The NEEShub Cyberinfrastructure for Earthquake Engineering

The US Network for Earthquake Engineering Simulation (NEES) operates a shared network of civil engineering experimental facilities aimed at facilitating research on mitigating earthquake damage and loss of life. The NEEShub gateway was created in response to the NEES community's needs, combining data, simulation, and analysis functionality with collaboration tools.

There's a growing need for a reliable, effective, and comprehensive cyberinfrastructure to unite and empower virtual communities of researchers, students, and practitioners in science and engineering. This need is driven in part by the globalization of science and engineering, as well as by the need to share the valuable research data and software tools used in discipline-specific communities. The recent 11 March earthquake and tsunami in Japan, which caused significant damage and loss of life, brings this need into sharp focus.

The George E. Brown Network for Earthquake Engineering Simulation (NEES), funded by the US National Science Foundation, operates a shared national network of civil engineering experimental facilities. The goal of NEES is to facilitate research into effective ways of mitigating earthquake damage and loss of life through the use of improved designs, materials, construction techniques, and monitoring methods. To support this research in the civil engineering community, NEES operates a distributed network of 14 research sites across the US. The NEES objectives are to

- develop a national, multiuser research infrastructure to enable research and innovation in earthquake and tsunami loss reduction;
- create an educated workforce in hazard mitigation; and
- conduct broader outreach and lifelong learning activities.

Experimental capabilities at the 14 NEES sites include large-scale earthquake simulators (or "shake tables"), a tsunami wave basin, large-scale testing facilities, centrifuges, field and mobile facilities, a large-scale ground displacement facility, and cyberinfrastructure capabilities.

NEES is one example of a large-scale science and engineering project that depends on a reliable cyberinfrastructure. NEEScomm, led by Purdue University, connects the 14 NEES research equipment sites and the earthquake-engineering community with a powerful IT infrastructure and a commitment to education, outreach, and training related to earthquake engineering. There are many other examples of successful large-scale cyberinfrastructure projects, including the Compact Muon Solenoid (CMS)[1] project (www.physics.purdue.edu/Tier2) and the Ocean Observatories Initiative (OOI).[2] The cyberinfrastructure-supporting projects are focused primarily on

1521-9615/11/$26.00 © 2011 IEEE
Copublished by the IEEE CS and the AIP

Thomas J. Hacker, Rudi Eigenmann, Saurabh Bagchi, Ayhan Irfanoglu, Santiago Pujol, and Ann Catlin
Purdue University
Ellen Rathje
University of Texas at Austin

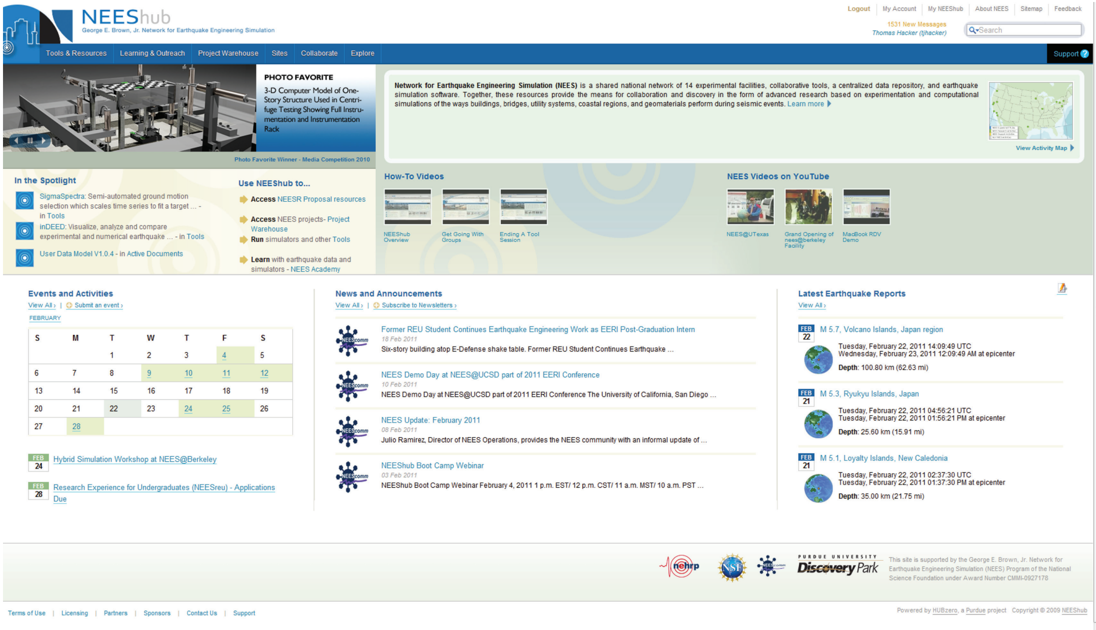

Figure 1. The Network for Earthquake Engineering Simulation hub (NEEShub). The hub provides the earthquake-engineering community with a shared virtual space where researchers and practitioners can combine data, simulation, and analysis functionality with collaboration tools.

collecting, managing, and disseminating research data to meet the needs of the research community to address the problems arising from the data deluge.[3]

Data and computation are intrinsically linked at the most fundamental level in computing; although data can represent the physical reality it seeks to characterize, we also need software tools that can explore and distill useful knowledge from these data. The suite of necessary tools include those for simulation and analysis as well as essential tools such as database management systems, project management software, data curation, revision-control systems for source code, and documentation systems. Although a project's software tools can be discipline specific, many packages are generic and can be reused across many projects. Bringing together computation and data isn't a complete solution, however; we also need to create a virtual space where the community can congregate.

In response to this need, we're developing the NEEShub for earthquake-engineering community (see Figure 1). Anchored in Purdue research[4] and based on HUBzero technology[5] (which is used by the popular nanoHUB science gateway[6]), NEEShub combines data, simulation, and analysis functionality with collaboration tools for earthquake-engineering researchers and practitioners. By developing NEEShub in close partnership with earthquake engineers, we're ensuring that NEEShub provides the community with the most meaningful and useful modes of access to data and tools.[7]

NEEShub Features and Functionalities

The NEEShub provides functionalities for

- collecting and using experimental and simulation data,
- running simulation and analysis software using data stored in NEEShub,
- content and tools for learning and outreach, and
- virtual facilities for developing and sustaining a virtual community.

The NEEShub development process follows the computing industry's standard practices of gathering user requirements, developing designs and prototypes in response to user needs, and developing and following a project schedule for software development and testing. Following a structured software development approach has proven successful in delivering a high-quality cyberinfrastructure on time and within budget.

Building on a Solid Science Gateway

NEEShub is built on the HUBzero framework developed by Purdue University's Hub Technology Group led by Michael McLennan. Hub technology was originally developed in conjunction with the Network for Computational

Nanotechnology, sponsored by the US National Science Foundation, to provide the science gateway www.nanohub.org to support the distributed nanotechnology community. HUBzero technology[5] uses a customizable Web-based science gateway that provides

- the ability to develop, execute, and share software within the hub environment;
- a community forum for posting questions about and answers to problems;
- a shared live tool session that facilitates direct cloud-enabled collaborative work; and
- user interfaces for simulation and visualization.

A detailed overview of HUBzero technology is described in the April 2010 issue of *CiSE*.[5] One of the most novel features of HUBzero technology that NEEShub uses is middleware for hosted execution, which provides a secure environment based on OpenVZ (http://download.openvz.org/doc/OpenVZ-Users-Guide.pdf) virtualization for hosting user applications connected to a browser window using virtual network computing (VNC).

Figure 2 shows the NEEShub science gateway's architecture. Tools running in the hub can be shared in real time with other hub users in a multiuser, single-screen sharing model, which allows direct collaboration among researchers in tool use and development. NEEShub leverages these HUBzero functionalities and adds new features for collecting, managing, and disseminating the large amounts of scientific data generated by the earthquake-engineering research community.

The NEES cyberinfrastructure built from the NEEShub science gateway includes a large centralized file storage server connected to key infrastructure components via a high-speed network. We designed the IT infrastructure with the capacity and performance needed to manage increasing user load and data volume as demand increases over time. Another feature of the NEES cyberinfrastructure is the extensive use of virtualization for operating NEEShub infrastructure components across approximately a dozen shared servers. We operate approximately 60 development platforms, preproduction staging areas, and training servers as VMware virtual machines, where each VM is a clone of the production NEEShub server. This lets us

- test changes prior to migrating to production,
- quickly establish high-volume services for flash crowds arising from intensive training or other events,

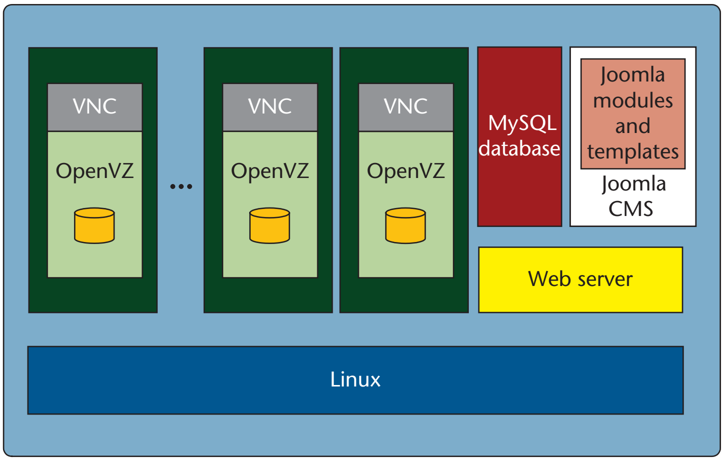

Figure 2. NEEShub science gateway architecture. The architecture couples standard software components, such as Virtual Network Computing (VNC), the Joomla Content Management System (CMS), OpenVZ, and Linux into a power cyberinfrastructure for the NEES community.

- and easily version and archive different generations of the cyberinfrastructure as we develop and deploy new services.

Our approach to cyberinfrastructure has proven scalable and successful, and has significantly reduced the amount of physical infrastructure needed while simultaneously letting us provide more services.

We developed the NEEShub through an iterative process of building prototypes and gathering rapid feedback from earthquake engineers. To create stage environments, we used VMware to construct VM images of the NEEShub server to quickly establish development sandboxes and prototypes. Based on feedback from the earthquake-engineering community on structure and content, we propagate approved NEEShub updates to the production server. The resulting NEEShub design lets earthquake engineers easily navigate and access data, discover relevant and useful content and software tools, and quickly launch hub-based earthquake-engineering tools.

Our extensions to HUBzero have added new capabilities to integrate data with site, project, and experiment metadata; simplify the project data navigation and use; and run Windows software tools within the hub environment. The data features we added provide

- a connection to an external Oracle database to retrieve NEES metadata;
- access to NEES project data residing on a file server through a 10-Gigabits-per-second (Gbps) network;

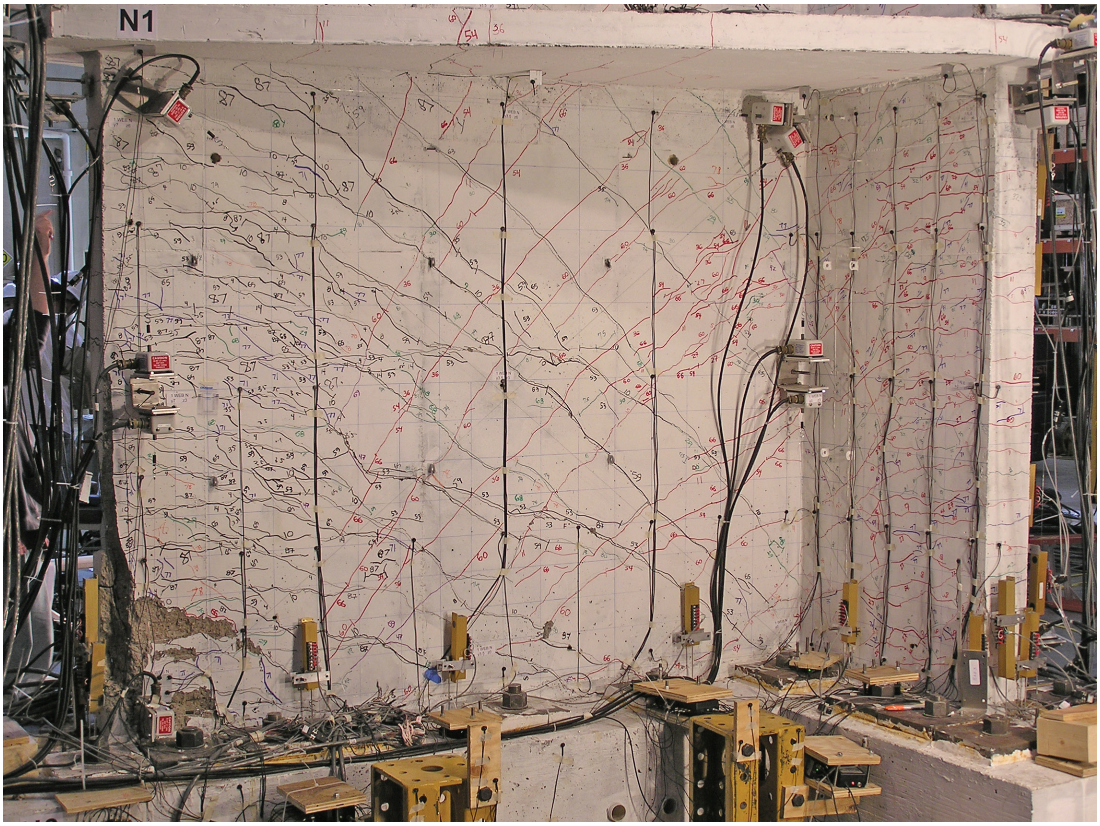

Figure 3. An example NEES project. To investigate the effects of multidirectional loads on nonrectangular walls, a reinforced concrete wall was instrumented with strain gauges, displacement transducers, inclinometers, and load cells.

- a Representational State Transfer (REST) Web services interface to NEES data; and
- the Project Warehouse, editor, and explorer for NEES (PEN).

All tools running in the NEEShub environment can access NEES project data and the user's private data through a mounted file system in the user workspace. There are currently 42 earthquake-engineering tools available in NEEShub, and we're adding the capability of using Windows software tools through the hub interface within secure VM containers.

Managing Data

Research conducted within the NEES community produces data from physical experiments, computer simulations, and hybrid simulations that link physical and computational simulation. Experiments conducted at the NEES sites involve several months of test structure construction. Figure 3 shows an example of experiments conducted at NEES sites—a reinforced concrete structure tested in an experiment conducted by Professor Catherine French at the University of Minnesota NEES Multi-Axial Subassemblage Testing (MAST) laboratory.[8,9]

As the image shows, in this project, a reinforced concrete wall was instrumented with strain gauges, displacement transducers, inclinometers, and load cells. Figure 4 shows an image from a YouTube video of the experiment, which is linked from the NEEShub. The concrete wall was loaded using a multiaxial frame attached to a strong wall at the MAST laboratory. This experiment, which was part of a 2.5-year project, generated a total of 327 Gbytes of curated experimental data and documentation contained within 6,937 files. The data types curated included photographs, spreadsheets containing columns of time-series sensor data, documentation (Word documents and PDFs), and videos of the experiments.

The magnitude and complexity of the experimental data for this project make it difficult to easily navigate and study the information collected from the experiment. The original NEEScentral data navigator[10] presented data to users on a per-file basis, which required users to download and review numerous files on their computers on a file-by-file basis. This made it

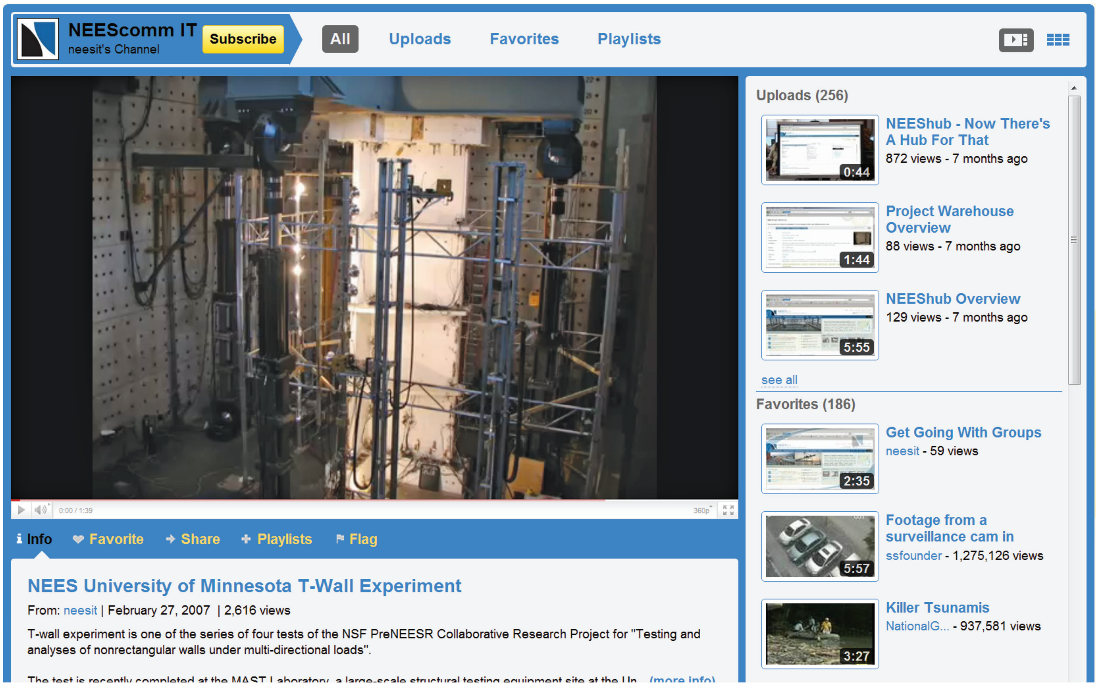

Figure 4. NEES Experiment YouTube video accessible through the NEEShub. Videos collected from experiments conducted by the NEES research community and NEEShub tutorials are disseminated to researchers, practitioners, and the public using NEEShub and YouTube. Users can search for information about specific NEEShub projects and YouTube video content using Google.

difficult for them to visualize and understand quickly the experiment's salient features.

Most NEES research projects are similar to Professor French's project in several ways: significant time, effort, and facilities are required to build physical specimens; experiments conducted on specimens are destructive and can take place only once; data are collected from a variety of instruments and come in a variety of formats; the data's sheer quantity and variety make it difficult to distill insight without an analysis and visualization environment; and the final data are irreplaceable and must be archived and curated for long-term preservation and use.

The earthquake-engineering community has identified needs for high-speed transport, use, management, and long-term curation of NEES project data as one of the highest priorities for the NEES cyberinfrastructure. In response, we're creating a comprehensive project repository system that seeks to encompass the entire data life cycle, which includes the definition, creation, ingestion, use, and eventual long-term data curation.

The NEES data life cycle involves several stages: definition, which provides standard schemas and formats for data; data collection, annotation, and uploading; data cleaning and curation; data use and downloading; and long-term curation of experimental data. Our cyberinfrastructure approach is novel in that we're working to develop a cyberinfrastructure linked to the data life cycle framewire, instead of a cyberinfrastructure that's solely oriented around computation. This data-oriented cyberinfrastructure framework seeks to simplify and ease the work involved in collecting, uploading, managing, and using project information. We've already accomplished work on the data life cycle's first stage—definition—by defining a standard schema and data model for NEES experimental data.[11,12] We've also developed the software tools needed for data collection, use, annotation, download, and curation.

Improving NEES Data Definition

Science and engineering data destined for reuse must be carefully and completely described because the data's eventual reuse might be far removed from the researcher who initially collected the information. The historical analogue to this problem is in the definition and use of standard formats for physical laboratory notebooks, in which sufficient metadata and certification is required to ensure the long-term integrity and use of the data that they hold. Data collected by NEES researchers are represented in a wide variety of

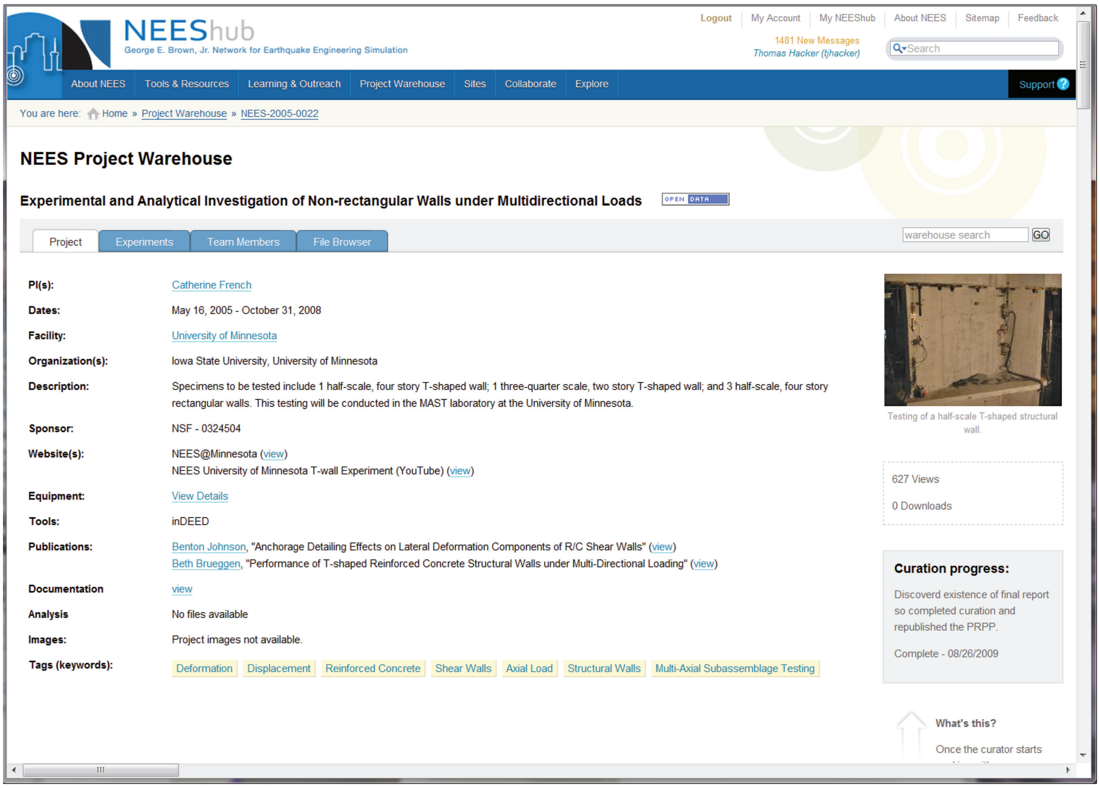

Figure 5. NEES project displayed in the Project Warehouse. The warehouse provides the project abstract, the list of project researchers, and the list of publications resulting from the project.

formats and sizes and are stored in individual data files, which, without context, have little meaning and limited reusability for others.

To address this problem, the San Diego Supercomputer Center's NEESit group developed a standard metadata schema format and data model for NEES experimental data. The group's NEESit Data Model[12] describes common data schemas and relationships for metadata and files associated with a NEES project. Project data are organized into a general hierarchical structure that reflects the way in which data are collected and processed at NEES research sites. The top level, *project*, represents the overall research project. The second level, *experiment*, represents a physical test specimen or computational simulation model that was developed within a project. The third level, *trial*, is an individual test conducted as part of an experiment. The fourth level, *repetition*, refers to repetitions of a given test conducted under the same conditions or parameters.

Each level in this data model has associated schemas that contain information about the specific instances of a project, experiment, trial, or repetition. The metadata these schema represent provide the information needed to describe fully the experimental data and the relationships among data. For example, the project-level metadata includes the NSF award number and identities of the principal investigators leading the project; the trial level includes a description of the trial's objective. Subsequently, we can use the resulting metadata for searching for relevant information in stored NEES projects.

Purdue University's NEEScomm team is building on this model and has extended it to create the NEES Project Warehouse, a Web-based project display and visualization system. The Project Warehouse provides a comprehensive and integrated view of NEES data stored within the central NEES database and file system. Figure 5 shows a Project Warehouse project within the NEEShub. The warehouse provides a project abstract, a list of project researchers, and a list of publications resulting from the project. The Web page provides tabs to let users review and download detailed experimental data, a list of project personnel, and links to invoke visualization tools within the NEEShub environment. As of October 2010, there were 333 projects in the Project Warehouse; 74 projects had been completed, and 60 had been completely cleaned, organized, and curated.

The NEEShub lets users create new projects and provide project metadata through the Project Editor, which lets users upload a few data files for the project, describe the number of project experiments and trials; and cite project publications. To upload or download more files or a large amount of data from the Project Warehouse to their local computer system, researchers can use PEN.

PEN: Data Collection and Transfer

One of the major problems the NEES community identified in using data were difficulties in uploading and downloading the gigabytes of data associated with a NEES experiment. In 2008, the University of Texas at Austin NEES site developed a tool named SingleShot,[13] which let users organize files to be uploaded into a standard directory structure for a bulk data upload operation. SingleShot let users organize many files on their local computer and then upload the file set to NEEScentral in a single operation.

The NEES team at Purdue evolved SingleShot's functionality to create a suite of data management tools that lets users build a local repository of project data on their own computers and synchronize it with the NEEShub's main data sources. PEN lets users connect to a project in the Project Warehouse and download the project data as a whole or in subsets in a single operation. As the project progresses and new data are ready for upload, users simply upload the newly added files. To ensure data integrity across download and upload operations, PEN uses a stored checksum for each file to detect any changes in the files. PEN can transfer numerous files in a short time; for example, we transferred 5,553 Mbytes (2,000 files) from a desktop computer to the Project Warehouse in 35 minutes.

Using the Data

Computation fundamentally involves the interaction of data with algorithms. In a distributed computing system, in which data are physically distant from computation, the speed and ease of processing data are significantly affected by the bandwidth and latency of the communication channel connecting the data and computational resources. This is a fundamental problem for cyberinfrastructure systems that rely on a core data repository or core computing facility.

To resolve this problem, cloud-computing systems have emerged over the past few years that provide software-as-a-service (SaaS).[14] SaaS seeks to place computation near the data to improve performance and to expand the size and scope of data sets that software can access. The HUBzero framework provides SaaS by letting researchers run software tools within the hub environment on a server that's physically near the core data.[5] Instead of the usual process of downloading and installing software on a local computer followed by downloading or accessing data through a network, the HUBzero approach makes it much faster and simpler to run software without the need to install it on a local computer system or download large amounts of data. NEEShub uses HUBzero SaaS functionality to provide access to a wide range of civil and earthquake-engineering tools.

The Interactive Display for Earthquake Engineering Data (inDEED) is an example of a software tool that links to data in the NEES Project Warehouse. inDEED offers NEES researchers an integrated way to process, visualize, analyze, and compare data from NEES projects. Using inDEED, information from sensors, drawings, images, test annotations, and videos can all be visualized within a single program. inDEED leverages one of the NEEShub environment's fundamental strengths—the ability to co-locate and integrate large volumes of complex data, software analysis and simulation tools, and the researcher and practitioner community.

In a distributed computing system, in which data are physically distant from computation, the speed and ease of processing data are significantly affected by the bandwidth and latency of the communication channel connecting the data and computational resources.

Figure 6 shows a screenshot of inDEED displaying results from Professor Saiid Saidi's University of Nevada at Reno NEES project "Collaborative Research: Using NEES as a Testbed for Studying Soil-Foundation-Structure Interaction" (http://nees.org/warehouse/projects/32). The display originates directly from the project metadata and curated project data stored in the NEEShub. The raw sensor displacement data from the shake table used in this experiment can then be plotted with inDEED within the NEEShub, as Figure 7

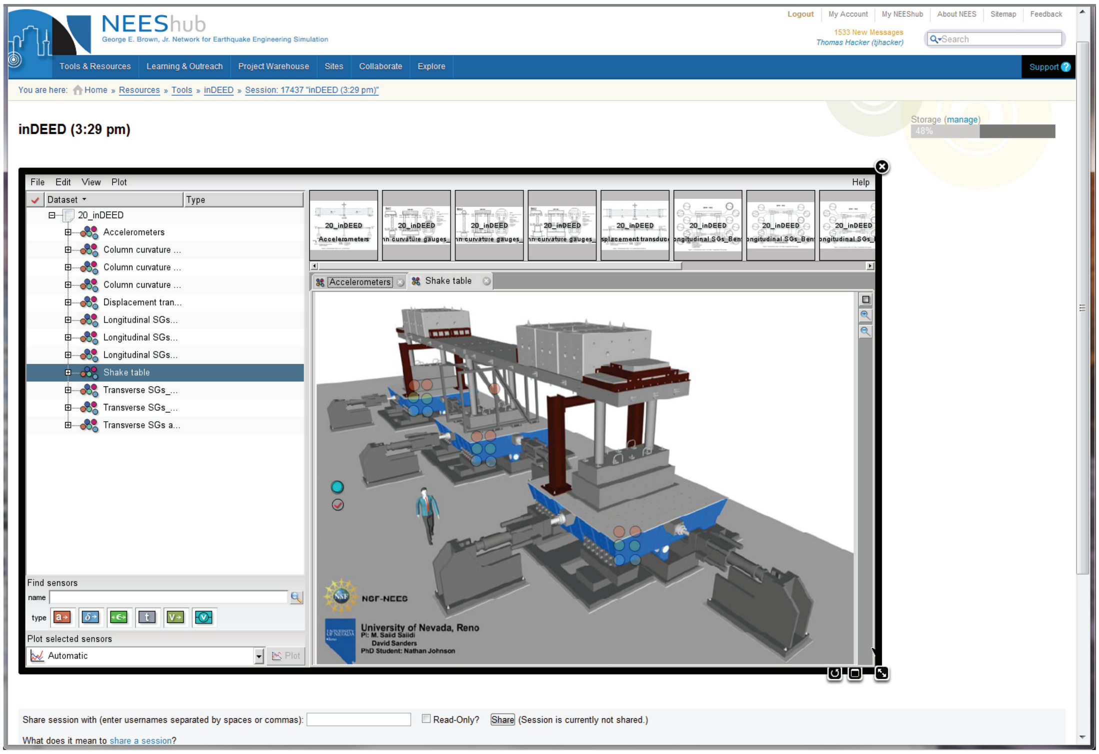

Figure 6. The interactive Display for Earthquake Engineering Data (inDEED). The inDEED interface offers researchers a way to process, visualize, analyze, and compare experimental and numerical earthquake simulation data. inDEED lets NEEShub users access data stored in the NEES Project Warehouse.

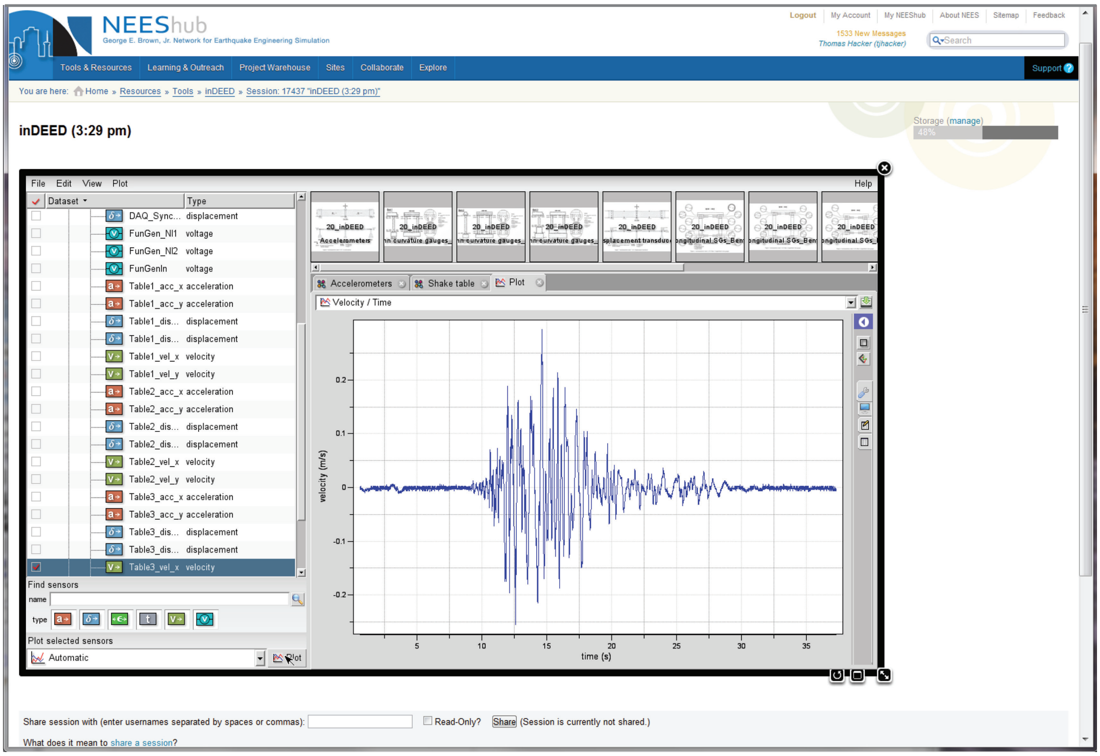

Figure 7. The sensor displacement data plotted using inDEED within the NEEShub. inDEED features include single- and multiple-sensor plots, interactive visualization, and data value display. It also lets users export graphs for use in other applications.

shows. inDEED provides features for single- and multiple-sensor plots, interactive visualization, data value display, and exporting graphs for use in other applications.

The NEEShub currently provides access to 34 tools that can be run within the NEEShub environment or downloaded for use on the user's computer. An example of a NEEShub tool is OpenSees,[15] which is an open source simulation package widely used in the earthquake-engineering research community to simulate the response of a structural and geotechnical system to seismic loads. Frank McKenna's article, "OpenSees: A Framework for Earthquake Engineering Simulation" (page 58) offers detailed information on OpenSees. Through its NEEShub implementation, researchers can use OpenSees to execute standard- to large-scale simulations.

Frank McKenna, one of the authors of OpenSees, measured OpenSees' performance in NEEShub in late 2010, and compared the execution time with the same problem running on a local computer system; he found a 32-percent reduction in execution time. To support the development of new tools, NEEShub provides a workspace tool with a development environment that lets users prepare and test their applications for use as a NEEShub-supported tool.

NEEShub's science gateway approach also offers users access to applications that use high-performance computing resources (such as the Open Science Grid and TeraGrid). By simplifying the process of using HPC resources, NEEShub aims to expand the community of high-end resource users who might not otherwise join the community. NEEShub provides a new form of SaaS cloud computing that puts more control over the software in users' hands; they can then actively contribute to and create cloud computing content.

Cybersecurity in NEES

NEES has developed a comprehensive cybersecurity approach that includes best-practice cybersecurity policies and mechanisms at NEEScomm and an annual security audit at each of the NEES sites. The goal of our cybersecurity practices is to protect the computational resources and data at NEEScomm from malicious access that might violate the data's confidentiality or integrity or render the NEEShub or its data unavailable. We deploy five classes of security controls: access control (to regulate access to various NEEShub resources); user authentication; network intrusion detection; host intrusion detection; and diligent maintenance of the software infrastructure.

In terms of the human cybersecurity element, we don't have administrative control of the servers at the 14 sites, even though they're used to access NEEScomm resources. This emphasizes the need for training, communication, and coordination with the site IT managers for ensuring the IT infrastructure's overall security. Further, the cybersecurity team is successful only if it keeps the larger IT group at NEEScomm aware of the security controls and makes the effort to understand the group's workflow.

In addition to NEES specific cybersecurity concerns, there are certain domain-specific issues that we deal with in our cybersecurity work. These issues include access to TeraGrid resources, protecting against denial of service, finding an appropriate balance between security and usability.

Supporting Group Collaboration

A critical part of the NEES mission is to foster collaborations that enhance both the network's

By simplifying the process of using HPC resources, NEEShub aims to expand the community of high-end resource users who might not otherwise join the community.

value and the research conducted within the NEES sites. It encourages collaborations in several ways. The NEEShub's cyberinfrastructure facilities support and encourage remote collaborations within the earthquake-engineering community using a mix of commercial and open source software. NEES provides access to the WebEx conferencing software, which has proven to be a popular tool in the NEES community. NEEShub also lets users create project groups (including a wiki page, discussion, and email list) and data sharing capabilities—through Web-based Distributed Authoring and Versioning (WebDAV), Secure File Transfer Protocol (SFTP), secure copy (scp), and the SynchroNEES tool—to run on their local computers. By using such group collaboration tools, the NEEShub helps create a virtual sense of "place" that's one of the crucial elements of fostering a community.

Using the power of HUBzero technology through the NEEShub, the NEEScomm team at Purdue has created an effective system that combines data, analysis tools, and collaboration software into a powerful cyberinfrastructure platform for the earthquake-engineering community. Recent events in Japan have only sharpened our focus and deepened our resolve to support the efforts of researchers and practitioners worldwide to develop safer and more resilient structures, and thereby help society better endure and recover from seismic events. CiSE

Acknowledgments

The NEEShub is the result of work by many dedicated IT professionals on the NEEScomm and HUBzero teams at Purdue University, including Michael McLennan, Dawn Weisman, Gemez Marshall, Brian Rohler, Greg Rodgers, David Benham, Gaspar Modelo-Howard, Rajesh Thyagarajan, Shirley Dyke, Standa Pejsa, Jason Lambert, Jerry Rowley, Ian Mathew, Nicholas Kisseberth, Sam Wilson, Beth Schroeder, John Rosheck, George Howlett, and Tanzima Islam. The US National Science Foundation's George E. Brown, Jr. Network for Earthquake Engineering Simulation (NEES) Program supports the work we describe here through grant number CMMI-0927178.

References

1. C. Grandi, D. Strickland, and L. Taylor, *The CMS Computing Model*, CMS Note 2004-031, 2004.
2. L. Brasseur, M. Tamburri, and A. Plueddemann, "Sensor Needs and Readiness Levels for Ocean Observing: An Example from the Ocean Observatories Initiative (OOI)," *Proc. OceanObs' 09: Sustained Ocean Observations and Information for Society*, vol. 2, Congrex, 2010, pp. 21–25; https://abstracts.congrex.com/scripts/jmevent/abstracts/FCXNL-09A02a-1739373-1-cwp4a03.pdf.
3. T. Hey and A. Trefethen, *The Data Deluge: Grid Computing—Making the Global Infrastructure a Reality*, John Wiley & Sons, 2003.
4. I. Park et al., "Towards an Integrated, Web-Executable Parallel Programming Tool Environment," *Proc. Supercomputing*, IEEE Press, 2000; doi:10.1109/SC.2000.10044.
5. M. McLennan, and R. Kennell, "HUBzero: A Platform for Dissemination and Collaboration in Computational Science and Engineering," *Computing in Science & Eng.*, vol. 12, no. 2, 2010, pp. 48–53.
6. B.P. Haley et al., "Advancing Nanoelectronic Devices through Peta-Scale Computing and Deployment on NanoHUB," *J. Physics Conf. Series*, vol. 180, no. 1, 2009; doi:10.1088/1742-6596/180/1/012075.
7. B. Spencer et al., *Cyberenvironment Project Management: Lessons Learned*, tech. report, US Nat'l Science Foundation, 2006.
8. B. Johnson, *Anchorage Detailing Effects on Lateral Deformation Components of R/C Shear Walls*, master's thesis, Department of Civil Engineering, Univ. Minnesota, NEEShub 2010; http://nees.org/resources/235/download/Johnson-MSThesis.pdf.
9. B. Brueggen, "Performance of T-Shaped Reinforced Concrete Structural Walls under Multi-Directional Loading," master's thesis, Department of Civil Engineering, Univ. Minnesota, NEEShub, 2010; http://nees.org/resources/236.
10. L. Van Den Einde et al., "The NEEScentral Data Repository: A Framework for Data Collaboration in Earthquake Engineering," *Proc. 14th World Conf. Earthquake Eng.*, Mira Digital Publishing, 2008, paper 11-0167; www.14wcee.org/Proceedings/files/11-0167.pdf#page=1.
11. J. Bardet et al., "Overview of NEES Data and Metadata Models," *Proc. 13th World Conf. Earthquake Engineering*, Mira Digital Publishing, 2007, pp. 1–6.
12. L. Van Den Einde et al., *NEESit Data Model in Support of Earthquake Engineering Research*, tech. report TN-2007-05, San Diego Supercomputer Center, 2007.
13. R. Kent et al., *Using SingleShot for NEEScentral Data Entry*, tech. report, George E. Brown, Jr. Network for Earthquake Eng. Simulation, 20 Oct. 2008; http://nees.utexas.edu/files/neesUTexas_Using_SingleShot_for_NEEScentral_Data_Entry.pdf.
14. L. Vaquero et al., "A Break in the Clouds: Towards a Cloud Definition," *ACM Sigcomm Computer Comm. Review*, vol. 39, no. 1, 2008, pp. 50–55.
15. F. McKenna, G. Fenves, and M. Scott, *OpenSees: Open System for Earthquake Engineering Simulation*, Pacific Earthquake Eng. Center, Univ. California, Berkeley, 2006; http://opensees.berkeley.edu.

Thomas J. Hacker *is an assistant professor of computer and information technology at Purdue University and coleader of IT for NEES. His research interests include cyberinfrastructure systems, high-performance computing, and reliability of large-scale supercomputing systems. Hacker has a PhD in computer science and engineering from the University of Michigan, Ann Arbor. He's a member of IEEE, the ACM, and ASEE. Contact him at tjhacker@purdue.edu.*

Rudi Eigenmann *is a professor in the School of Electrical and Computer Engineering at Purdue University and coleader of IT for NEES. His research interests*

include compiler optimization, programming methodologies and tools, performance evaluation for high-performance computers and applications, and Internet sharing technology. Eigenmann has a PhD in electrical engineering and computer science from ETH Zurich. Contact him at eigenman@purdue.edu.

***Saurabh Bagchi** is an associate professor in the School of Electrical and Computer Engineering at Purdue University and the Cybersecurity officer for NEES. His research interests include dependable distributed systems, network security, reliable high-performance computing, wireless mesh, and sensor networks. Bagchi has a PhD in computer science from the University of Illinois at Urbana-Champaign. Contact him at sbagchi@purdue.edu.*

***Ayhan Irfanoglu** is an assistant professor of civil engineering at Purdue University. His research interests include earthquake engineering, structural health monitoring and capacity estimation, and large-scale simulation of structural response to dynamic loads. Irfanoglu has a PhD in civil engineering from the California Institute of Technology. Contact him at ayhan@purdue.edu.*

***Santiago Pujol** is an assistant professor of civil engineering at Purdue University. His research interests include the seismic vulnerability of existing structures, displacement-based seismic design, instrumentation and testing of structures, the response of structures to impulsive loads, and structural-health monitoring. Pujol has a PhD in civil engineering from Purdue University. Contact him at spujol@purdue.edu.*

***Ann Catlin** is a research scientist at Purdue University's Rosen Center for Advanced Computing. Her research interests are in application design and development, mathematical software, programming environments, and real-world applications of computing technology. Catlin has an MS in mathematics from Notre Dame University. Contact her at acc@purdue.edu.*

***Ellen Rathje** is the Warren S. Bellow Centennial Professor of Civil, Structural, and Environmental Engineering at the University of Texas at Austin. Her research interests include understanding and predicting geotechnical earthquake hazards, with particular emphasis on the evaluation of earthquake-induced ground failure (slope instability and soil liquefaction), the effect of soil conditions on earthquake ground motions, and the use of remote sensing to document earthquake hazards. Rathje has a PhD in civil engineering from the University of California at Berkeley. Contact her at e.rathje@mail.utexas.edu.*

Call for Papers

Scientific Computing with GPUs

For *CiSE*'s May/June 2012 issue

Submissions due: 4 September 2011

Graphics processing units (GPUs) aren't just for graphics anymore. These high-performance many-core processors are used to accelerate a wide range of science and engineering applications, in many cases offering dramatically increased performance compared to CPUs. Computer architects also use them to build the world's largest supercomputers. However, the use of GPUs in scientific computing comes with an added risk: the effort needed to port applications can be substantial, and not every application benefits from GPU acceleration equally well.

This special issue focuses on the use of GPUs in science and engineering applications. Contributions covering all aspects of using GPUs for solving challenging computational science problems are welcome. Of special interest are articles presenting the results of porting efforts of large-scale scientific applications on large-scale GPU-based high-performance computers.

Copublished by the IEEE Computer Society and the American Institute of Physics, *Computing in Science & Engineering* magazine features the latest computational science and engineering research in an accessible format, along with departments covering news and analysis, CSE in education, and emerging technologies.

We strongly encourage submissions that include multimedia, data, and community content, which will be featured on the IEEE Computer Society website along with the accepted papers.

Submission Guidelines

Authors are asked to submit high-quality original work that has neither appeared in nor is under consideration by other journals. All submissions will be peer-reviewed following standard journal practices. Manuscripts based on previously published conference papers must be extended substantially to include at least 50 percent new material.

Manuscripts should be written in the active voice, should be no longer than 6,000 words (counting each standard figure and table as 250 words), and should follow the style and presentation guidelines of *CiSE* (see www.computer.org/cise/author for details).

Questions?

- For general information about the special issue, contact Volodymyr Kindratenko (include the keyword "GPUs" in the subject line) at kindrtnk@illinois.edu.
- For general author guidelines, see www.computer.org/cise/author.
- For submission questions, contact cise@computer.org.
- To submit an article, go to https://mc.manuscriptcentral.com/cise-cs (log in and then select "Special Issue on GPUs").

www.computer.org/cise