Experiences with a Hybrid Architecture for Remote Laboratories

Steve Murray, David Lowe, Euan Lindsay, Vladimir Lasky, and Dikai Liu
stevem@eng.uts.edu.au, David.Lowe@uts.edu.au, E.Lindsay@curtin.edu.au, Vladimir.Lasky@eng.uts.edu.au, Dikai.Liu@uts.edu.au

Abstract - **There is growing interest in the use of remote laboratories to access physical laboratory infrastructure. These laboratories can support additional practical components in courses, provide improved access at reduced cost, and encourage sharing of expensive resources. Effective design of remote laboratories requires attention to both the pedagogic design and the technical support, as well as how these elements interact. We discuss our experiences with a remote laboratory implementation based on a hybrid architecture. This architecture utilises a Web front-end allowing student access to an arbitration system, which permits students to select one of a number of experiments, before being allocated to a particular experimental station. The interaction with the equipment then occurs through a separate stand-alone application which runs on its own virtualized server which the user accesses via a remote desktop client. This hybrid architecture has many benefits, as well as some limitations. For example, it allows rich control and monitoring interfaces to be developed, but also requires students to understand a slightly more complex process for establishing the control. We discuss the reactions to this architecture by different cohorts of students as well as the extent to which the architecture facilitates evolution and expansion of the laboratories.**

Index Terms – architecture, hybrid, laboratory, remote.

INTRODUCTION

With laboratory work being identified as an important element of undergraduate degree programs in engineering and the applied sciences [1]-[2] and an established ubiquity of telecommunications infrastructure in most areas of the world, there has been a steady increase in the development of remote laboratories over the past few years [3]. There are several motivating factors supporting this including cost, security, reliability and convenience [4]. The earlier era of remote laboratory development saw most effort directed at technical evolution – preoccupations included experimenting with technologies for real-time audio and video streaming in an effort to overcome bandwidth limitations whilst ensuring service quality, and dealing successfully with the arbitration of multiple simultaneous connections to shared online laboratory apparatus and equipment.

To a significant extent, most of these issues have been successfully overcome. Continuous, reliable and high quality services have been maintained for much of the past decade [4]-[5]. This progress has resulted in a shift in the focus of development effort away from technical refinement. Recent trends focus upon enriching the nature of the student interaction (for example, including support for student-student collaboration and student-teacher interaction). In parallel there have been moves towards a clearer understanding of the pedagogical aspects related to conducting laboratory work remotely and indeed a more reflective consideration of the laboratory learning context in general (both conventional laboratories where students are proximate to the equipment they're using as well as remote laboratories) and the place of experiment simulation [6].

REMOTE LABORATORIES IN THE CURRENT CONTEXT

A standpoint advocating that all undergraduate practical experimentation should (or even could) be carried out remotely would be difficult to defend, but remote laboratories do offer some undeniable advantages over the conventional proximate laboratory setting when used in the appropriate context.

Operating costs can be reduced with the equipment and apparatus being held in a physically secure environment with tightly constrained access which limits either intentional or unintentional misuse. This reduction in attrition and "wear and tear" on the equipment, an entrenched characteristic of proximate laboratories, means that more elaborate, expensive and/or delicate experiments can be constructed. This in turn makes possible, student exposure to systems that might not have otherwise been afforded them. The result is that when viewed on a macro scale, more rather than less experimentation by students becomes possible. Additionally, the convenience and flexibility of being able to complete laboratory experiments remotely tends to fit well within the complex lifestyle of the contemporary undergraduate student profile – it is as welcome amongst the student body which is comprised of full-time or part-time "on campus" students as it is with those that are distance-mode. A final advantage which remote laboratories offer as an obvious consequence of their very make up is that beyond the scope of individual institutions, they offer a capability of inter-collegial sharing of expensive laboratory infrastructure and resources [5]. The potential benefits to students are enormous and profound, but it requires a global view if it is to be realised.

REMOTE LABORATORIES IN A LEARNING CONTEXT

It is essential to view remote laboratories as an enabling technology – not simply the use of technology because we can – and this implies we need to understand what it is that we are enabling. With this approach accepted we can ensure that the pedagogical aspects, the educational objectives targeted by laboratory work, will not be forfeited. The most constructive appreciation of the application of remote laboratories can be obtained if there is a conscious effort taken to facilitate the experimental procedures which would be used in a proximate setting to be unaltered when the same experiment is being conducted remotely. In this way, a superficial observation might be stated that compared to conventional proximate laboratories where students all enter a room and stand around benches of equipment and apparatus, remote laboratories are not influencing learning at all – the same experiments can be carried out on the same equipment, with the same students learning the same concepts, with the only difference simply being spatial.

A more realistic assessment though, uncovers some subtleties – in a proximate laboratory, students are usually grouped at an experiment and there are undeniably interactions both intra- and inter-group during the course of conducting the experiment. These interactions can have a significant effect on the learning experience – and hence the learning outcomes which are achieved. There will also typically be a laboratory staff member (teaching assistant, tutor or facilitator of some description) present, and their interaction with the students can be seen to take two forms: "Consultative" – the students attempting to conduct an experiment wish to seek advice and approach the staff member for assistance, and "Interrogative" – the staff member will instigate an interaction as a way of prompting students to consider and reflect (for example, "What are you expecting to achieve by measuring that?"). These sorts of interactions are limited when laboratory work is carried out remotely and are one of the foci of our ongoing investigations. In particular, we are interested in the nature of the tripartite interaction between the technological infrastructure used in remote laboratories (both hardware systems and especially the software environment), the characteristics of the interactions which students experience in using remote laboratories, and the learning outcomes which these interactions support. It is the first of these elements which is the focus of his paper. We consider a particular architecture which we have utilised and how this relates to student interactions and learning.

THE HYBRID ARCHITECTURE

One of the design challenges in the practical development of a remote laboratory is provide a consistent user access facility to what may be a collection of varied experiments. Within the remote laboratory at the University of Technology, Sydney (UTS) there are currently five collections of significantly different experiment apparatus and equipment [4],[7]-[8].

- Microcontroller design (Embedded Operating System Experiment) – Computer Systems Engineering.
- Beam Deflection (Loaded Beam Experiment) – Civil and Construction Engineering.
- Dynamics and Control pneumatics (PLC Experiment) – Mechanical and Mechatronic Engineering.
- Fluid Mechanics (Coupled Water-Tanks Experiment) – Mechanical Engineering.
- Programmable Hardware design (FPGA Experiment) – Computer Systems Engineering.

The current facility is shown in Figure 1. Whilst sharing a common architecture, the specific interfaces and access mechanisms vary. One involves the use of Linux hosted software development tools which are character-based and accessed through terminal sessions, yet provides a web-based output user interface [4]. Others require windows based development tools to be available to the user in order to create control programs for industrial PLCs (Programmable Logic Controllers), and still others have been constructed to present a LabVIEW derived application to the user to manage the testing of control algorithms for coupled tank apparatus models [7]-[8].

FIGURE 1
UTS REMOTE LABORATORY FACILITY

The heterogeneity of the experiment types (and the pre-existing tools which are important to their use) complicates the development of a unified user access system. One common feature amongst the experiments is the multiplicity of apparatus types. For example, there are up to one dozen microcontrollers all of the same type to be allocated to students as they request access to conduct particular laboratory experiments, five PLCs all with identical configuration and attached electromechanical actuators, and three identical sets of coupled tanks fluid level control experiments. A suite of programs running on a main server work in collaboration to allocate apparatus to students from the pool of unused devices, queuing the allocation requests when necessary. This software system (the "Arbitrator") transparently handles a request for a piece of apparatus from a student and their authentication and authorisation, the allocation of a particular piece of experimental apparatus and the connection of the user interfaces presented to the student to the selected device. When a session of use is completed by a student, the Arbitrator reclaims the apparatus, re-initialises it so that it has a state that is healthy for the next usage session and returns the device to the free pool. The Arbitrator directs resource allocation and does so without direct user interaction, however it does not attempt to deal with the differing user interface and client-side requirements.

Many engineering laboratory exercises require specialised software tools to be available to the user. Examples include tools to develop bitstreams for Field Programmable Gate Array (FPGA) devices, and student-edition versions of proprietary tools for constructing PLC programs in ladder-logic. Ordinarily, this would require that a licensed version of the tool at the correct maintenance level be installed by the student on the remote client computer that they are using to carry out a remote laboratory experiment. This is a logistical requirement which is not without cost and complication. It might transpire that the tool is available as an accompanying aid to a textbook, which alleviates acquisition cost – but that doesn't reduce the cost of the requirement that it be installed and correctly configured at any/every computer that the student might use over the period of time that they complete a potentially complex laboratory assignment. As well as installing it on a student-owned computer at their home, they may also then need to install it on a computer they might have access to at their place of work, or on a computer available in their university/college computing centre. These last two environments would almost certainly be controlled by policy preventing the ad-hoc installation of programs by users, and license restriction on the student-edition software might require it be uninstalled from their home computer before an attempt could be legally made to install elsewhere. Finally, the software tool might make demands upon the operating system type and version resident on the computer that the student would be attempting to install the tool on. These administrative encumbrances make attempting to complete laboratory work remotely problematic, which is in conflict with one of the principle goals of remote access laboratory work.

The solution which we have adopted offered an elegant side-stepping of these problems. The approach was to use virtualisation software on the Linux-based remote laboratories servers (in particular, VMware) to set up virtual windows machines running on the remote laboratories server [9]. All necessary operating systems software (of the correct type and release level) was installed into the virtual machines, and any user-level software tools the students might require was also bundled into the remote virtual machines. Upon login and experimental apparatus selection by the student, a virtual machine is started up to run the correct version of the guest operating system and required user programs. A desktop is then exported from this virtual machine by the remote laboratories' server to the client computer used by the student – this graphical user interface (GUI) is then used in an intuitively obvious way by the student, as if it was running on the user's local machine. This technique offers a hybrid approach to systems architecture. In effect, the remote client computer being used by the student acts as a remote interface to a virtual machine running locally on our servers, and this local virtual machine can then interact directly with the laboratory hardware as necessary.

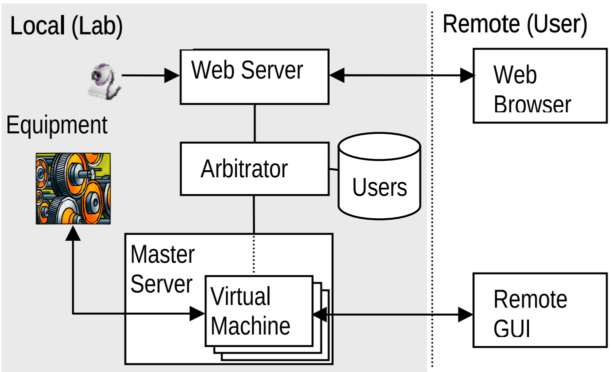

FIGURE 2
UTS REMOTE LABORATORY ARCHITECTURE

The resultant architecture, as shown in Figure 2, comprises an Arbitrator to manage access to laboratory resources, a Web interface to support student interaction with both the Arbitrator and some elements of the interaction with the hardware (such as video feeds of the equipment operation), and the virtual machines used to provide richer access to laboratory-specific interaction software applications when needed. Figure 3 shows the student interface for the Beam Deflection experiment. The left part of the screen shows a Web browser being used to provide video information from the experiment. The right part of the screen shows the interface to the virtual machine running on the local server, and (in this case) running a custom application for controlling the Beam experiment.

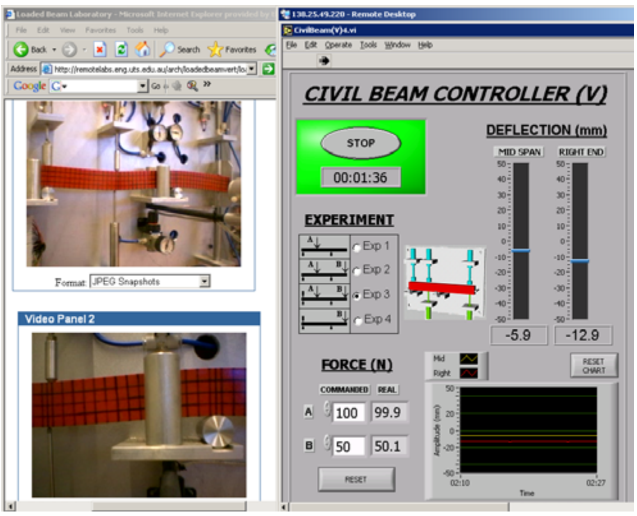

FIGURE 3
STUDENT INTERFACE TO THE BEAM DEFLECTION REMOTE LABORATORY

BENEFITS AND LIMITATIONS OF THE HYBRID ARCHITECTURE

This hybrid architecture offers management and logistical benefits that are easily identified – client side computers have no special software requirements along the lines of application software installation and configuration which increases availability and access to the remote laboratory. The only requirements are a web browser and remote desktop connection software – both of which are supplied standard with recent versions of Microsoft Windows and have comparable alternatives which may be suitable for use on other client operating systems. Apart from these benefits, this architecture makes rich control and monitoring interfaces reasonably easy to implement. Specialised applications can be constructed using laboratory instrumentation toolkits and development environments like LabVIEW. These provide real-time control inputs and a diverse collection of outputs including tables, charts and graphs of physical parameters within the experimental apparatus. These elements can be collected by the experimenter and included within documents prepared by the students as reports for assessment.

One potential negative side effect is a demand that the student be capable of completing somewhat more complex interaction processes (needing to use a Web browser to log-in, be allocated equipment, and then access some elements of the experiment – particularly video feeds – but also needing to use the virtual desktop to access other elements of the experiment). However, carefully constructed user documentation by academic staff along with a demonstration completed by knowledgeable technical staff can reduce the severity of any possible problems to do with user familiarity. A significant bandwidth demand is also inherent in this architecture, but that is less significant with continually evolving hardware and the cost-effectiveness of networking infrastructure.

A cohort of students using a remotely accessible PLC laboratory [7] were surveyed before and after their use of the laboratory at the UTS 2007 Autumn semester (February-June 2007). Of 43 students to use the remote laboratory, 21 were located in and around Sydney NSW and 22 were located in and around Perth, WA. The remote laboratory is located on the Broadway campus of the UTS and the cities of Sydney and Perth are separated by a distance of approximately 3,900 km. A pair of pre-use and post-use surveys were aimed at collecting a variety of observations pertaining to the use of the remote laboratory, but the subset that is relevant in this paper focused on gauging the effectiveness of the hybrid architecture. Of the population surveyed, 79% reported that they felt either “fairly”, “quite” or “very” comfortable with the process of remote interaction with the equipment and apparatus. Possibly the most relevant outcome is that there was no marked difference in responses related to learning between the students in Perth and those in Sydney, but the Perth based students were attracted to the idea of this degree of "remoteness". We would contend that the hybrid architecture is responsible at least in part, for the observation that the greater majority of students were at ease with the process of accessing the remote laboratory and conducting interactive experiments.

Finally, one other feature of the hybrid architecture which is beneficial is that there is no anticipated limits on scalability and flexibility. As more remote laboratories are evaluated and considered for development, it is likely that this architecture will accommodate them. More virtual machines can be created on the servers with each being a container for the user application software necessary to permit the student to complete their assigned laboratory exercises. The scalability as already been demonstrated as we have grown from a single experiment with a small number of stations to 5 different experiments (with several more currently under development) with multiple stations in each case. This growth has not necessitated any significant changes to the core architecture of our system.

CONCLUSION

The hybrid remote laboratory architecture presented here, builds on the many advantages that remote laboratories offer. It represents a robust solution to the problems of management of a varied collection of experiments, yet simultaneously reduces the effort required by the end user to obtain access to the equipment and to the specialized software tools that are required to configure and complete the experiment. Furthermore, it is not likely to introduce a limit with respect to scalability in the context of future expansion.

REFERENCES

[1] Feisel, L. D. and G. D. Peterson, "Learning Objectives for Engineering Education Laboratories". *32nd ASEE/ IEEE Frontiers in Education Conference*, Boston MA. 2002.

[2] Feisel, L. D. and A. J. Rosa, 2005 "The Role of the Laboratory in Undergraduate Engineering Education", *Journal of Engineering Education*, 94(1): pp. 121-130.

[3] Corter, J. E., J. V. Nickerson, et. al. 2007 "Constructing Reality: A Study of Remote, Hands-on and Simulated Laboratories", *ACM Transactions on Computer-Human Interaction*, (14),2, Article 7.

[4] Murray, S. J. and V. L. Lasky "A Remotely Accessible Embedded Systems Laboratory" in Sarkar (ed.) 2006. *Tools for Teaching Computer Networking and Hardware Concepts*. Hershey: Information Science Publishing, pp. 284-302.

[5] "MIT iCampus: iLabs" Massachusetts Institute of Technology. http://icampus.mit.edu/iLabs/default.aspx Accessed: 24 March 2008.

[6] Lindsay, E. D. and M. C. Good. November 2005. "Effects of Laboratory Access Modes Upon Learning Outcomes". IEEE Transactions on Education. Vol. 48, pp. 619-631.

[7] Lasky, V. L., D. K. Liu, S. J. Murray and K. K. L. Choy. "A Remote PLC System for e-Learning", Proceedings of the 4th ASEE/AaeE Global Colloquium in Engineering Education, 26-29 September 2005, Sydney, Australia.

[8] McIntyre, D., D. K. Liu, V. L. Lasky and S. J. Murray. "A Remote Water-level Control Laboratory for e-Learning", Proceedings of the 7th International Conference on Information Technology based Higher Education and Training (ITHET), July 2006, Sydney, Australia.

[9] Lasky, V. L. and S. J. Murray. "Implementing Viable Remote Laboratories using Server Virtualisation". V. Uskov (ed.) 2007. Proceedings of Web-based Education. 14-16 March 2007, Chamonix, France.

AUTHOR INFORMATION

Steve Murray, Academic Coordinator, Faculty of Engineering Remote Laboratory Project, The University of Technology, Sydney, stevem@eng.uts.edu.au.

David Lowe, Director – Centre for Real-time Information Networks, Faculty of Engineering, The University of Technology, Sydney. David.Lowe@uts.edu.au.

Euan Lindsay, Faculty of Engineering and Computing, Curtin University of Technology, E.Lindsay@curtin.edu.au.

Vladimir Lasky, Vladimir.Lasky@eng.uts.edu.au.

Dikai Liu, Faculty of Engineering, The University of Technology, Sydney. Dikai.Liu@uts.edu.au.