of one or more views can be saved or viewed, and the Data Viewer automatically organizes a given arrangement to allow users to see each of the views. At the top of the Data Viewer, a set of VCR buttons allows users to play, pause, rewind, and fast-forward the data viewer, while at the bottom a clickable timeline allows users to see the state of the Data Viewer at any given time point.

During the MOST experiment, real-time video from both of physical testing sites was also available, with at least one accessible camera at each site.

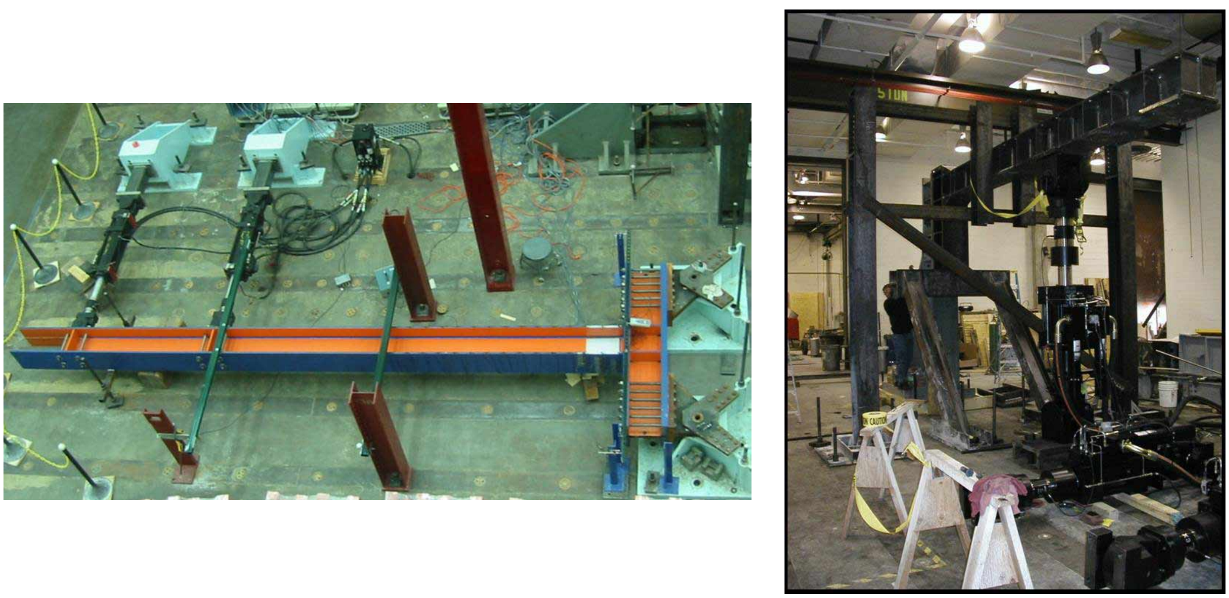

Figure 4: The Physical Substructure Test at the University of Illinois (left)
and the University of Colorado (right).

MOST Software Configuration

Details of the configuration of the NTCP control structure used in MOST are shown in Figure 6. The simulation coordinator was written in MATLAB. A MATLAB toolbox was developed for providing a convenient interface to NTCP; an earthquake engineer could then write and debug the NTCP client without detailed knowledge of the NTCP protocol.

The servo-hydraulics at UIUC were controlled by a Shore-Western control system, that would accept commands using a simple protocol over TCP/IP. The Colorado experiment used a control system running MATLAB's real-time operating system and a real-time MATLAB /Simulink simulation. Thus, different plugin interfaces were developed for each control system, utilizing NTCP's plug-in API.

The computational simulation at NCSA was also written in MATLAB. While the backend function of the simulation is quite different than the software used to implement the interface to the Simulink based control system, the exact same plug-in implementation used to connect NTCP to the Colorado control system is used to connect NTCP to the simulation.

Remote Monitoring Components in MOST

Sensor data from the two physical experiments were collected by a local data acquisition (DAQ) system. Conveniently, both sites choose LabVIEW as the software for their data acquisition. Thus, to interface the DAQ to NEESgrid, a simple LabVIEW interface was built that ran at the UIUC and Colorado sites and periodically gathered data deposited by the DAQ in a network-mounted file system; NFMS and GridFTP were then used to upload it securely to a NEESgrid accessible data repository. Once there, the combined data could be visualized using the CHEF-based data viewer. The same strategy was used to capture data generated by the simulation at NCSA. The same LabVIEW interfaced was used to stream a subset of the data via NSDS to remote observers (via CHEF) for immediate viewing.

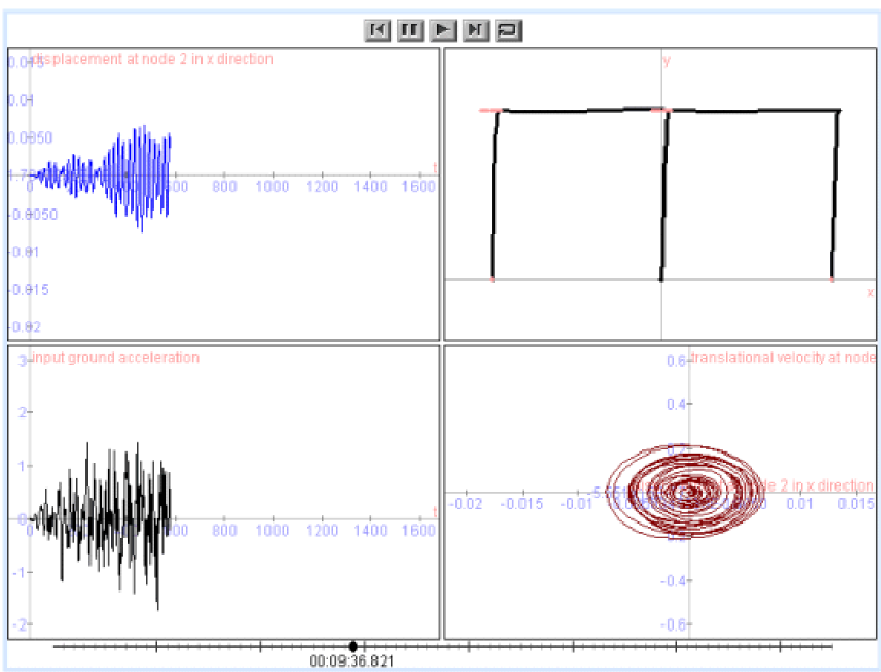

Figure 5: Configurable Data Viewers Implemented in CHEF Provided Near Real-Time Visualization of the Structure Response, Time Services Data from a Sensor, as well as Hysteresis Plots.

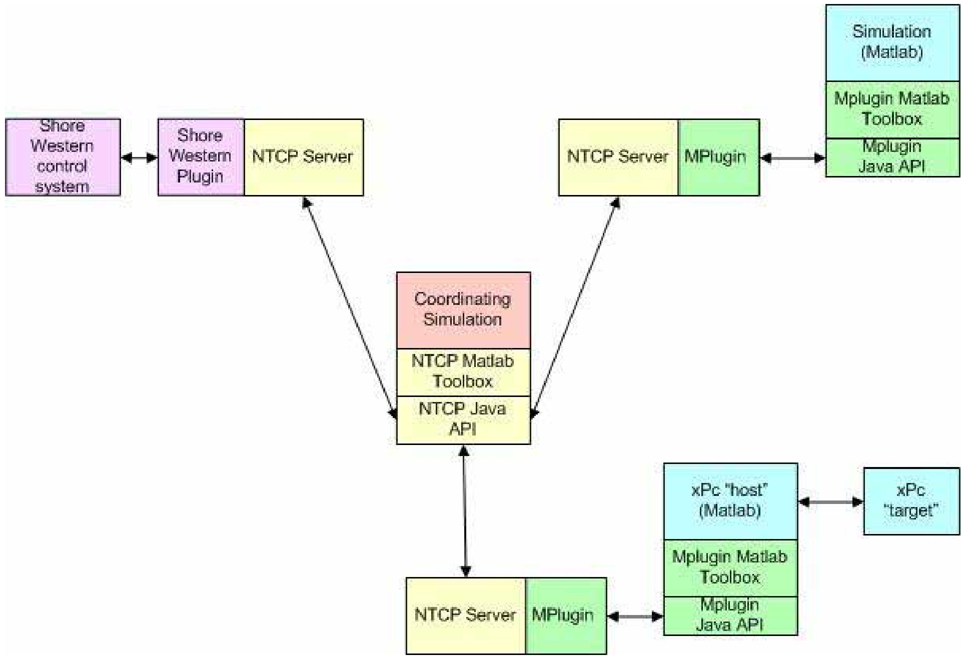

Figure 6: Control Components Used in the MOST Experiment.

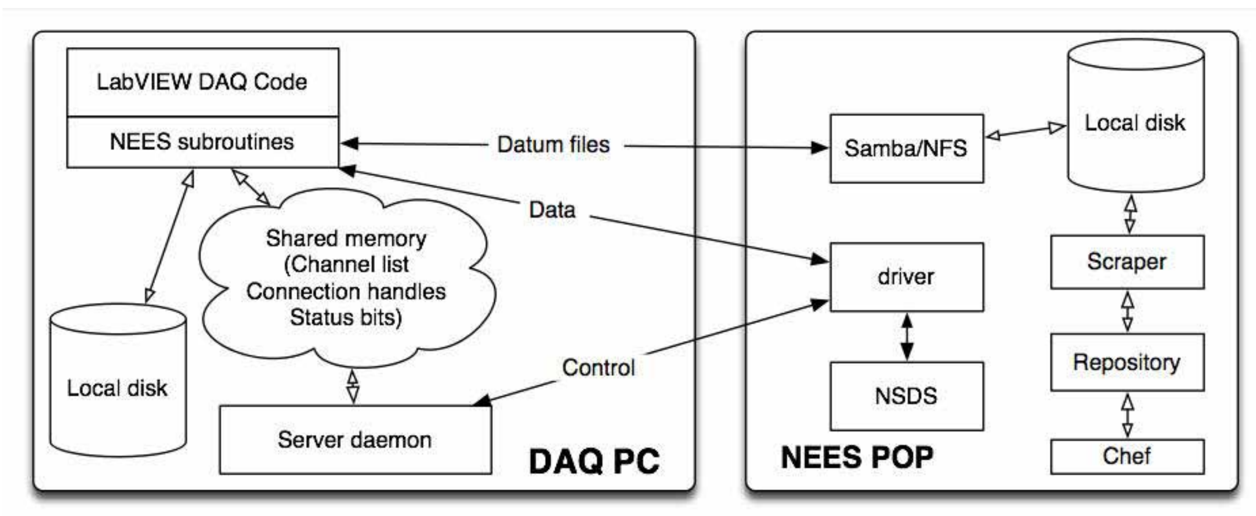

Figure 7: Major Components Related to Data Acquisition.

Metadata in MOST

For the MOST experiment, metadata was mostly generated manually and data was generated automatically from sensors. Metadata was developed by experimenters that described each of the three components of the experiment in terms of the structural configuration, material properties, and instrumentation. The metadata was designed so that non-participants viewing the stored data can understand the meaning of the sensor data in the context of the experiment. The metadata was developed in NEESML and uploaded to the repository prior to the experiment.

An early version of the NEESgrid data and metadata repository was used for the MOST experiment. The experiment served to exercise the data functionality and helped identify some areas that will be more fully developed in later releases, such CAS-based access control.

MOST Results

The results from the MOST experiment can be categorized into two distinct sections: the hybrid simulation experiment, and user interaction and participation.

The full, 1500-timestep distributed experiment was actually run twice: once as a “dry run” of the components directly involved in the simulation (the NTCP servers, physical experiments, and simulations), and then as the full experiment, available for viewing by remote participants. The dry run took about 5.5 hours and ran successfully to completion. The public experiment ran for more than 5 hours but exited prematurely at step 1493 (out of 1500). The fault tolerance features of NTCP enabled the simulation to tolerate several transient network failures throughout the day; however, the simulation coordinator had not been coded to take advantage of all the fault-tolerance features, and a final network error caused the simulation to terminate prematurely. This discrepancy has been addressed in the newer version of the software release (see http://www.neesgrid.org/software/neesgrid2.2/doc.php).

During the execution of the experiment, over 130 remote participants (some participants shared computers) logged on to observe the MOST experiment. These users observed live data stream from the University of Colorado CHEF server. The Chat feature was crucial to user interaction during the experiment. It allowed the developers to communicate with one another, while keeping the rest of the participants informed of the status and progress of the experiment. The sense of participation of the remote users was enhanced using the three telepresence cameras, which could be operated remotely.

Mini-MOST

Once the MOST experiment was complete, there was an obvious desire for a less-expensive, self-contained version that could be installed into an average lab. The Mini-MOST experiment (Figure 8) is a tabletop-sized system, with a single (1m by 10cm) beam, using stepper motors. It is an emulation of the CU portion of MOST and provides an excellent platform for education, training, and outreach for NEESgrid.

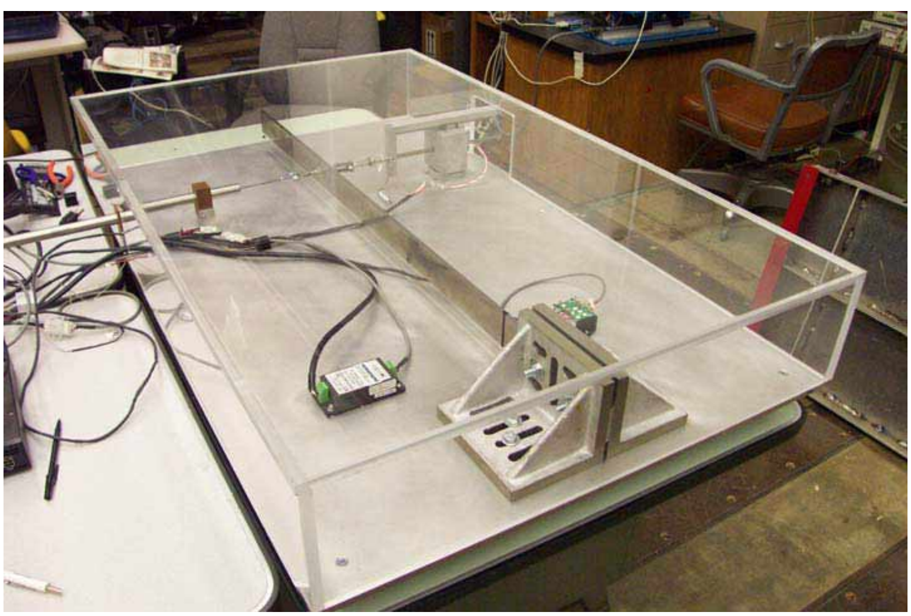

Figure 8: Mini-MOST.

The control and DAQ are run from a single Windows-based PC, which can also host the MATLAB simulation coordinator if required. Sensors are also scaled back to a strain gage, LVDT for position, and a load cell for force. In the first version, a single 25lb HIS size 23 non-captive stepper motor was used. Work is currently underway to add the second stepper motor and a rotation sensor to emulate the UIUC column of the full scale MOST experiment (Gehrig, 2004).

For simulation and debugging, code corresponding to the MOST variations was tested. A program where the beam is replaced by a numerical simulation has also been developed such that experiment can still be conducted when the actual hardware is not available. LabVIEW-based simulations of NTCP and the simulation coordinator have also been made available for purely local testing of motion and DAQ systems.

The control code is developed in LabVIEW, with a separate daemon program to handle the NTCP communications.

Security Considerations

In contrast to many Grid applications, telecontrol incurs serious health and safety risks, as well as the risk of damaging expensive equipment (Johnston et al. 1999). Several mechanisms are provided to help alleviate these risks: the usual Grid-based authentication and access control (Butler et al. 2000; Foster et al. 1998), and the ability in NTCP for sites, through the control plug-in mechanism, to enforce limits on what actions are allowed. Moreover, these services have been designed in such a way that the actual control systems do not need direct access to the external Internet.

However, because the NTCP implementation was not designed as a provably-secure software, and because NTCP and related components run on commodity operating systems (Linux and Windows), it is the responsibility of the experiment sites to employ appropriate operational procedures. In the case of the MOST experiment, these procedures included powering up the servo-hydraulics only when no one is near the experiment specimen, running a plug-in/backend system that required a human to approve each action (used only during initial testing at UIUC), and, whenever the servo-hydraulics are powered up, always having engineers nearby monitoring the experiment and prepared to execute an emergency stop if necessary. The small scale nature of Mini-MOST makes the associated health and safety risks far smaller; the primary precaution taken was the creation of a plexi-glass cover for the Mini-MOST experimental setup.

ONGOING WORK

The software used in the MOST experiment was initially released in October 2003, and a newer version was released in March 2004 as a standard package in the NEESgrid Software 2.2 (see http://www.neesgrid.org/software/neesgrid2.2/doc.php).

Several new experiments using the NEESgrid framework are being planned as part of NEESgrid's Experiment-based Deployment activities (Marcusiu, 2003). A UCLA team of earthquake engineers plans to perform field testing of an existing four-story office building in Los Angeles. They intend to apply earthquake-type and harmonic forces to the building, gathering acceleration, strain, and displacement data using wireless sensor arrays (802.11 wireless telemetry) to evaluate the response and behavior of the structure. Data and video streams will be recorded and archived at a mobile command center before transmission to the laboratory using satellite telemetry.

Earthquake engineers at RPI, UIUC, and Lehigh University plan to use the NEESgrid framework to study soil-structure interaction in an experiment involving two structural sites (UIUC and Lehigh), one geotechnical site (RPI), and a computational simulation node at NCSA. The experiment will focus on an idealized model of the Collector-Distributor 36 of the Santa Monica Freeway that was damaged in the 1994 Northridge earthquake of California.

Engineers at UC Davis are working on an experiment that uses the NEESgrid framework to characterize how the properties of soil change during shaking or ground improvement. This experiment includes remote operation of a robot arm that will be attached to their centrifuge and of piezo-electric bender element sources and receivers embedded within the centrifuge model. The robot arm has exchangeable tools: a stereo video camera tool for telepresence, an ultrasound tool for imaging, a cone penetrometer, a needle probe for high resolution imaging, and a gripper tool for installation of piles and manipulation/loading.

At the University of Minnesota, an experiment is planned that will use the NEESgrid framework to operate a six-degree-of-freedom controller, to apply realistic deformations and loading quasi-statically to large-scale structures. This experiment will also use video and still images as data, using the NEESgrid framework to trigger still image capture.

The MOST experiment and most of the follow-on experiments can tolerate fairly long delays without affecting the experiment's results. The NEESgrid team is working with engineers from UC Berkeley, the University of Colorado, SUNY-Buffalo, and Lehigh University to support distributed experiments with near-real-time requirements. This work has two facets: computer scientists are working on improving the performance of NTCP, while the earthquake engineers are developing simulation and control software that can better tolerate delays.

CONCLUSIONS

NEESgrid is an Internet-based, national-scale high performance network system for NEES, build on proven, existing Grid technology. The layered, modular architecture of NEESgrid is flexible and highly extensible and will enable rapid development of new end-user applications, the introduction of new services, and the integration of new simulation software and experimental facilities as they are developed.

In constructing NEESgrid, an important new test facility has been created. A Grid-based framework for distributed hybrid tests makes these tests more practical to design and perform, providing tools to deal with heterogeneity and policy issues. In addition, as part of the exercise, the effectiveness of the service oriented architecture and the stateful service model that is at the core of OGSI has been demonstrated. Many of the technologies that we have developed for NEESgrid are expected to have application in domains outside of earthquake engineering. For example, NTCP and NSDS can be used to control and observe a wide range of devices; plans are underway to investigate this possibility in the setting of other remote sensing and control applications such as tele-microscopy. In summary, NEESgrid has the potential to be of great value to the earthquake engineering community, as well as representing an important class of Grid application.

ACKNOWLEDGEMENTS

This work was supported by the George E. Brown, Jr. Network for Earthquake Engineering Simulation (NEES) Program of the National Science Foundation under Award Numbers CMS-0117853 (NEESgrid), CMS-0217325 (NEES MUST-SIM at UIUC), and CMS-0086592 (NEES FHT at CU-BOULDER).

We gratefully acknowledge the contributions of Daniel P. Abrams, Sung Joo Bae, Jean-Pierre Bardet, Ian Buckle, Randy Butler, Ben Clifford, Mike D'Arcy, Liang-Kuang Chen, Amr Elnashai, David Gehrig, Dan Horn, Erik Johnson, Young Suk Kim, Dan Kuchma, Ravi Madduri, Mark Marikos, Gilberto Mosqueda, Nancy Moussa, Narutoshi Nakata, Gokhan Pekcan, Pawel Plaszczak, Tom Prudhomme, Chase Phillips, Daniel Reed, Andrei Reinhorn, Kathleen Ricker, Hatem Seliem, Benson Shing, Eric Stauffer, Bozidar Stojadinovic, and Nestor Zaluzec to the development of the NEESgrid system and the MOST experiment.

REFERENCES

1. Comprehensive Collaborative Framework (CHEF) Project Web Site, 2004. http://chefproject.org.
2. "The MOST Experiment, July 30, 2003" (2003). *NEESgrid Technical Report*, http://www.neesgrid.org/documents/MOST_document_v1.0.pdf.
3. Butler, R., Engert, D., Foster, I., Kesselman, C., Tuecke, S., Volmer, J., and Welch, V. (2000). "A National-Scale Authentication Infrastructure." *IEEE Computer*, 33 (12). 60–66.
4. Gehrig, D. (2004) "Guide to the NEESgrid Reference Implementation." *NEESgrid TR-2004-04*, http://www.neesgrid.org/documents/TR_2004_04.pdf
5. Foster, I., Kesselman, C., Tsudik, G., and Tuecke, S. (1998). "A Security Architecture for Computational Grids." *5th ACM Conference on Computer and Communications Security*, 83–91.
6. Johnston, W. (1999). "Realtime Widely Distributed Instrumentation Systems." Foster, I. and Kesselman, C. eds. *The Grid: Blueprint for a New Computing Infrastructure*, Morgan Kaufmann, 75–103.
7. Marcusiu, D. (2003) "NEESgrid Experiment-based Deployment Strategy and Requirements." http://www.neesgrid.org/documents/NEESgrid_epd_strat_req_v1.0.doc
8. Nakashima, M., Kato, H., and Takaoka, E. (1999). "Development of Real-Time Pseudo Dynamic Testing." Earthquake Engineering and Structural Dynamics, 21. 79–92.
9. Pearlman, L., D'Arcy, M., Plaszczak, P., and Kesselman, C. (2003) "NTCP Control Plugin. NEESgrid." *NEESgrid TR-2003-16*, http://www.neesgrid.org/documents/TR_2003_162.pdf.
10. Watanabe, E., Sugiura, K., Nagata, K., Yamaguchi, T., and Niwa, K. (1999). "Multi-phase Interaction Testing System by Means of the Internet." *1st International Conference on Advances in Structural Engineering and Mechanics*, Seoul, Korea, 43–54.