A large-scale Web-based virtual oscilloscope laboratory experiment

by C. C. Ko, Ben M. Chen, S. H. Chen, V. Ramakrishnan, R. Chen, S. Y. Hu and Y. Zhuang,

VLAB, a pilot Web-based virtual laboratory on an oscilloscope experiment, has been developed and launched for over 1000 first year undergraduate engineering students in the Faculty of Engineering at the National University of Singapore (NUS). Rather than simulating the oscilloscope display on the client as is often done in other virtual laboratories, the system uses real-time video capture of the actual oscilloscope display. In addition, the use of the mouse to turn the control buttons and knobs of the instrument has been implemented so that a more realistic feel of the instrument is provided. Since its launch, VLAB has in general received positive feedback from numerous users and can be accessed from http://vlab.ee.nus.edu.sg/vlab.

Use of the Internet has been expanding exponentially; it is now extensively used as a connectivity and reference tool for commercial, personal and educational purposes. In particular, the Internet provides many new learning paradigms, including distance and group learning. More recently, the technology has even extended into the realm of conducting laboratory experiments at any time and from any location through the Internet.

A university distance teaching system that provides lessons, seminars and tutoring has been operating at the Politecnico di Milano[1,2] for a few years. A virtual classroom[3] which uses HTML (HyperText Markup Language) and HTTP (HyperText Transport Protocol) to introduce the capabilities of VHDL (Very large scale integrated Hardware Description Language) tools has been constructed at the University of Erlangen-Nuremberg. The interactive and computational capabilities of the Java language have been demonstrated in Reference 4 using a simple matrix assembly Java applet. These virtual laboratories provide instrument simulators, which can be a powerful auxiliary didactic tool to give students a basic idea of the instruments, their control and operation.

Reference 5 describes some laboratory experiments that can be run remotely via a Web interface. This type of laboratory is well suited for distance learning, where students may not be physically on campus. Parameters

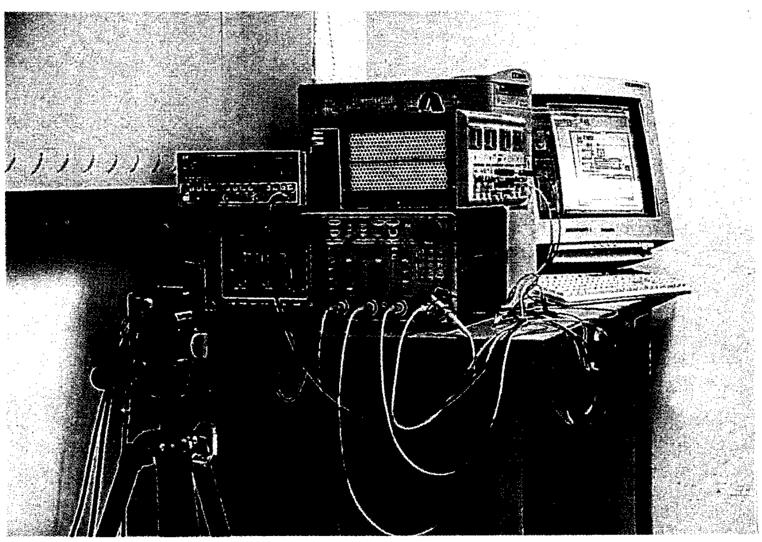

Fig. 1 Laboratory set-up of the oscilloscope experiment

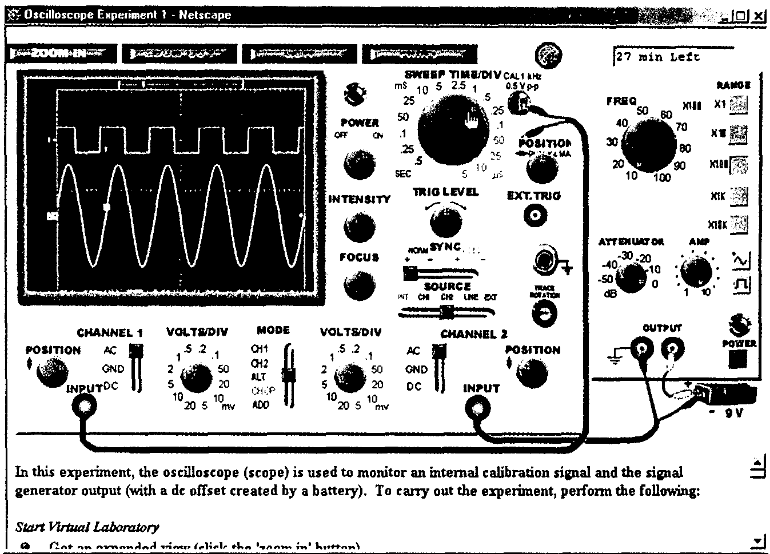

Fig. 2 One-user interface of VLAB on the client side

can be set on the Web, and a software interface converts these parameters to a form that is accepted by the local computer running the experiment. Atul and Molly[5] developed the software for 'Second best to being there', a distance-learning application that allows a remotely located user to conduct experiments in the Control Engineering Laboratory at Oregon State University. This permits the experiments to be run via the WWW (World Wide Web). Access requires only a basic Web browser that runs Java and is possible from most computer platforms. A robot experiment was developed for the demonstration. The Bytronic Process Control unit[6], referred to as the process rig in the Process Control and Automation Laboratory at Case Western Reserve University, can be accessed remotely via the Internet. Using a Web browser, the user can log on and post the parameters from a remote client to a LabVIEW (Laboratory Virtual Instrument Engineering Workbench, a program from National Instruments[7,8]) Web server that is connected to a process rig via a PLC (programmable logic control) module. The image is generated by a program and refreshed using server push technology.

In this paper, we present an approach to controlling real instruments remotely and describe a remote oscilloscope experimentation system, VLAB, developed using this approach. LabVIEW as well as WWW techniques such as CGI (Common Gateway Interface), JavaScript and HTML were used. Users only need a graphical browser which supports JavaScript to conduct and monitor the experiment remotely. User-friendly interfaces have been built and a video server is being used to broadcast live what happens in the real laboratory. VLAB[9,10] was launched in March 1999, and is part of a compulsory experimentation course for the more than 1000 first year undergraduate engineering students in the Faculty of Engineering at the National University of Singapore (NUS). It realises an interactive learning environment in which engineering students can control an oscilloscope and signal generator through the Internet to make real measurements from a remote site 24 hours a day.

The oscilloscope and signal generator in VLAB are connected to a set of PCs through the GPIB (General Purpose Interface Bus). A video camera in the laboratory is used to allow the remote users to view the oscilloscope display. VLAB resides in a World Wide Web site maintained by our research group and the URL (Uniform Resource Location) is http://vlab.ee.nus.edu.sg/vlab. Fig. 1 shows a photograph of the physical experimental set-up in the laboratory.

We recognise that no simulation system will provide the same feel as conducting experiments in an actual laboratory. Compared with the virtual laboratories described in References 1-4, VLAB allows the student to work on real instruments and to conduct the experiment without simulating instruments. We have focused on developing a general methodology for making GPIB-based instruments accessible remotely. To achieve the aim of creating an environment in which students can gain the same experience as in a traditional laboratory, 'knobs' and 'controls' have been made as realistic as possible. Specifically, rather than simulating the oscilloscope display, VLAB carries out real-time video capture of the actual oscilloscope display and sends it to remote users.

The user interface

A remote laboratory for an oscilloscope experiment should have a realistic and user-friendly interface through which users can control the real instruments. One important consideration is that if the user sees a non-realistic interface, an impression will be created that the experiment is not easy to conduct and this will have a negative effect on using the virtual laboratory as a teaching tool.

The graphic interface of VLAB on the client side is implemented using JavaScript[11] and HTML[12]. Fig. 2 shows the front panels of the oscilloscope and signal generator, the two main instruments in the laboratory. In the case illustated, the two instruments are used to make measurements on an RC circuit. The experiment is set up by 'dragging' various leads and cables to connect relevant points of the circuit to the two instruments. This is done so as to create a realistic impression of the connections that have to be made in an actual experiment.

The main functions and features of the user interface are summarised below:

- The front panels of the oscilloscope and signal generator, the circuit and the connectors are implemented in a graphical manner.
- As shown at the top left of Fig. 2, a real-time video window on the oscilloscope display has been implemented. Rather than simulating the oscilloscope display on the client as is often done in other virtual laboratories, real-time video capture of the actual oscilloscope screen is carried out.
- Control buttons and knobs can be turned by dragging the mouse. This provides a more realistic feel to operating the instrument. For example, frequency can be changed by simply positioning the mouse inside the 'FREQ' knob and dragging the knob in a circular manner.
- The user interface is designed to work logically. For example, in carrying out the experiment, the students have to connect the circuit to the oscilloscope and to the signal generator. (As shown in Fig. 2, this connection is performed by dragging the coaxial connector to the Channel INPUT and by dragging the clip near the signal generator to the signal generator terminal.) If the circuit is not connected correctly, the oscilloscope will not show the desired display.
- An online operating manual on experimental procedure has been developed, as shown at the bottom of Fig. 2. This enables students to grasp quickly key concepts and knowledge in a 'learn by seeing and doing' manner.
- A clock at the top right of the user interface keeps the user informed of the time remaining before VLAB logs the user out of the system. Each user is allocated 30 minutes after gaining access.
- The important Web pages that control the real instruments are protected by VLAB and only authorised users can access these pages.

A typical session

This section describes briefly a typical session for conducting the remote oscilloscope experiment using the interface described in the last section. The VLAB experiment comprises four parts: 'Basic functions and controls', 'Measurement of phase', 'Frequency response of LP (low pass) filter', and 'Response of series RC circuit to square-wave input'.

A typical VLAB session consists of the following steps:

- Activate a Web browser, access the VLAB home page and click the 'Laboratory' button.
- An introduction page will appear. Before conducting the actual experiment, the user can view various Web pages related to the experiment by clicking the links provided on the left. Fig. 3 shows a web page on the functionality of the oscilloscope. The aim of these Web pages is to give some preliminary information to the user on VLAB, the experiment and the various instruments involved.
- After gaining some familiarity with the experiment, the user can click 'Conduct Expt' to conduct the actual experiment under VLAB. A logon page

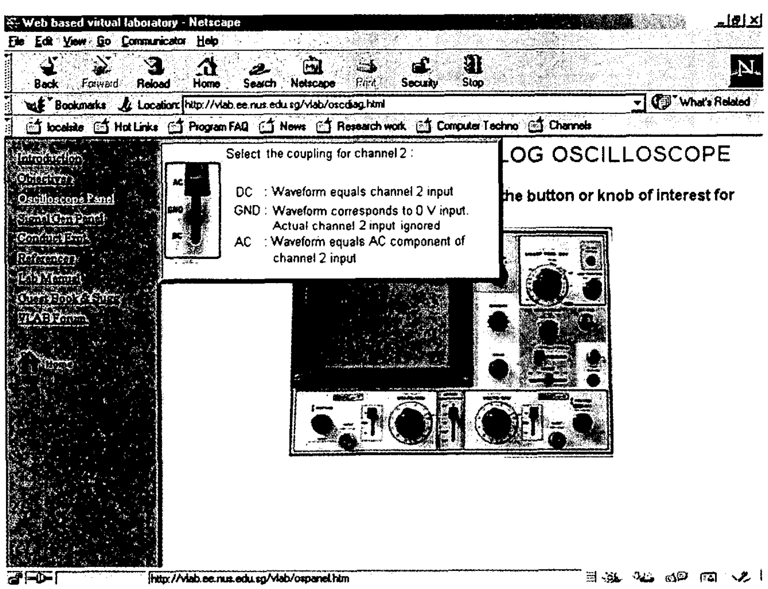

Fig. 3 A Web page on the functionality of the oscilloscope

requesting the user to provide the username and password for authentication will follow. If the user does not have an account, they can still log on using the guest account with the username 'guest' and the password 'welcome'. The guest account has the same privileges as an authorised user account except for the ability to log all interactions in the laboratory session (this feature is useful to the student if they wish to repeat the experiment at a later time). After the user has been authenticated, an internal session ID is granted by VLAB.

- After authentication, the user can select one of the 4 parts of the experiment. The relevant interface page for controlling the real instruments will then appear.
- The user can now conduct the actual experiment remotely following the online manual on procedures and obtain results through real-time video feedback.
- The session can be terminated by the user at any time or after a default time limit of 30 minutes for conducting the experiment has been reached. A clock, at the top right of the user interface, keeps the user informed of the time remaining. After the user has logged off or the session has been terminated, all connections are released.

Hardware and software structures

Hardware structure

Fig. 4 gives a block diagram of the main hardware structure and components in VLAB. The various hardware subsystems are:

- A PC serving as the instrument controller and equipped with a GPIB interface card and an Ethernet card. This is connected to the Internet through the NUSNET-III[13] network in NUS. (NUSNET-III is a campus-wide network which interconnects 104 departments in 90 buildings and covers a campus area of 150 hectares. It serves a population of 24 000 students, and 2700 academic, research and administrative staff members.) The instrument controller receives the command string for instrument control from the WWW server through a TCP/IP channel. Thus, in addition to being used as an instrument controller, it represents the server side of the communication with the WWW server (which represents the client side in this scenario).
- The programmable instruments, namely the oscilloscope and the signal generator, are connected to the controller PC through a GPIB card and a GPIB cable. The GPIB card can also be connected to other instruments which support the SCPI (Standard Commands for Programmable Instruments) command structures defined in IEEE488.2[14]. A maximum of 15 device loads can be connected to each bus, with no more than two-thirds powered on.
- A real-time video server enables visual feedback on what is happening in the real laboratory. A monochrome camera connected to this video server provides the visual feedback.
- A WWW server hosts the Web site of VLAB.

Software structure

The software structure of VLAB is shown in Fig. 5. On the server side of our realisation, VLAB controls the local instruments, hosts the Web site, and provides video feedback. Local instrument contol is implemented using LabVIEW, while the WWW server is

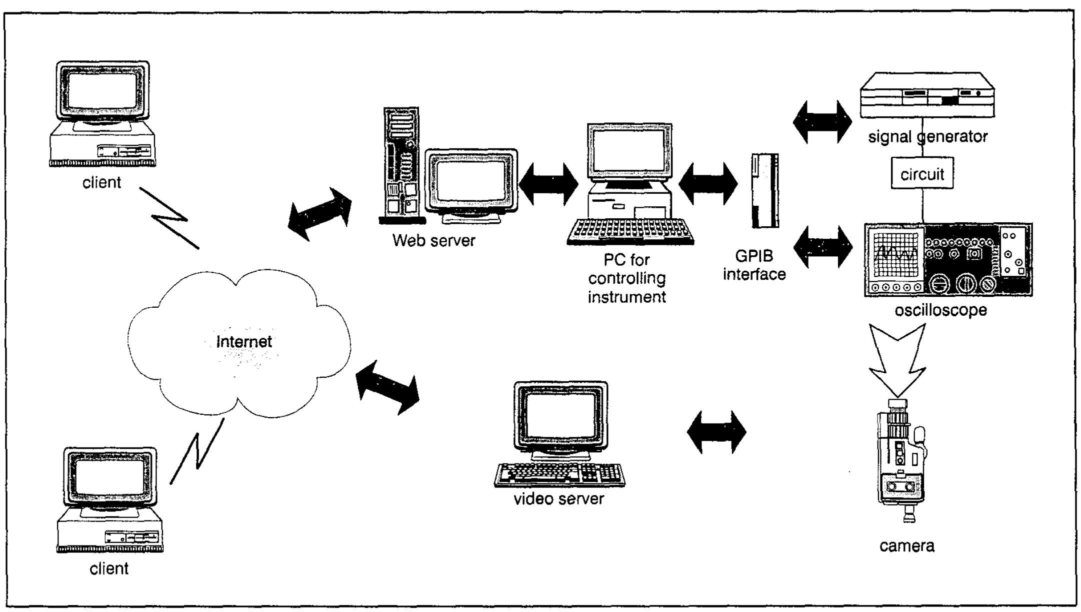

Fig. 4 Hardware structure of VLAB

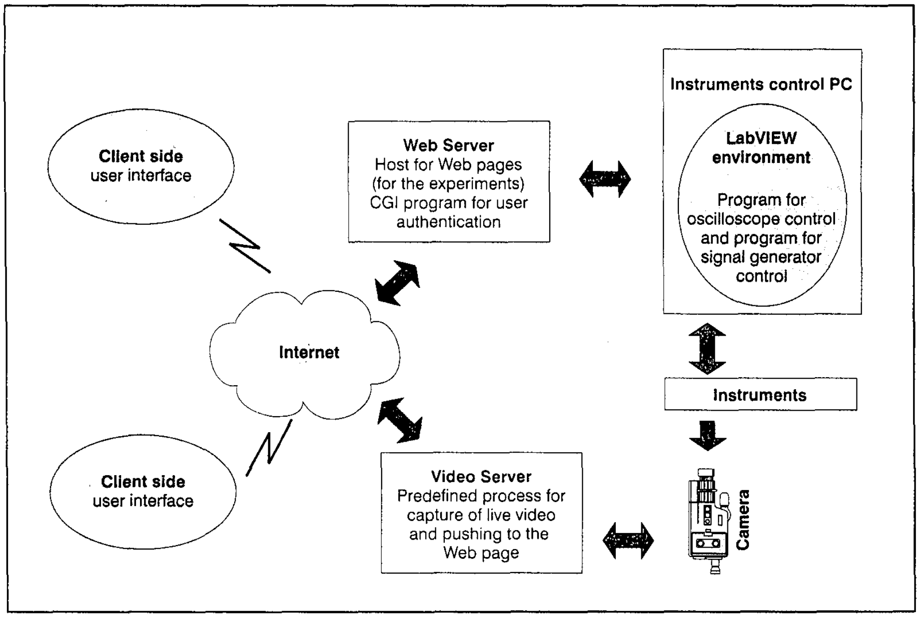

Fig. 5 Software structure of VLAB

implemented using a PC running Red Hat Linux 5.2 with the mSQL database from Hughes Technologies and the Apache HTTP Server. A video server is used to provide users with video feedback. For platform independence and ease of setting up the system, InetCAM[15], which does not require any plug-ins, has been chosen as the video server. The live video captured by the camera is pushed by the InetCAM program to the client's IP (Internet Protocol) address obtained when a client places a request with the video server. The client program will run optimally in the environment provided by Netscape Communicator 4.0 or later versions.

Implementation of local instrument control using LabVIEW

LabVIEW[7,8] has been chosen to implement local instrument control. This is a powerful instrumentation and analysis programming environment for PCs running Microsoft Windows 95 and various other operating systems. A graphical programming language called G enables programming in a block diagram manner and subsequent compilation into machine code.

LabVIEW integrates data acquisition, analysis, and presentation in one system. For acquiring data and controlling instruments, LabVIEW supports RS-232, GPIB, including VISA (Virtual Instrument Software Architecture) functions, as well as plug-in DAQ (Data Acquisition) boards.

Commands for modifying settings or initiating specific actions are sent to programmable instruments through the GPIB interface. Based on the GPIB port number assigned to the instrument, the commands are transferred through the GPIB card and cable to the corresponding instrument, which interprets the commands and takes appropriate actions. Some typical commands for controlling the oscilloscope[16] are described in Table 1 and those for controlling the signal generator[17] are described in Table 2.

Implementation of remote instrument control through the Internet

CGI and TCP (Transmission Control Protocol) have been selected as the methods of communication between the client PC and the WWW server, and also between the WWW server and the controller PC.

Table 1: Commands for oscilloscope

Action	Mnemonic	Value delimiter
Turns the specified channel display on and selects channel	select:ch<X> on	X = the number of the channel
Turns the specified channel display off	select:ch<X> off	X = the number of the channel
Sets the vertical gain of the specified channel	Ch<X>:SCALE<NR3>	X = the number of the channel <NR3> is the gain, in volts per division
Sets the vertical position of the specified channel	Ch<X>:Postion<NR3>	X = the number of the channel <NR3> = the desired position
Sets the time per division for the main time base	HORizontal:SCALE <NR3>	<NR3> = the time per division
Positions the waveform horizontally on the display	HORizontal:Postion <NR3>	<NR3> = the desired position
Establish AC | DC | GND coupling on the specified channel	Ch<X>:coupling AC Ch<X>:coupling DC Ch<X>:coupling GND	X = the number of the channel AC = AC coupling DC = DC coupling GND = ground

Table 2: Commands for signal generator

Action	Mnemonic	Value delimiter
Set frequency	FRQ	MZ = millihertz HZ = hertz KHZ = kilohertz MHZ = megahertz
Set amplitude	AMP	MV = millivolts V = volts
Select sine waveform	W1	
Select square waveform	W3	
Enable output	D0	
Disable output	D1	

CGI is a standard for setting up interaction between external applications and information servers, such as WWW servers. CGI programming involves designing and writing programs that are usually invoked from HTML forms on Web pages. The HTML form has become a popular method for sending data across the network because of the ease of setting up the user interface using HTML 'form' and 'input' tags.

LabVIEW supports network communication protocols, such as TCP and UDP (User Datagram Protocol), implemented in the form of Virtual Instruments' sub VIs. LabVIEW uses the following VIs to communicate through the Internet:

- The TCP *Listen.vi*: This VI waits for an incoming TCP connection request. It also returns a connection ID when the TCP connection is created.
- The TCP *Read.vi*: This VI receives a specified maximum number of bytes to read from the specified TCP connection.
- The TCP *Write.vi*: This VI writes the string data to the specified TCP connection.
- The TCP *Close Connection.vi*: This VI is used to release the TCP connection.

In our implementation, the WWW server accepts the parameters from the client and then passes these parameters to a CGI program (*server.pl* coded in Perl). The CGI program determines whether the entered parameters are acceptable or not. If the parameters are not acceptable, the CGI program sends error messages back to the client. If the parameters are acceptable, the CGI program establishes a TCP connection with the instrument controller to communicate with a LabVIEW program (*listener.vi*) and passes it the parameters. This LabVIEW program invokes a local instrument control program (coded in LabVIEW G), which commands the attached instruments via the GPIB interface. After this, the TCP connection between the instrument controller and the WWW server is released. This is a 'double client–server' implementation.

Such an architecture was chosen for security reasons. If the commands were sent to the instruments directly, more than one user could control the real instruments simultaneously and this would lead to a chaotic situation. Additionally, before the request reaches the controller PC, the system has to be assured that the request is authorised. Therefore, a method to enable indirect communication between the client browser and the instrument controller had to be developed. As can be seen from Figs. 4 and 5, VLAB involves CGI programs running on the WWW server, instruments attached to the instrument controller, and a CGI interface for communication between the client browser and the WWW server. For indirect communication between the browser and the instrument controller, VLAB includes a program developed to provide connectivity between the CGI program on the WWW server and the LabVIEW program on the instrument controller.

Fig. 6 depicts the double client–server structure, which enables interaction between the client browser and the instrument controller. As shown, server.pl located on the WWW server acts as a server for the client browser and represents the first client–server structure, while listerner.vi located on the instrument controller acts as a server for server.pl. This double client–server structure prevents access conflict, guarantees the security of VLAB and ensures that only one user can control the instruments remotely at one time. This method of remote instrument control can be generalised to implement the Web-based control of any GPIB-based instrument.

Authentication in VLAB

In order to guarantee that only one user has active

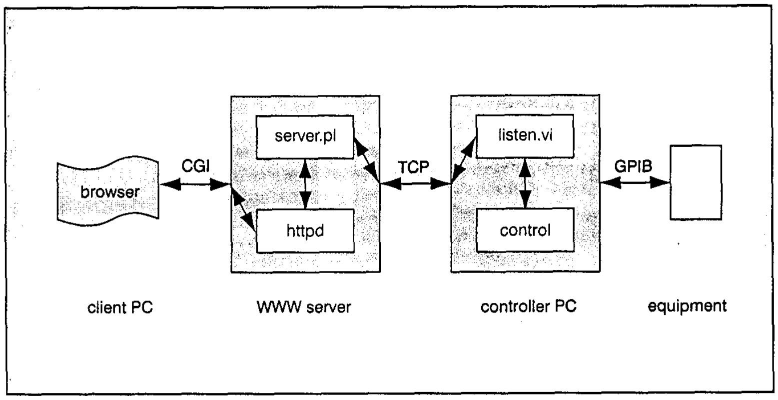

Fig. 6 Double client-server structure

control of the experiment at any time and that only authorised users can access VLAB, user authentication has been implemented. The program for user authentication is coded in Perl and consists of four basic blocks, as depicted in Fig. 7. The login block is concerned with accepting user input (from the client) and breaking up the input data into field name-value pairs. The session manager performs the function of verifying the user name and password (which are obtained from the login block). This block also indicates to the login block an access grant or denial by sending it appropriate data. Another function performed by this block is the setting of the internal session ID (by writing a record into a data file indicating that a user has logged on). The session manager also logs all the client's interactions during the experimental session. Logging helps in maintaining a record of the students' day-to-day laboratory work. This feature is useful for reconducting the experiment. The controller block prevents user access conflict by checking with the session manager whether a user has logged on. The controller block is requested by the program (referred to as the 'Laboratory Manager' in Fig. 7) implementing communication between the instrument controller and the WWW server to return the amount of time the user has spent in accessing the laboratory. If this exceeds 30 minutes, the laboratory manager sends a 'terminate access' request to the session manager. The latter logs the user out and the session is terminated.

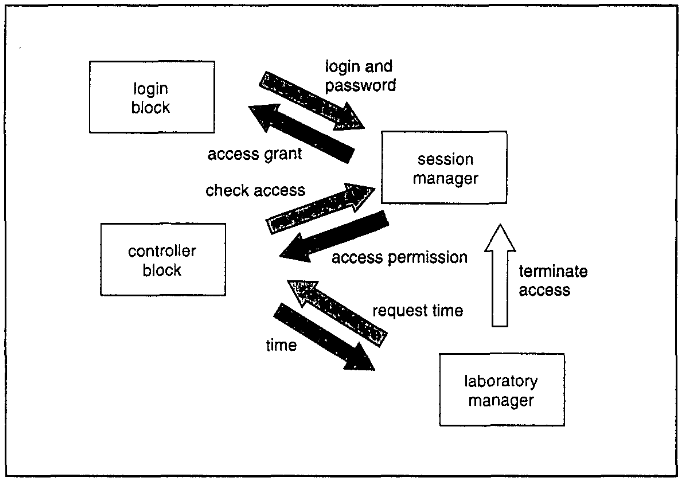

Fig. 7 Server side authentication

Future applications

Currently, VLAB can be accessed from http://vlab.ee.nus.edu.sg/vlab. Based on the methodology developed, other virtual laboratories and similar applications are currently being developed. These include a 'DC Motor Control' experiment on the computer-based control of a DC motor, a voltage-controllable electromechanical system; and a 'Flow and Pressure Measurement' industrial control experiment. In the latter experiment, a 'Flow Measurement Test Bed' can be connected to a PC via the AD/DA card. The output of the test bed can be sampled by an oscilloscope and transmitted to the student and a camera can enable the students to view what is happening on-site as well as the oscilloscope display.

Further research and development work is being carried out on the video server system in VLAB. This, as mentioned earlier, enables the user to view real-time video on the experiment through the Internet. Because of the high network bandwidth required, remote experiments developed using such a system may require access through a high-speed campus or similar network. Specifically, the possibility of mounting VLAB through SingAREN (Singapore Advanced Research and Education Network)[18], a high-speed network with a 14 Mbit/s ATM link to vBNS (very High-speed Backbone Network Service) in the USA and to a CA*NET II in Canada, is being explored.

On another front, the possibility of using different robots to control instruments which are non-programmable is also being investigated.

Conclusion

A general methodology to create a Web-based remote laboratory has been presented. Based on this methodology and, specifically, using LabVIEW for implementing local instrument control and a double client–server structure for remote instrument control, VLAB, a Web-based virtual laboratory on an oscilloscope experiment for 1000 undergraduate engineering students, has been developed and was mounted in March 1999. Rather than simulate the oscilloscope display on the client as is often done in other virtual laboratories, the system uses real-time video capture of the actual oscilloscope display. In addition, the use of the mouse to turn the control buttons and knobs of the instrument has been implemented so that a more realistic feel of the instrument is provided. Since its launch, VLAB has in general received positive feedback from numerous users and can be accessed from http://vlab.ee.nus.edu.sg/vlab.

References

1 BROFFERIO, S. C.: 'A university distance lesson system: experiments, services, and future developments', *IEEE Trans. Educ.*, February 1998, **41**, (1), pp.17–24
2 FERRERO, A., and PIURI, V.: 'A simulation tool for virtual laboratory experiments in a WWW environment'. Proc. IEEE Conf. on Instrumentation and Measurement Technology, St. Paul, Minnesota, USA, 18th–21st May 1998, **1**, pp.102–107

3 HEINKEL, U., PADEFFKE, M., KRAUS, O., and FRICKEL, J.: 'Virtual-classroom: an interactive teaching system for the design of integrated circuits', *Global J. Eng. Educ.*, 1997, **1**, (2), pp.173–181

4 CABELL V, B., RENCIS, J. J., and GRANDIN, H. T.: 'Using Java to develop interactive learning material for the World-Wide Web', *Int. J. Eng. Educ.*, 1997, **13**, (6), pp.397–406

5 SHOR, M., and BHANDARI, A.: 'Access to an instructional control laboratory experiment through the World Wide Web'. Proc. 1998 American Control Conf., Philadelphia, pp.1319–1325

6 SHAHEEN, M., LOPARO, A., and BUCHNER, M. R.: 'Remote laboratory experimentation'. Proc. 1998 American Control Conf., Philadelphia, pp.1326–1329

7 'LabVIEW User Manual'. National Instruments, 1998

8 'LabVIEW online help'. National Instruments, 1998

9 Web-based Virtual Laboratory in the National University of Singapore. Available at http://vlab.ee.nus.edu.sg/vlab

10 CHEN, S. H., CHEN, R., RAMAKRISHNAN, V., HU, S. Y., ZHUANG, Y., KO, C. C.,and CHEN, Ben M.: 'Development of remote laboratory experimentation through Internet'. Proc. 1999 IEEE Hong Kong Symposium on Robotics and Control, Hong Kong, July 1999, **II**, pp.756–760

11 KENT, P.: 'Official Netscape JavaScript 1.2: programmer's reference, Windows, Macintosh, & Unix' (Ventana Communications Group, Research Triangle Park, NC, USA, 1997)

12 W3C, HTML 4.0 Specification. Available at http://www.w3.org/TR/REC-html40/

13 Home page of NUSNET-III. Available at http://www.cc.nus.edu.sg/

14 CARISTI, A. J.: 'IEEE-488, general purpose instrumentation bus manual' (Academic Press, San Diego, USA, 1989)

15 InetCAM. See http://www.inetcam.com/

16 'TDS family digitizing oscilloscope program manual'. Tektronix Inc., 1995

17 'HP 8116A 50 MHz programmable pulse/function generator operating, programming and servicing manual'. Hewlett Packard, 1993

18 Singapore Advanced Research and Education Network (SingAREN), home page of SingAREN. See http://www.singaren.net.sg/

The authors are with the Department of Electrical Engineering, National University of Singapore, 10 Kent Ridge Crescent, Singapore 117576
E-mail: elekocc@nus.edu.s

BOOK REVIEW

Properties, processing and applications of glass and rare earth-doped glasses for optical fibres

Dan Hewak (Ed.)
INSPEC, The Institution of Electrical Engineers
1998, 376pp., £125
ISBN 0 85296 952 X

Optical fibres manufactured from high-purity silica-based glass form the basis of the global telecommunications infrastructure. Although silica is the dominant glass material, it is less well known that there are many other material systems under active development for applications in the telecommunications and (possibly) sensor networks of the future.

This publication provides a timely review of these various glass systems (silica, other (non-silica) oxides, fluorides, chalcogenides) currently under investigation in government and commercial laboratories around the world. Each of these is addressed through a uniform series of topics, making it easy to compare and contrast the properties and performance of the different glasses. The topics covered are the optical, thermal and mechanical properties, rare-earth spectroscopy, optical-fibre manufacture and selected applications.

The authors of each section have been chosen mainly from Europe and Japan and as such the book provides a nice counterpoint to the sometimes US-centric view of both historical and current developments in optical fibres. That having been said, the coverage of the topics is excellent in both its breadth and depth and will provide useful information to engineers and scientists at all levels in the optical-fibre industry. The references provided with each section provide an excellent review of the main publications in each area, with a good balance between the 'classic' references (the earliest I found was from 1887) and the current 'state of the art'.

If I have a quibble with this admirable and carefully assembled series of papers, it is that the 'applications' sections are somewhat 'thin' and, with a couple of limited exceptions, focused solely on the use of rare-earth-doped fibre in fibre lasers and amplifiers. For the next issue, an expansion of this area would make this volume an even more useful publication.

Finally, mention must be made of the introduction, by Professor Alec Gambling. It is a delight, linking such diverse topics as the funeral draperies of Napoleon and the erbium-doped fibre amplifier in a concise, easily-readable form. It provides an exemplary 'potted history' of the past 200 years of optical-fibre development which should be read by all neophytes—and some of us more experienced practitioners too!

Overall, this volume is an excellent collection of articles which provides a highly readable review of the current status of glasses for optical fibres. I hope that the editor can be prevailed upon to update it regularly to maintain its currency in this rapidly developing field.

SIMON POOLE
Technical Director
JDS Uniphase Pty Ltd.