Reconfiguring practice: the interdependence of experimental procedure and computing infrastructure in distributed earthquake engineering

BY GRACE DE LA FLOR1,*, MOBIN OJAGHI2, IGNACIO LAMATA MARTÍNEZ2, MARINA JIROTKA3, MARTIN S. WILLIAMS2 AND ANTHONY BLAKEBOROUGH2

1 Computing Laboratory, University of Oxford, Oxford OX1 3QD, UK
2 Department of Engineering Science, University of Oxford, Oxford OX1 3PJ, UK
3 Oxford e-Research Centre, University of Oxford, Oxford OX1 3QG, UK

When transitioning local laboratory practices into distributed environments, the interdependent relationship between experimental procedure and the technologies used to execute experiments becomes highly visible and a focal point for system requirements. We present an analysis of ways in which this reciprocal relationship is reconfiguring laboratory practices in earthquake engineering as a new computing infrastructure is embedded within three laboratories in order to facilitate the execution of shared experiments across geographically distributed sites. The system has been developed as part of the UK Network for Earthquake Engineering Simulation e-Research project, which links together three earthquake engineering laboratories at the universities of Bristol, Cambridge and Oxford. We consider the ways in which researchers have successfully adapted their local laboratory practices through the modification of experimental procedure so that they may meet the challenges of coordinating distributed earthquake experiments.

Keywords: distributed hybrid testing; earthquake engineering; workplace studies; requirements engineering; e-research; usability

1. Introduction

Over the last decade or so, it has increasingly been claimed that scientific research is undergoing a major transformation (Taylor 2001; Foster & Kesselman 2004). Advances in computing performance and network infrastructure provide the potential for equipment, applications and databases to be shared across multiple locations, thus creating the potential for the emergence of new forms of scientific research (Atkins et al. 2003). e-Research infrastructures aim to establish a reorientation across academic disciplines towards a model of collaboration that more closely reflects activities undertaken in ‘big science’ (Finholt 2003; Welsh et al. 2006). The term big science (Weinberg 1961; de Solla Price 1963)

*Author for correspondence (grace.de.la.flor@comlab.ox.ac.uk).

One contribution of 15 to a Theme Issue ‘e-Science: past, present and future II’.
appeared in the early 1960s and was coined to describe research activities that require large facilities, large budgets and large research teams. For big science projects to succeed, they require the expertise of multi-disciplinary teams that include engineers and researchers working together to address emerging scientific challenges through the design of technologies and instruments.

Emulating the big science model through the implementation of large-scale, multi-institutional, multi-disciplinary computing infrastructures, e-research has the potential to facilitate and increase collective information sharing and collaborative activities. However, in order to realize this vision, widespread adoption within disciplines is necessary. This requires that e-research systems do not cause major disruption or have a negative impact upon everyday work activities. To ensure success in their embedding, uptake and use, e-research project members must develop an understanding of what researchers do in their scientific work, what materials and resources they require and how they collaborate and interact with co-workers (Jirotka & Wallen 2000; Procter et al. 2006; de la Flor & Jirotka 2009; de la Flor et al. 2010).

Considerable research within computer supported cooperative work (CSCW) has been undertaken that analyses in detail how collaborative activities are accomplished within both co-located and distributed environments. Many of these studies investigate the use of novel technologies in real-world settings and focus on understanding how local work practices might be supported and how distributed activities might be coordinated among remote participants. Four key areas of CSCW research relevant to e-research include designing systems that support work in hybrid spaces where real-world and virtual activities need to be aligned in order to establish common understanding among remote participants (Kuzuoka et al. 2004; Luff et al. 2006); coordination and control of distributed activities in which participants rely upon monitoring aggregated data from visual displays and graphs while simultaneously taking account of local observable events (Suchman 1993; Goodwin & Goodwin 1996); support for awareness across locations where understanding the activities of remote participants provides the context for concurrent local activities (Dourish & Bellotti 1992; Heath et al. 2002); and articulation work where a project’s communication and decision-making activities including its plans and procedures, organizational structure and technologies used to manage multi-disciplinary work are investigated (Gerson & Star 1986; Schmidt & Bannon 1992; Jirotka et al. 2005).

We draw upon this body of work to investigate the ways in which technologies designed to extend the functionality of local scientific equipment have been implemented in practice so that distributed experiments may be conducted. We discuss the challenges and consequences of designing a proof-of-concept e-research system developed to enable shared distributed experiments across remote locations within the domain of earthquake engineering. The UK Network for Earthquake Engineering Simulation (UK-NEES) is a scientific grid network that began in 2006 and initially links three earthquake engineering laboratories at the universities of Bristol, Cambridge and Oxford. It shares similar research goals with the US-NEES project that began in 1999 and includes major facilities upgrades that connect 15 experimental facilities. The main research focus of UK-NEES has been to develop technology that enables a new experimental technique: distributed hybrid testing (DHT). This technique allows for much larger hybrid experiments than would otherwise be possible through the linking
of experimental and computational resources. In this paper, we examine how the proof-of-concept UK-NEES system has extended local practices into a distributed setting and transformed experimental procedure to support and enhance scientific experimentation and research.

2. Scientific instrumentation

The relationship between scientific instruments and technical methods of experimental science is one in which research questions are inextricably linked to technologies used to observe, produce and capture data from phenomena (de Solla Price 1984; Rosenberg 1992). This coupling of science and technology has produced equipment and instruments that facilitate the extension of embodied perception (Lynch & Woolgar 1990), enabling the acquisition of new kinds of data and new knowledge that were inaccessible prior to the instrument’s appearance. The design of experimental instruments and improvements to the existing ones are, in most cases, conceptualized and constructed by researchers working in the field within a specific research area (von Hippel 1976). Because researchers have the in-depth knowledge necessary to recognize what new devices and processes might support and enhance their research, they are more successful than manufacturers in identifying requirements, developing design ideas, building prototypes, testing their applicability in the field and finally diffusing the innovation into the research domain. Described as a ‘bottom-up’ process, it is considered the most effective way of embedding technological innovations into research communities, through its repeated demonstrable usefulness and through word-of-mouth (von Hippel 1976). In our case study, we chart the course of this bottom-up process and describe how scientists develop a prototype for a new scientific instrument or meta-instrument that links laboratory equipment across three earthquake simulation laboratories. By linking these laboratories, local capabilities are extended, allowing researchers at one location to gain access to and share the resources of other geographically distributed laboratories so that they may conduct joint experiments. Having access to this new meta-instrument will extend the kinds of phenomena available for observation and measurement.

3. Research practices in earthquake engineering

Earthquake engineers study the response of structures subject to seismic action for the purposes of developing technologies that improve seismic resistance. In this domain, there remains a pressing need for large-scale experimentation that captures non-linear and rate-dependent behaviour of structural components in order to give confidence in new anti-seismic technologies. Various experimental techniques exist, the simplest of which apply ramp or sinusoidal displacements on the test specimen. However, the more realistic the reproduction of true seismic loading on a specimen, the greater the confidence that the findings represent true seismic performance.

Seismic testing of structures is often carried out at large shaking table facilities, in which the structural system, or more often a scaled representation, is mounted upon a table and can be actuated in real time (i.e. where real earthquake duration equals the simulation time) to simulate seismic ground motion. Although shaking
table testing is arguably the most realistic technique, capturing both global and local seismic responses physically, it is expensive to implement at large scale. Additionally, smaller scale tests can be difficult to extrapolate to prototype scale, and tests can be difficult to control accurately.

An alternative seismic testing technique, hybrid testing, bridges the gap between fully physical and fully numerical testing. This is achieved by separating the structure of interest into numerical and physical parts or substructures. Here, only key parts of the structure, for example a novel seismic-resistant device and its attachments, are tested physically and coupled to an adequately representative numerical model of the rest of the structure (not the main focus of the test). Substructures are coupled together via a transfer system of actuators and sensors with an earthquake simulated through the entire structural system. The test proceeds in time-stepping fashion and is continuously loaded with, typically, interface displacements calculated and derived from the numerical part, applied to the physical part and interface forces measured and fed back at the end of

*Phil. Trans. R. Soc. A* (2010)
each time step to allow calculation of the next interface displacement. In this manner, the complete structure’s seismic response is determined while focusing experimentally on the performance of the physical part, the technology under development (figure 1).

Hybrid testing has advanced in recent years from original pseudo-dynamic tests (Mahin & Shing 1985), in which experiments take place over an extended time scale. With a need to capture rate-dependent behaviour, real-time tests have been developed that use dynamic actuators (Blakeborough et al. 2001). Hybrid testing is a more cost-effective seismic testing technique and is less affected by actuation capacity limits, though requiring sophisticated control, since only the part of the structure of main interest is tested physically.

In answer to capacity limits and scaling issues associated with shaking tables, the E-defense table was commissioned in Miki, Japan, in 2005. As the largest shaking table in the world, it can subject an entire medium-sized building to realistic earthquake shaking. However, with a growing requirement for testing ever larger and complex structural systems at full scale (Nakashima 2008), even E-defense has significant capacity and cost limits in testing larger structures, such as those represented by figure 1. As an alternative, hybrid testing, particularly in real time, is capable of simulating the seismic response of larger structures. However, in practice, as the number and size of the physical substructures increase, it can quickly saturate the testing capability of any one laboratory. Taking advantage of the principles of hybrid testing and the expansion of networking infrastructure, a new concept, DHT, can extend the local hybrid testing technique to address many of the limitations within local laboratory settings.

4. Distributed hybrid testing

DHT aims to address local laboratory capacity issues by allowing the numerical portion of an experiment to be coupled to multiple geographically distributed substructures over a computer network, thus forming a much larger structural system. DHT is a new and novel approach to earthquake experimentation that may eventually make it possible for researchers to take advantage of complementary facilities and specialist technical expertise across distributed locations, so that experiments may be carried out that have either never before been possible or are larger than has previously been possible (Ojaghi et al. 2007). Although extended time-scale tests have been demonstrated, particularly by US-NEES (Spencer et al. 2004), the technique introduces new and prohibitive obstacles that remained to be solved. One of the main research themes of UK-NEES and the focus of work at the Oxford node has been to develop new algorithms and software, enabling DHT at rates up to and including real time. The project successfully demonstrated its capabilities in a series of two- and three-site distributed experiments conducted in 2009. An example of a three-site prototype evaluation test is shown in figure 2 (adapted from Ojaghi et al. 2010).

Particularly in achieving robust, stable and accurate two-site hard real-time distributed hybrid tests, UK-NEES was successful. However, many new challenges emerged in the development process. For example, control of distributed experiments is far more complex than equivalent local tests because robust
connections between existing legacy hardware are required. Additionally, the distributed environment makes it necessary to overcome network delays, data loss and issues concerning time-step synchronization. Local testing control issues also need to be attended to, such as actuation performance and sensor noise. These are all to ensure that the system is capable of reliably completing each time-step loop as well as it would in an equivalent local test. Overcoming these technical issues has been essential to conducting an accurate test. However, the new distributed testing environment posed another very different challenge: developing a way for remote researchers to manage the coordination and control of distributed experiments. To achieve this, local procedures must be reconfigured, including the ways in which experiments begin, are monitored and terminated.

5. Transitioning local laboratory procedures into a distributed environment

Although there have been many technical challenges involved in linking together remote laboratories, embedding this new computing infrastructure into research settings has also exposed a key usability challenge: supporting the coordination and control of distributed experiments as they occur often in real time.
Earthquake experiments are stressful not only because researchers are working in a safety critical setting, but also because test procedures can be complicated and unforgiving of error. In such an environment, an incorrect test control or test procedure will destroy unique or expensive physical specimens; add to this the likelihood of loud mechanical equipment, high temperatures and even the possibility of hydraulic fumes.

There are also spatial and ambient properties to consider. In distributed experiments, those local properties are lost to remote participants. Common graphs that display sensor measurements can ‘tell you what’s happening’ only up to a point. Visual inspection, in conjunction with graphs, provides complete access to a physical specimen’s behaviour, including indications of yielding and structural failure. Additionally, audio features of the local setting are lost at remote locations where sounds suggest breaking and observational notes are taken at the time of cracking. Sounds may also indicate how smoothly mechanical components are operating. Also, ambulatory checks around the test rig are made to ensure that undesirable movement does not occur.

Transitioning experiments into a distributed setting requires that information be conveyed and activities coordinated across remote laboratories, such as powering up equipment, monitoring events and communicating observations. Additionally, mechanisms need to be developed that stop the experiment at each location. These activities are crucial, especially in a real-time setting in which experiments may take place for a brief 1 min duration. Researchers must ensure a correct test start and pay careful attention to graphical and environmental information during an experiment to be satisfied with its outcome and to plan the next course of action. To provide support for coordination, project members designed communication channels for researchers so that they can maintain awareness of concurrent multiple activities as they occur across remote locations. Having this capability can provide an understanding of the overall context of distributed activities and is therefore crucial for synchronizing experimental procedures.

The project met these challenges by designing two easy-to-use applications that link the technical aspects of an experiment, such as control algorithms and hardware (the background process), with the ability to coordinate activities and to ensure that the correct procedure is followed (the foreground process). The aim of this strategy has been to remove as much complexity as possible from the test procedure, so that researchers can concentrate on the earthquake engineering aspects of an experiment.

As part of the design, a client/server architecture was created, assigning the lead laboratory as client, allowing it to maintain control of both the numerical model and the overall test. Laboratories supplying additional physical specimens are assigned as servers that share but maintain control over their local equipment (Ojaghi et al. 2010). This configuration has enabled tighter coordination and control of distributed experiments. In what follows, we present the reconfiguration of practices in one laboratory to illustrate its impact on social interaction and software usability. First, we describe the established procedure of local hybrid testing. Second, the new procedure used to coordinate DHT is presented. Finally, we discuss how the successful implementation of distributed experimental practices has also reconfigured the local test procedure.

*Phil. Trans. R. Soc. A* (2010)
(a) Established procedure in the local laboratory setting

Commonplace in laboratories of this type, legacy systems exist and hardware has been adapted so that equipment can work together. Developing technical workarounds is expected in most laboratories with each designing its own solutions. Workarounds ensure operability; however, sometimes, systems may not be optimally integrated. In one case, the procedure for local testing was awkward, requiring the use of two interaction devices (computer mice) simultaneously or in quick succession to start an experiment. This was a local workaround for integrating two systems: the mechanical testing equipment and the machine running the numerical model. The procedure allowed for an acceptable start to an experiment with no conflict between commands from the time-stepping test controller and the actuation system and so was considered good enough for local testing even though it was awkward for the user.

Early on in the design of the distributed system, it became clear that this procedure would not be robust enough because synchronization of remote time steps would need to be accurately controlled to ensure both reliability and repeatability. As a result of complex technical solutions that synchronize remote equipment, a new procedure was developed. From a technical perspective, the numerical model, mechanical devices and the recording of data were better integrated, and, from a usability perspective, mouse actions were integrated, systems placed on standby and a simpler procedure introduced.

(b) Adapting local procedure for the distributed environment

A local hybrid test has certain key procedural components that ensure successful execution and data capture. When transitioning to a distributed environment, it is critical to ensure that these procedures are properly reconfigured. For instance, it is no longer possible for one researcher to oversee every aspect of an experiment. Instead, a team of researchers must rely on each other, sharing their expertise to ensure that experiments run properly and that results are relayed and recorded. In a local experiment, the most critical aspect is the start, where incorrect settings or incorrectly followed procedure will result in test failure. During an experiment, researchers typically make observations to determine its quality and the significance of its results. At the end, researchers are concerned with where data will be saved for later analysis and making sure that systems are shutdown safely.

In response to the loss of ambient properties at remote locations during the start, duration and end of a distributed experiment, two new software interfaces have been developed that guide researchers through coordinated procedures: a command line interface designed to direct activities (figure 3a) and a visual ‘traffic light’ (figure 3b) designed to display the state of an experiment. These new applications support the coordination of distributed events by both directing the activities of researchers and displaying the state of the hardware and hence the experiment. They provide better integration between systems and are designed to enable not only the synchronization of machines but also the coordination of human activities, and, as a result, researchers’ activities have been reconfigured and streamlined.
Using software that supports the new procedure, remote researchers maintain awareness of each location at a high level of detail. In a distributed experiment, the client initiates it by invoking each server, synchronizing services and data communication between locations through complex algorithms. Although invoked automatically, researchers, nonetheless, require confirmation during this activity, and the traffic light display provides colour-coded indicators. Once each location is connected, the visual display changes on all systems from red (stop) to amber (pause). Upon connection, the server sites are prompted through the command line interface to place their actuation systems on standby and to confirm laboratory readiness.

Even though connections are invoked and time steps synchronized automatically, confirmation of laboratory readiness requires that researchers manually indicate such a state. This is necessary because invoking successful connections does not imply readiness. Rather, readiness is a condition determined at each local laboratory in which researchers conduct final safety checks and secure the experimental area. Once remote researchers are satisfied with local conditions, they confirm through the command line that their actuators are on standby and also indicate that they are ready for the experiment to begin. Upon receipt of all remote server confirmations, the client is prompted to place its local actuation systems on standby and, when ready, to start the distributed experiment with a single final command. On receiving the start signal, all sites begin actuation and data communication, setting the state of each local system as green (run), indicating that the experiment is active. At the end of the test, the state of all sites returns to red (stop) automatically, placing actuation back on standby.
to standby and confirming the status of each server site to the client. Data are saved automatically or the researcher is prompted to do so, and each site having been made aware of the test end can reset for the next test.

The synchronization of services and equipment, although automated, requires confirmation, in this case, through a visual display which verifies that connections are successful. In a distributed hybrid test, scientific equipment at several local laboratories become components of a new meta-instrument synchronized using complex algorithms. Successful test start is determined by software and hardware synchronization activities; however, this also implies local readiness, which is related to both participant and test-rig readiness and safety issues. Although a federated process, each laboratory maintains control of when an experiment may actually start at its location as researchers still need to maintain this control in a similar manner as they do in local experiments. Only now, that decision is multiplied dependent on how many other laboratories take part in a distributed experiment. This added complexity has been addressed with the introduction of the command line interface, enabling researchers to indicate their readiness and thus coordinate activities across locations. Both these new software tools are essential to the success of coordinating and executing the new distributed procedure because they provide information about both the synchronization of equipment (traffic light display) and the readiness of local laboratories (command line interface).

Both work together to convey information about the state of the system at the hardware level, giving confidence that connections have been made, and at the coordination level, giving indications of readiness at each laboratory. These complementary tools provide researchers with the ability to orchestrate activities as well as to maintain a sense of how a distributed experiment is progressing, albeit at a very high level. Most importantly, the new software designed to support a distributed procedure allows researchers to focus on events occurring at their location and less on ancillary activities associated with orchestrating experiments. Both these tools were designed, as one researcher explained, to give remote participants an indication of ‘what to do, when to do it and what’s going on’. Once an experiment is running, graphical data representing command and sensor signals are referenced to determine test performance. After an experiment, local and remote data are analysed to determine the significance of the experiment and test quality. The new procedure, although disciplined, is flexible, allowing researchers to decide when an experiment should begin with an indication of readiness. With these new features developed, the prototype was evaluated in order to beta-test its robustness and to identify how well routine work activities were supported.

(i) Prototype evaluation

The UK-NEES prototype evaluation tests were conducted to gain a better understanding of the robustness of the connection between two and later three laboratories and to refine both technical and non-technical procedures. The evaluation focused on links between distributed hardware and software, maintenance of a robust and reliable connection and the ease of coordination and control among remote researchers. Here, we focus on an evaluation of the two interfaces described earlier and the activities they were designed to support.
The study took place over 10 months, during which in-depth interviews were conducted with participants, who discussed the variety of artefacts and materials used in their work. Additionally, a series of video-recordings of focused activities were gathered in situ, where researchers worked both individually conducting local hybrid tests and collaboratively conducting distributed hybrid tests using the prototype system. We draw upon ethnomethodology (Garfinkel 1967) and conversation analysis (Sacks 1992) to analyse the materials collected. The aim of the analysis has been to understand the details of the skills and practices that participants draw upon to do their research.

Initially, we found significant differences between two- and three-site tests, in which some remote researchers were familiar with the distributed procedure and others less so. In this case, increasing the size of experiments from two to three sites proved more difficult to manage. For instance, one server site (Bristol) is well versed in the test procedure, and so the need to coordinate activities through vocal conduct was reduced. At the second site (Cambridge), vocal conduct was crucial because the researchers were initially unfamiliar with the testing technique and procedures. In addition, establishing local control was challenging because of a lower level of integration of their legacy systems. This placed an increased burden on the client site (Oxford), resulting in occasional failure owing to operator error because of the overwhelming procedural responsibility placed upon them. As the remote site at Cambridge gained experience in the test method, the burden was reduced and the tools in place served their purpose, leading to successful experiments, with more emphasis placed on observing physical specimens as opposed to the test procedure.

Interestingly, during the prototype evaluation, we found that activities were coordinated through vocal conduct, which was essential particularly in troubleshooting the non-automated aspects of experiments, specifically activities such as conveying the fine-grained details of events and ad hoc contingencies. During the evaluation, experiments occurred in quick succession and parameters varied greatly between tests. Even though remote researchers followed the prompts given by the command line interface and monitored the visual display, they remained in constant telephone communication. The following fragments reveal how activities are coordinated between the two laboratories,\(^1\) and this first one reveals how vocal conduct allows researchers to work through contingencies, moving seamlessly from one activity to the next (figure 4).

```
Oxford: I'm ready to go so you can just press play
Bristol: You're going to have to give me a moment
Oxford: ((gives researcher parameters to put into system))
Bristol: ((researcher enters in parameters))
	Okay ready when you are
Oxford: Okay, I'm ready to go (. ) Start the server when you're ready
Bristol: Server started
```

\(^1\)Guide to notation—pauses in speech are calculated to the nearest 0.2s, with (. ) equalling a maximum of 0.2s. Numbers within parenthesis indicate this elapsed time. Descriptions of local activities are given in italics within double brackets (( )). Indiscernible words in single brackets (adapted from Sacks et al. (1974)).
In this fragment, each researcher coordinates activities in a manner they could not if using the command line interface alone. The command line prompts researchers to perform and confirm two high-level activities: standby actuation and readiness. However, here, researchers are manipulating the details of variables within the actuation control software requiring more complex coordination. After the parameters are entered, the researcher at Oxford requests that the researcher at Bristol start his server and he confirms this. Such vocal conduct appears important in facilitating contingent activities where situations arise that do not follow the pre-scripted procedure. Here, talk is used to make sense of activities that the two software interfaces cannot convey and helps to coordinate activities at a fine-grained level of detail; in this case, setting the correct parameters across locations in order to start the experiment.

In the next fragment, researchers evaluate the performance of the network connection between two laboratories:

Bristol: I’m receiving a command signal
Oxford: Hold on (3.3) it’s a bit jerky
Bristol: Seems very (.5) yea step-like
Oxford: Going up a little bit (2.3) still manageable but
( ) millisecond delays ( ) drop down again
went up again (2.0) Yea, we’re looking at ( )
Bristol: Very high delays

Here, researchers discuss the command signal sent from Oxford to Bristol. They both have access to similar graphs, which convey information about the network connection and discuss its performance agreeing that what they are observing are ‘very high delays’. The recorded data are saved to a file to be reviewed later to discover what may have produced this result. Integral to distributed experiments is the ability to monitor the efficiency of data transfer, and this is achieved through monitoring these graphs while simultaneously providing commentary of what is observed. Although the traffic light display provides the ability to track the status of client/server connections at a glance, it does so using three high-level
indicators (start, running and stop). However, in this case, the researchers need to convey more fine-grained observations in order to evaluate how well the test is progressing. In this way, researchers must align their attention towards these graphs, from their separate locations, in order to both monitor and comment upon unfolding events.

Through vocal conduct, the researchers coordinate activities using both local (actuation control software) and shared (the network graph) interfaces. Although the command line interface does facilitate some textual communication, it is intended to communicate high-level procedures rather than facilitate ad hoc discussions relating to other aspects of an experiment. What is significant here is that vocal conduct allows researchers to recover information that the two software interfaces cannot convey, which is crucial to successfully completing the distributed experiment. Although researchers have experimented with other systems to facilitate communication, these proved either problematic or lacking significant benefit. For example, instant messaging actually distracted researchers away from crucial activities because it requires focused attention to typing and reading. Although video and audio links have potential benefits, they did not seem to significantly enhance audio-only connections. Additionally, these might affect test robustness by risking reducing network performance. Although a communication system was discussed incorporating, for example, video conferencing through dedicated network connections and a customized messaging system maximizing the benefits of such technology, it fell beyond the scope of this stage of the project.

The prototype evaluation provided important points to consider for future development, particularly for the manner in which the system might support remote participants as they convey local activities and discuss experimental observations. In considering the integration of communication technologies, the ability to discuss fine-grained details and to address ad hoc contingencies is crucial. Activities we observed, which required this level of detail, include initiating an experiment through confirmation of participant readiness; changing experimental parameters; confirming data transfer including receipt of commands; and commenting upon an experiment’s progress. Such communication can recover information necessary to conduct distributed experiments including discussion of experimental details such as test plans and pre-test variables and assisting and directing others with test procedures. Overall, procedural control of a multi-site experiment was successful using the software developed. Both the command line interface and the traffic light display in conjunction with vocal conduct provide differing levels of detail, ensuring that tests are effectively coordinated and controlled. Additionally, the new distributed procedure has replaced the awkward two-mice controlled local experimental procedure in one laboratory.

(c) Introduction of a revised local procedure

An unexpected outcome of the UK-NEES project is that software designed to support distributed experimental procedures has been adopted for use in local laboratories because it provides better integration between the same two systems, hydraulic equipment and the numerical model controller, used in the local hybrid experiments. Additionally, the introduction of a simplified, guided procedure
and visual confirmations of hardware synchronization reduce procedural errors. The improved local procedure provides confidence that systems are working as expected and allows researchers to focus on the science rather than on implementing technical workarounds.

6. Conclusion

Developing software to support new procedures required to conduct distributed experiments has focused on designing mechanisms that enable remote researchers to coordinate multiple activities across geographically distributed locations. To achieve this, two software applications were designed: one directing the activities of remote researchers through command line prompts and the other displaying the state of experiments through a traffic light display. Support for the coordination of activities between remote researchers has been essential for the success of distributed experiments. Even though verbal communication is extensively used to assist in the coordination of activities at a fine-grained level of detail, researchers would like to reduce the need for telephone communication by improving software functionality. Particularly as the number of sites participating in distributed experiments increases, verbal communication across them may become more difficult.

The development of these two applications has been central to supporting the coordination of distributed activities, and the new procedures introduced have been essential to ensuring the usability of the UK-NEES e-research system. Real-time distributed collaboration involves the coordination of complex activities, some of which cannot be conveyed through software that displays high-level descriptions of activities. In an already stressful environment, researchers must attend to multiple interfaces and systems while simultaneously monitoring equipment and noting down experimental observations. When extending these practices into a distributed setting, project members need to consider the manner in which transformations in procedures and work arrangements may be reconfigured to both support and enhance scientific experimentation and research.

The authors wish to thank the EPSRC for funding both the UK-NEES project (grant no. EP/D079101/1) and the Embedding e-Science Applications—Designing and Managing for Usability project (grant no. EP/D049733/1). We also wish to thank past and present members of the UK-NEES team.

References


de la Flor, G. & Jirotka, M. 2009 Blurring the distinction between software design and work practice. In Proc. 2nd Int. Symp. on End User Development, Work-in-Progress Session, Siegen,

Phil. Trans. R. Soc. A (2010)
Reconfiguring practice


Ojaghi, M. et al. 2010 UK-NEES—distributed hybrid testing between Bristol, Cambridge and Oxford universities: connecting structural dynamics labs to a geotechnical centrifuge.


