GridLabs: Facilitating collaborative access to remote laboratories[1]

Joerg M. Haake, Sven Laaks, Ludger Menzel, Wolfram Schiffmann

FernUniversität in Hagen, Fakultät für Mathematik und Informatik,
Universitätsstr. 1, 58084 Hagen, Germany
Email: {joerg.haake, sven.laaks, ludger.menzel, wolfram.schiffmann}@fernuni-hagen.de
phone: (+49 2331) 987 325, *fax:* (+49 2331) 987 332

Abstract

eScience is usually characterized by the cooperation of distributed groups of researchers who share data and computing environments and perform experiments together. Often immense data sets that were produced by expensive equipments need to be accessed and evaluated. Such eScience scenarios require both, support for collaboration of researchers at distant locations and also the remote control of the shared laboratory devices. However, this type of remote experimentation and collaboration must be taught during university education. In this paper, we propose a framework that supports the training of above practices through the provision of a dedicated collaboration environment. It extends current approaches with support for a life cycle of remote labs, including scheduling the access to remote labs as well as defining access permissions. Our experiences in teaching lab courses suggest that the approach is also applicable in eScience scenarios.

1 Introduction

Experiments are a crucial means for conducting scientific research in many sciences. While eScience is more focused on supporting collaborative research, e.g. through conducting and analyzing experiments jointly, collaborative eLearning emphasizes the benefits of learning together using the computer as a supportive means. In terms of learning experimental work, this means that learners must be able to conduct and analyze experiments jointly, too.

Since experimental equipment is expensive, there is never enough of it available. This is true for research as well as for learning. Current technology provides the opportunity to implement remote labs on a large scale. Such remote labs would be accessible over the Internet and provide researchers and learners with easy access to real hardware and software needed for the experiment at hand. Many benefits are associated with the sharing of remote real hardware lab equipment, such as higher efficiency, lower cost per experiment, increased availability (24 hours per day), increasing number

[1] This work has been partially supported by the German Ministry of Research and Education (BMBF) under grant no. 01PQ05004

of available experiments due to sharing and consequently improved teaching and learning due to access to more resources.

Current support for remote labs can be found in the eScience research area, where special purpose environments (collaboratories [15, 16]) have been developed for specific scientific campaigns, as well as in the eLearning area, where work has focused on shared project work [17] and access to remote instruments [18]. Current approaches are based on application sharing and the provision of additional communication and document exchange services.

Both types of work share a number of deficiencies when aiming at large scale sharing of remote labs for teaching purposes, such as a method for simple construction of remote labs, managing and scheduling access to remote labs, and defining access permissions.

In this paper we present the GridLabs framework, which aims at supporting large scale sharing of remote labs on demand. We show how a combination of a web-based shared workspace environment with remote controlled lab boxes can - together with a respective method - support distributed researchers/learners to get access to the needed laboratory equipment and perform experimental work together.

This paper is organized as follows: Section 2 analyzes the requirements of large scale sharing of remote labs. Section 3 analyzes related work and main deficits. Section 4 presents the GridLabs framework. Section 5 describes the current implementation and section 6 summarizes our experiences with the approach. The paper concludes with a summary, comparison to related work and further research.

2 Requirements of collaborative remote labs

Collaborative remote labs support a number of users in conducting experiments and data analysis by using distant laboratory equipment (see figure 1). For each experiment, a so-called Remote Lab Box (RL_Box) provides the required equipment (e.g., instruments, cameras, light, power) with the configurations (e.g., wiring) specific to the experiment. Each RL_Box must offer an interface for remote controlling of the equipment. Supporting efficient use of the RL_Box requires that the same box should be used by many teams, one after the other. This way, maximum usage can be achieved.

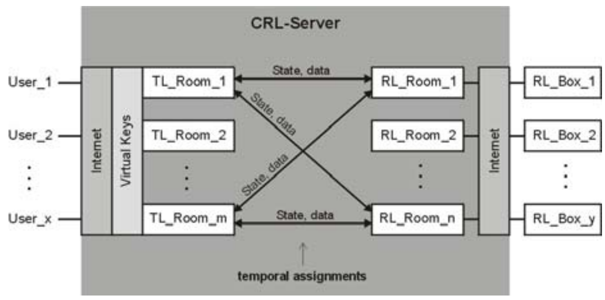

Figure 1: The GridLabs framework of collaborative remote labs

However, this means that each team must have a separate collaboration environment through which they can access and share their state of the experiment. We call this environment a virtual Remote Lab Room (RL_Room). Each RL_Room is, when used, connected to an RL_Box matching the requirements of the experiment. Since RL_Boxes and

RL_Rooms are used in alternating order by different teams, the actual state of the experiment must be kept persistently at a special place for each team. We call this place a TeamLab_Room (TL_Room), which is used to save the status of an experiment and to restore this state when the team, after a break, returns to continue their experiment. Likewise, experimental data need to be recorded and kept persistently in the TL_Room. In order to support joint experimentation, the team members need to be able to control the experiment remotely as well as to communicate about the results, coordinate their activities, and perform joint documentation and data analysis.

Since many teams may compete for a limited number of matching RL_Boxes, some kind of scheduling is needed, which ensures that every team gets a fair amount of time and no resources (RL_Boxes) are left unused (given a sufficient demand). Furthermore, if teams experience problems with the equipment, some sort of support (e.g., tutoring, technical assistance, or manual reset of an RL_Box) is needed. Finally, some support for accounting and billing is needed, as the cost of accessing remote equipment may need to be covered. Even if this is a crucial aspect of large scale sharing across organizations, we cannot cover it due to space reasons in this paper.

The process of providing and using such collaborative remote labs includes the following tasks:

(1) During design of collaborative remote labs, developers need to develop the required types of RL_Boxes and create matching types of RL_Rooms, which contain the matching remote control and status store/restore functionalities.

(2) During set up of collaborative remote labs, the administrator must construct the RL_Rooms and specify the required (type of) RL_Box.

(3) During reservation of collaborative remote labs, the administrator must form teams in TL_Rooms and assign the RL_rooms to users. E.g., the administrator may create time periods for which users may sign up.

(4) During experimentation with collaborative remote labs, users may use the remote control facilities provided by their RL_Room, record data and communicate with each other. A group of users that collaborate in conducting experiments share a so called Team Lab Room (TL_Room) where the state of the experiment and measurement data can be stored. The platform must ensure that only one user of a team can change the state of an RL_Room (i.e. control the experiment) at a particular time, and that the state of this RL_Room is restored before experimentation can continue. The room's access model must support restricted access to its functionality, if different roles (e.g., remote operator, tutor, observer, and analyst) require different access permissions to functionality.

(5) During post processing with collaborative remote labs, users may use the stored data and results in their room in order to jointly analyze the data and to jointly write reports.

An environment for collaborative remote labs must support all above tasks. Ideally, such a system should be based on the WWW, as to allow users simple access through a web browser. In the next section, we examine related work for these requirements.

3 Related Work

Since the early 1990's, a number of collaboratories have been developed, which aim at supporting distributed groups of researchers. A good example is UARC [15], a collaborative software environment supporting distributed researchers in capturing and discussing data from remote instruments.

The number of universities that provide remotely controllable laboratories is increasing steadily. While simulations are cheap and easy to establish they can only model a limited degree of realism. A satisfactory simulation of the dynamic behavior of complex systems requires much computing power and even then it could not accomplish real-time quality. Especially, the simulation of embedded computers is very computational demanding and a simulation of a radio-telescope would still be impossible. Thus, to develop and test the real-time behavior of those systems we have to rely on real hardware instead of simulations.

Remote laboratories allow access to experimental setups that can be operated and controlled 24 hours on 7 days a week. Beside the device that should be investigated, the experimental setups usually comprise several expensive instruments for excitation and measurement of the device's reaction. The spectrum of remotely controllable instruments can range from a quite low budget oscilloscope up to an extremely expensive electron-microscope.

In the recent literature different categories of remotely controllable experiments were described:

- Analysis of the characteristics of semiconductor devices [8]
- Setup of electronic circuits and measurement of their behaviour [5, 1]
- Physical observations like mechanic experiments [10], or analysis of the characteristics of the upper atmosphere [15, 16]
- Different kinds of control systems for studying the parameter setup and analysis of the dynamic behaviour of the control loop [11]
- Communication engineering experiments [4, 18]
- Digital programmable logic [14]

Beside the description of specific remote lab systems also general distance learning and pedagogical issues were addressed. In [13] a framework for analyzing the effectiveness of remote labs is proposed. Cooper [3] mainly focuses on issues that impinge on the specification and design of remote labs. He also identifies the open problems that must be solved to achieve a widespread acceptance of remote labs in education and eScience.

The main deficits of current remote laboratories concern a unified appearance of the instruments' remote control panels, access by remote users, and the support for collaboration among the lab's participants. Only a few papers describe support for user management and toolkits for the experimental setups [8]. In this paper we offer solutions to the deficits cited above. We will present a unified framework that allows not only for lab access, but also for reservation of experiments, conducting the experiments and a collaborative data interpretation and discussion of the experimental results. Moreover, we also show how traditional instruments can be made network-enabled and how easy-to-use control panels can be realized.

4 Approach

The GridLabs framework (see Figure 1) allows multiple users to collaboratively perform an experiment on real hardware (i.e. RL_Box) anywhere on the Internet via a shared lab workspace (i.e. RL_Room) on the Collaborative Remote Lab (CRL) server. RL_Rooms provide remote experimentation facilities and are temporally assigned to TL_Rooms, which serve as a persistent group workspace of a team of learners or researchers.

The main idea of CRL is to integrate collaboration support with remote access to lab equipment. CRL provides virtual rooms (such as in CBE [12] and CURE [6, 7]). Each virtual room is a shared workspace supporting data sharing, communication and coordination within its group of users. In CRL these virtual rooms are augmented with real hardware through connected remote labs (RL_Boxes). Reservation and access management for rooms are provided to facilitate access to remote labs.

First, we will introduce the collaborative platform CRL (Collaborative Remote Laboratories) that provides the necessary support for web-based collaboration. Then, we demonstrate how this platform facilitates the access to real laboratory equipment (the RL_Boxes), followed by a discussion of how the approach supports the process of providing and using collaborative remote labs.

4.1 Collaboration support

We use the virtual room metaphor to structure shared remote labs by providing a virtual laboratory building consisting of virtual labs. Each room is accessed through a web browser (see Figure 2). For this purpose, we create the RL_Room type. Each instance of RL_Room represents a virtual lab and supports communication among its users through threaded email (asynchronous) and chat (synchronous, see bottom of the screendump in Figure 2). Both media are persistently stored and allow later access.

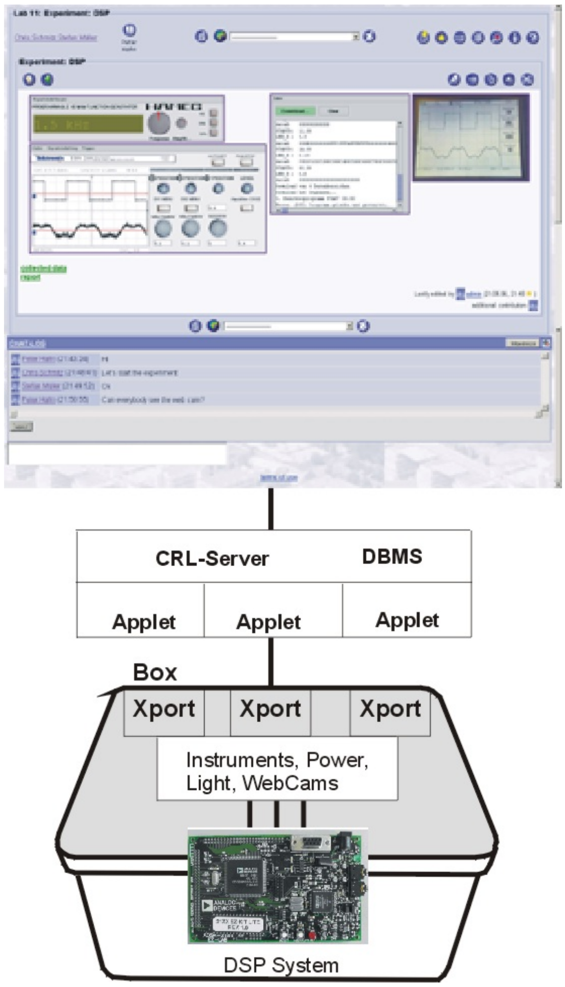

Figure 2: A DSP experiment set-up

Each room provides so-called pages as a means to share persistent data. Two types of pages are supported: content pages are directly editable in the room via a simple WIKI syntax, while binary pages allow upload/download of binary documents. In order to provide access to a remote lab (RL_Box) we created a new type of page in a room containing the remote control features (see Figure 2).

In order to specify access permissions for rooms (labs) we use a virtual key metaphor. A key specifies access permissions of the key holder. Access permissions may include manipulation of room properties (such as keys), access and manipulation of its content (pages), and its communication channels. Keys can also be used to form groups: all key holders of a room form its user group.

4.2 Remote control of experiments

Remote labs require the provision of remote control of real instruments from a distance. In our approach, we provide the remote control functionality as applets contained in an RL_Room of the CRL shared workspace environment. These remote control applets communicate with the real instruments contained in an RL_Box over the Internet. Unfortunately, most of the common laboratory instruments like function generators, oscilloscopes or digital multi-meters are not equipped with a network or web interface. Instead, usually serial interfaces (RS232) are provided. Therefore, we need a device that can act as a bridge between these two different interface standards.

Lantronix's XPort [19] provides an embedded Ethernet device server that can easily be used to implement bridges to any electronic device with serial capability. Beside the bridging functionality there is also a built-in web server. XPort uses a 16 Bit CPU which is clocked at 48 MHz (cf. Figure 3). It is equipped with 256 KByte RAM and 512 KByte flash memory. 384 KByte remain for user data and Java applets while the rest is used by the XPort's operating system and the server applications (bridge and web server). By means of a device installer or a DHCP-protocol the IP-address can be configured. As soon as the IP-address is assigned other configurations can be done through a web-front end. Java applets and html pages are uploaded via the TFTP-utility.

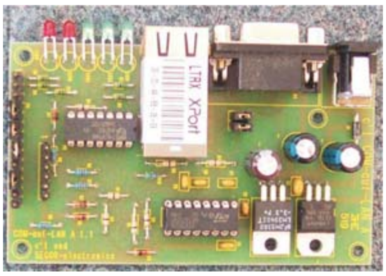

Figure 3: Lantronix's XPort provides a LAN-to-serial bridge and a web server

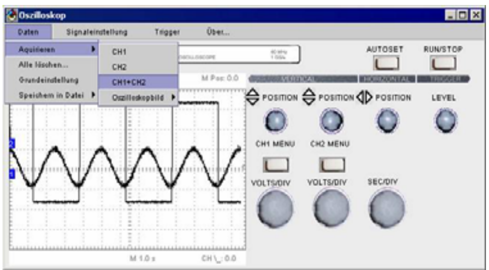

Figure 4: Remote control applet for the Tektronix TDS 210 oscilloscope

We use the XPort to provide a bridge between the Ethernet LAN-interface and serial interfaces, which are mostly used to access measurement devices. We use the XPort's web server to offer one or more Java applets that provide remote controls to the connected instruments. By means of instrument-specific applets we can provide internet access to the instrument that is connected to the XPort. Users only need a web-browser and the URL of the Java applet. Figure 4 shows a screenshot of a remote control applet (RCA) for the Tektronix TDS 210 oscilloscope. This applet can easily be adapted to other 2-channel oscilloscopes by changing the control word sentences associated with the generic parameter functions (e.g. how to set channel amplification or time base). In this way, Java applets for the common laboratory instruments can be used as templates to create RCAs for instruments of other manufacturers. Several device specific control-applets for widespread instruments (e.g. oscilloscope, function generator, multimeter etc.) have been implemented. They can be easily adapted to instruments from different manufacturers.

The access to the instruments by means of a control-applet on a page in the RL_Room is only established if the user owns a valid key with appropriate permissions. Time checking is implemented by the control-applet that gets the time budget at its start up from CRL. When the available time is up any action of the experimenter or observer will terminate the connection to the XPort, which serves as the interface to the respective instrument in the RL_Box. Afterwards, the reality augmented virtual room can be used by other users who own a valid key.

In order to implement remotely controllable experiments we need network-enabled function generators and measurement devices. The signal sources are used to excite the device under test (DUT) and its reactions are detected by the measurement devices. Of course, also the DUT setup must be network-enabled.

Beside the bridging and web server capability an XPort also provides three parallel output signals that can be used to switch between different DUTs or experimental setups. By means of a four-way analog multiplexer that needs two of those signals as control lines, we can choose between up to four configurations that reside in one RL_Box to be presented in the next section. The third output can be used as a reset signal. Of course, we can arbitrarily scale these switching possibilities by additional XPorts.

One may argue that the XPort's functions could also be implemented by one powerful PC that is equipped with multiple serial interfaces and that serves many instruments at once. Even though one can proceed in this way the use of multiple XPorts has several advantages: Due to the fact that they use a flash memory, XPorts can be switched on together with the associated instrument. Shortly after power-on they provide the desired network interface to the instrument. Another advantage is the lower power consumption of the XPorts and the better scaling of the web server throughput. Also with respect to energy saving it is useful to switch on remotely controlled experiments only when they are actually needed. This is much easier to accomplish by means of a dedicated XPort server.

4.3 Sharing of remote labs

We use CRL, a web-based collaboration platform that supports both, mutual exclusive access to the RL_Box and collaboration among small student working groups. To provide an intuitive software interface we use the metaphor of a virtual building that consists of virtual rooms augmented with real experimental equipment. As in real life these augmented virtual reality rooms are connected by floors that can create hierarchical structures. Virtual rooms can only be entered if one holds a valid key. A remote lab building for one or more related experiments comprises several virtual rooms for different purposes:

- main entry room,
- showroom,
- administration,
- TL_Rooms and
- RL_Rooms.

Every student gets a key for the main entry room. In this room the students find detailed descriptions of the experiments, assignments and user manuals of this lab's instruments and DUTs. In the showroom slide shows and tutorial movies explain the procedures for remote experimentation.

In section 2, we identified five major tasks that need to be supported during the life cycle of remote labs. In the following, we present how our approach supports each of these tasks.

4.3.1 Design of collaborative remote labs

When a new experiment needs to be supported, the designers develop a new type of RL_Box containing the appropriate instruments for the experiment. Any new applets for remote control of new instruments need to be developed. A matching new type of RL_Room is then created in CRL, which refers to the appropriate applets for remote control of the instrument. Likewise, the status store/restore functionality needs to include all instruments provided in the RL_Box.

4.3.2 Set-up of collaborative remote labs

During set-up, a supervisor sets up the virtual lab building for the experiment. The main entry room functions as a place for general information about the experiment, for team formation and for reservation of RL_Rooms as well as a hallway to the respective lab rooms for each team. The supervisor creates a TL_Room (a shared workspace for a team in CRL) for each ream. In addition, the supervisor creates for every RL_Box one instance of the respective type of RL_Room. A TL_Room serves as a project room for the group of experimenters for supporting data storage and collaborative work. Users meet in this TL_Room to collaborate. For experimentation, they go to the proper RL_Room, which is connected to its RL_Box containing the needed experimental equipment.

Two ways of group formation are supported: In a formal sign-up, students can enter the administration room and become a member of a team. As a result, they receive a key for the TL_Room of their team from the supervisor. As soon as a team member enters a RL_Room for the first time a corresponding sub-room will be created. In this room the CRL-server stores the current state of the instruments and the students will store there measurement data for later evaluation and discussions. If the same RL_Room is used later by another team member the last state of the instruments is restored and new measurement data can be added. Thus, all the data produced during experimentation is stored in the virtual room associated with each team.

A student can also create a group informally by creating a private room for storing the individual results of laboratory sessions, which are conducted in one of the RL_Rooms. In order to collaborate with other students he can make copies of his personal key and pass the copies to his colleagues. In this way, the private room becomes a (virtual) group room.

4.3.3 Reservation of collaborative remote labs

As RL_Boxes are a scarce resource that can be used only by one group at a time, there is a need for managing the access to the experiments. In CRL this is facilitated via the use of temporal keys.

RL_Rooms can only be entered for a limited time period. During this time a mutual exclusive access must be guaranteed. To accomplish this we generate many virtual key each one valid for a defined time slot. The current time slots are fixed to 60 minutes each. In the administration room a key rack provides all time-limited keys that are still available. If there are n equivalent RL_Rooms (actually each one requires a RL_Box) there will also be n keys for each time slot. Thus, the operational availability can be easily adapted to the needs by adding new RL_Box modules and corresponding RL_Rooms. By taking keys from the rack (in a FCFS-manner) the students can reserve a remote lab room for a specific time slot. Because there is only one key per RL_Room and time slot mutual exclusive access is guaranteed. The key holder can only act as an experimenter during the corresponding time slot defined by the key. He can also create and pass observer keys which allow the access of the lab room as an observer but not permit the control of the experimental setup. The results of an experiment can be stored in pages of a group's lab room where they can be discussed within the working group and prepared for submission to their supervisor.

4.3.4 Experimentation in collaborative remote labs

During experimentation (i.e. their temporal key grants access to the RL_Room), users may enter the RL_Room indicated in their temporal key and use the remote control facilities provided by the applets in the RL_Room, record data and communicate with each other over the room's communication channels. Using temporal keys, CRL ensures that only users assigned to this time period get access to an RL_Box. When the first user enters an RL_Room, CRL ensures that the state of the RL_Box is restored (based on the status data available in the lab room of the group) before experimentation can continue.

Different access permissions to rooms and their content can be incorporated into the keys. For example, we can assign different user roles: (1) supervisor (who sets up the experiment and defines possible control actions), (2) experimenter (who has the active control), and (3) observer (who can only see what happens in the laboratory). Such permissions must have been granted by a user with the required rights (e.g., the supervisor, or the student who got the temporal key during reservation) at the time when the keys were created.

4.3.5 Post processing in collaborative remote labs

After the experiment finished and the temporal key to the RL_Room expired, all experimental data, state/ configuration data as well as the communication logs recorded during the experiment are automatically transferred by CRL as persistent documents into the lab room (TL_Room) of the team. In the TL_Room, the research team can do the post processing of the experiment (e.g. data analysis and documentation).

If a need for a follow-up experiment arises, the supervisor can grant new keys for another period of exclusive access to either the same of another remote lab room of the same type.

5 Implementation

CRL has been implemented as an extension of the CURE collaborative learning environment of the FernUniversitaet. CURE is a web-based shared workspace system supporting distributed learning and working by shared rooms containing shared pages and communication channels [6]. CURE's notion of virtual keys [7] was extended with time-limited (temporal) keys and operations for generating tables of time-limited keys for a number of rooms. In addition, mechanisms for integrating applets into CURE pages and for restoring the state of RL_Boxes were added.

Extending CRL with remote controlled experiments required the implementation of a communication protocol between the RL_Box as described in section 2.2, allowing the set-up/restore of an RL_Box and connecting applets and the RL_Box (cf. Figure 2).

In order to support the development of new experiments we provide Java applets for controlling common instruments and the wiring of the experimental setup (e.g. oscilloscope, signal generator, analog multiplexer). These applets can be uploaded into XPort-modules and then these devices can be remotely controlled via the internet. Moreover, we also provide preconfigured boxes of different sizes that contain web cams as well as power switching and lightening facilities.

6 Experiences

Our implementation of remote labs has been tested successfully in a number of lab courses in the computer science programs at the FernUniversität. The respective distance learning courses are concerned with the architecture of various embedded computer systems like microprocessors and digital signal processors. In order to deepen the understanding of the working principles we offered our students remote access to development tools and to real experimental systems. The students had to conduct two experiments. Firstly, they implemented a program for real-time decoding of a radio clock signal by means of a microprocessor. Secondly, a linear low-pass filter for audio signals was realized by means of a digital signal processor.

We evaluated our approach by comparing 3 groups of remote students using our remote labs with regular student groups performing the experiments in labs at the FernUniversität. We used a questionnaire to evaluate usability and response time issues, measured completion times of experiments, and evaluated the learning outcome through oral exams.

The results of the questionnaires show that most of the students were confident with these two aspects of our remote lab. They appreciated that the functionality of the remote control applets was restricted to the essential functions needed. They also favoured the integration of all control applets in one page of the CRL room. The comparison of completion times shows that remote learners needed on average 5 minutes longer to conduct the experiments than presence students (requiring 150 minutes), probably because they had to make themselves familiar with the remote control applets. In the oral exams after the experiments remote learners performed as well as the presence students. Thus, the results indicate that our approach seem to be applicable.

In the InstantGrid project [20], CRL is currently ported on a Grid platform. Here, remote labs will be one eScience application available on a Grid.

Thus, InstantGrid will serve as a test bed for receiving feedback from the eScience community.

7 Conclusions

The framework presented in this paper enables (scientific) communities to realize the benefits of collaboration through sharing remote real laboratory equipment. It consists of a collaboration platform integrated with remote, real hardware. The current version supports the five tasks for creating and using remote labs, incl. reservation and access management as well as shared experimentation and analysis.

By means of CRL, remote laboratory rooms can be reserved and entered with temporally valid keys. In the reality augmented virtual rooms, users can access real laboratory equipment, conduct experiments, deposit measurement results and meet others for discussions about the experimental results. Thus, GridLabs seems useful for learning eScience and performing eScience.

Our approach exceeds current work by suggesting and supporting a life cycle of remote labs, including reservation and access control. Through the integration of control applets in RL_Rooms, a unified appearance of instruments' remote control panels is ensured. Users can access the remote equipment through their web browser and collaborate using the functionality of the shared workspace.

This approach is applicable to different types of experiments in different disciplines, such as microcomputer, DSP, and microcontroller experiments in computer science or typical lab experiments needed in physics, biology, medicine, electrical engineering etc. It provides a basis for national/international collaboration, such as mutual exchange of experiments, provision of a resource broker for searching for sharable experiments, and providing remote access to research infrastructure (e.g. remote controlled telescopes, sensor nets, particle accelerators).

For the future we plan to make our experiments publicly available over the Internet. Also we will develop an accounting scheme and enhance the communication between collaborating students by audio and video channels.

References

1. A. Bagnasco, G. Parodi, D. Ponta, A.M. Scapolla, "A Modular and Extensible Remote Electronic Laboratory", International Journal on Online Engineering (iJOE), Volume 1, No. 1, 2005
2. J. Baran, M. Connors, K. Quigley, R. Currie, "Remote Access to Instrumental Analysis for Distance Education in Science", The International Review of Research in Open and Distance Learning, Vol 6, No 3, 2005.
3. M. Cooper, "Remote laboratories in teaching and learning – issues impinging on widespread adoption in science and engineering education", International Journal on Online Engineering (iJOE), Volume 1, No. 1, 2005
4. S. Das, L. N. Sharma, A. K. Gogoi, "Remote Communication Engineering Experiments Through Internet", International Journal on Online Engineering (iJOE), Volume 2, No. 1, 2006.

5. I. Gustavsson, J. Zackrisson, A. Akesson, L. Hakansson, I. Claesson, and T. Lagö, “Remote Operation and Conrtol of Traditional Laboratory Equipment”, International Journal on Online Engineering (iJOE), Volume 2, No. 1, 2006.
6. J. M. Haake, T. Schümmer, A. Haake, M. Bourimi, B. Landgraf, “Supporting flexible collaborative distance learning in the CURE platform”, Proc. Hawaii International Conference on System Sciences, HICSS-37, January 5-8, IEEE Press, 2004.
7. J. M. Haake, A. Haake, T. Schümmer, M. Bourimi, B. Landgraf, “End-User Controlled Group Formation and Access Rights Management in a Shared Workspace System”, Proc. ACM CSCW’2004, November 6-10, Chicago, USA, pp. 554-563. ACM Press: NewYork, 2004.
8. V.J. Harvard, J.A. del Alamo, V.S. Choudhary, K. deLong, J.L. Hardison, S.R. Lerman, J. Northridge, D. Talaverna, C. Varadharajan, S. Wang, K. Yehia, D. Zych, “iLab: A Scalable Architecture for Sharing Online Experiments”, International Conference on Engineering Education, Gainesville, 2004.
9. M. Jurcevic, R. Malaric, and A. Sala, “Web based platform for distance training On Electrical Measurement Course”, Measurement Science Review, Volume 6, Section 1, No. 4, 2006.
10. D. Kennepohl, J. Baran, M. Connors, K. Quigley, R. Currie, “Remote Access to Instrumental Analysis for Distance Education in Science”, The International Review of Research in Open and Distance Learning, Vol 6, No 3, 2005.
11. R. Langmann: “Web-based Remote Control by LiveConnect”, International Journal on Online Engineering (iJOE), Volume 1, No. 1, 2005.
12. J. H. Lee, A. Prakash, T. Jaeger, G. Wu, “Supporting Multi-User, Multi-Applet Workspaces in CBE”, The Proceedings of the Sixth ACM Conference on Computer-Supported Cooperative Work (CSCW 96), Boston, MA, November, 1996.
13. D. Müller, J.M. Ferreira, “Online Labs and the MARVEL experience”, International Journal on Online Engineering (iJOE), Volume 1, No. 1, 2005.
14. J. Murphy, J. Grout, J. Walsh, and T.O. Shea, “Local and Remote Laboratory User Experimentation Access using Digital Programmable Logic”, International Journal on Online Engineering (iJOE), Volume 1, No. 1, 2005,
15. G. M. Olson, D. E. Atkins, R. Clauer, T. A. Finholt, F. Jahanian, T. L. Killeen, A. Prakash, T. Weymouth, “The Upper Atmospheric Research Collaboratory (UARC)”, Interactions, 5(3):48-55, 1998.
16. G. M. Olson, D. E. Atkins, R. Clauer, T. A. Bagnasco, G. Parodi, D. Ponta, A.M. Scapolla, “A Modular and Extensible Remote Electronic Laboratory”, International Journal on Online Engineering (iJOE), Volume 1, No. 1, 2005.
17. T. Schümmer, S. Lukosch, J. M. Haake, “Teaching Distributed Software Development with the Project Method”, In T. Koschmann, D. D. Suthers, and T.-W. Chan (eds.): COMPUTER SUPPORTED COLLABORATIVE LEARNING 2005, Proceeding of CSCL’2005, Taipei, Taiwan. Lawrence Earlbaum Publishers, 2005.
18. S.C. Sivakumar, W. Robertson, “Development of an Effective Remote Laboratory for Online Internetworking Education”, Proceedings of the 37th Hawaii International Conference on System Sciences, 2004.
19. http://www.lantronix.com/devicenetworking/embedded-device-servers/xport.html
20. http://www.instant-grid.de