

Developing a Prototype Virtual Laboratory for Distance Science and Engineering Education

Carl Steidley and Rafic Bachnak
Department of Computing and Mathematical Sciences
Texas A&M University-Corpus Christi

Abstract - Institutions of higher education are actively seeking new methods to complement their science and engineering distance education programs with online experimentation. This paper describes the design and development of a virtual laboratory environment that allows students to perform laboratory experiments from remote locations through a web browser. A front panel in LabVIEW displays the results and allows storage of acquired data for later processing. A digital camera that provides real-time pictures of the monitored equipment is part of the system configuration.

Index Terms – Distance learning, on-line labs, virtual laboratory

INTRODUCTION

Laboratory experiments in science and engineering courses play a significant role in supporting the learning process. With the increased demand for distance education, however, institutions are looking for methods that offer their students the opportunity to perform experiments through virtual laboratories [1- 4]. As a result, a number of web-based systems that rely on a variety of software development programs have been developed. Most software programs, however, make developing, maintaining, and upgrading laboratory experiments very expensive and a time consuming process.

A distance laboratory system has to provide efficient user operations, machine and platform independence, secure operations, graphical user interface capabilities, high processing bandwidth, and low cost maintenance. These factors and others have been affecting the implementation of distance education [5-7]. The ability to monitor and control processes from remote locations makes proper and timely analysis possible. Advances in networking technologies and development of measurement hardware and software has turned PCs into platforms capable of continuous remote monitoring using the Internet. A software program that facilitates the developments of such applications is LabVIEW [8-10]. LabVIEW, a graphical programming language for measurement and automation, allows embedding a user on a web page with access to the system. Access can be granted using security features available through the software. Features such as password protection can prevent private information from being viewed by the general public. This paper describes the design and implementation of a virtual

laboratory that enables students at Texas A&M University-Corpus Christi University to perform laboratory experiments online using a web browser. The system uses LabVIEW to provide data acquisition, storage, and processing capabilities and real-time viewing of the laboratory environment using a digital camera.

LABORATORY EQUIPMENT

The Control Systems Virtual Laboratory consists of the following major equipment: desktop computer stations, data acquisition systems, software (MATLAB and LabVIEW), level and flow measurement system, process control system, wireless data loggers, calibrators, and a variety of test equipment [11]. The data acquisition systems consists of several components, including a data acquisition board, signal conditioning units, mounting terminal block, and chassis. These components are integrated into real-time measurement units that interface to desktop computers and process control equipment. The National Instruments (NI) Software Solutions package, consisting of LabVIEW, control and simulation toolkit, Internet, Gmath, and signal processing suites, is used for programming these units.

The data acquisition system consists of a data acquisition board (PCI-MIO-16E-4), terminal blocks (SCXI 1303 and SCXI 1325), signal conditioning hardware (SCXI 1102), and a digital to analog conversion module (SCXI 1124). Assembling the system requires the use of the SCXI 1349 shielded cable assembly and the SCXI 1357 high voltage backplane. Fig. 1 shows a picture of the actual system.

SYSTEM CONFIGURATION AND WEB PUBLISHING

The Measurement & Automation Explorer (MAX) package is a program that comes with the professional version of LabVIEW. It is used to configure the data acquisition system. The instruction tree in MAX has two important folders. One of the folders is called “Data Neighborhood.” This folder is where channels are assigned and are given values to use throughout the data acquisition process. The software gives the user the option to click on “Create New-Virtual Channel” and then choose the type of channel needed. Analog Input, Analog Output and Digital I/O are the choices provided. The other folder is called “Devices and Interfaces”. Here, the modules that have been connected should be listed and can be configured by choosing properties. Once all of the modules and channels have been configured they can be tested automatically via the system GUI’s “Test Panel” button.

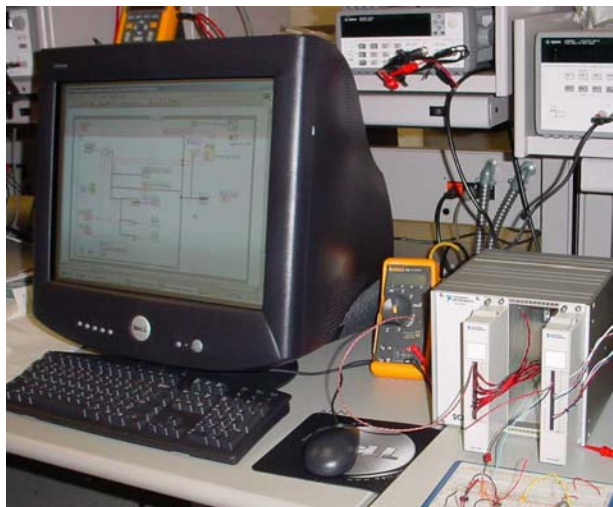


FIG. 1 DATA ACQUISITION SYSTEM SET-UP

As a prototype we have utilized the Temperature System Demo Virtual Instrument (VI) that comes with LabVIEW. We have modified the demo to accomplish real time measurements by reading data coming through the channels that were configured using MAX. The wiring diagram is reprogrammed by removing the randomly assigned numbers given in the subVI called, “Temp” and inserting the subVI called analog input and adding a DAQ channel control. The diagram is also modified so that an output voltage of five volts is sent through the output channel to the system demonstration LED.

LabVIEW has many Internet capabilities. It allows a user to publish a static report on the Internet, share data for processing, data storage, or data monitoring. It also allows the user to control a process from a remote location and provides distributed execution of a program. The Web-Publishing Tool facilitates publishing a VI on the Internet while granting and denying access to certain computers.

Once the VI is published, the server can grant anyone permission to take control of the VI but it also has the ability to regain control at any time. Providing read only, read/write, or denying access is easily managed with the Web-Publishing Tool.

INTEGRATING A VIDEO CAMERA

To provide students with the capability of observing virtually what they are physically controlling in the laboratory, we have installed a Sony DFW-VI500 camera. The MAX tool provides facilities for configuring cameras. MAX automatically detects cameras and any image acquisition board to which these cameras may be attached. MAX also provides the functions to change the configuration or test the camera. After MAX has launched, the properties of the camera can be set in the IMAQ IEEE-1394 Devices folder in the MAX configuration tree. Three NI-IMAQ image acquisition types are available in LabVIEW: snap, grab, and sequence. A snap acquires a single image into a memory buffer. A grab is a continuous, high-

speed acquisition of data. This function performs an acquisition that continuously loops, so it is suitable for the Web-Cam applications. Two VIs, IMAQ1394 Grab Setup and IMAQ1394 Grab Acquire, should be used for a grab acquisition. IMAQ1394 Grab Setup initializes the acquisition and starts capturing the image to an internal software buffer. IMAQ1394 Grab Acquire copies the image currently stored in the internal buffer to a LabVIEW image buffer. A sequence initiates a variable-length and variable-delay transfer to multiple buffers. A sequence can be used in applications that process multiple images.

There are two main ways to display images. First, a user can use IMAQ WindDraw provided by IMAQ Vision. IMAQ WindDraw provides a good functionality so it can display any type of images (JPG, PNG, BMP, 8-bit or 16-bit monochrome image, RGB image, etc.) and any acquisition types (snap, grab and sequence). However, it doesn't support Web Publishing Tool. Another way to display the images is to use picture control in LabVIEW. It supports Web Publishing Tool but a user has to know the exact image type. For instance, IEEE 1394 Cameras usually support three kinds of the image types: 8-bit and 16-bit monochrome, and RGB images.

Figure 2 illustrates the graphically programmed block diagram of our Web-Cam application. This application uses the grab acquisition type to continuously acquire RGB images captured by our Sony DFW-VI500 camera.

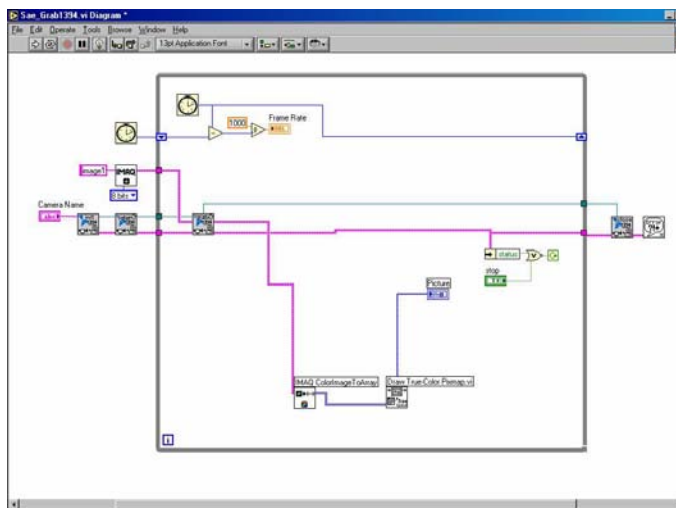


FIG. 2 BLOCK DIAGRAM OF A WEB-CAM APPLICATION

SOFTWARE DESIGN AND DEVELOPMENT

The Virtual Laboratory prototype currently has one experiment, the Temperature Measuring and Control System, mentioned above. This web application is a cross-platform application, located between clients and the LabVIEW Server. It is designed to serve as a communication tool between students and instructors, to control the secure access of students, and add/delete the links for LabVIEW experiments and the prototype is being tested by instructors and students. The Web application is coded with HTML, JavaScript and the PHP programming language. It can be accessed using any

July 7 – 9, 2005, Juan Dolio, Dominican Republic

major browser that supports HTML 4.0 and JavaScript. HTML 4 is used to generate the static Web pages and JavaScript is chosen to create dynamic Web pages on the client side because HTML 4 and JavaScript are compatible with many operating systems and supported by both Netscape and Internet Explorer. PHP is used to generate server-side scripts to communicate between the Web server and the data file. Like any other CGI program, PHP can generate content on the fly, or create a Web interface for adding, deleting, and modifying elements within the data file[12]. It can work with just about any combination of Web server and operating system. PHP user authentication can restrict access to the Web site.

SYSTEM ADMINISTRATION AND MANAGEMENT

All users, including instructors, gain access to the system through a Login Page. This page allows users to input User ID and Password and sends this information to an Authentication Page. Typically, the Authentication Page reads *userfile* to obtain a list of user IDs and passwords and compare the user input from the Login Page with a list of user IDs and passwords. When there is no matching pair, the Authentication Page sends an error message to the user browser. If the user ID and password from the Login Page is in the list of user IDs and passwords in *userfile*, the Session starts. Depending on the status of the user logged in, Authentication Page shows the Student Page or the Instructor Page. The Logout Page deletes the session ID from *sessionfile* and terminates the Session.

The functionality of the instructor page is illustrated in the diagram of Fig. 3. The Lab Info Page is for instructors to see the information on the lab. The Lab Info Page reads the information from *labfile* and displays the lab ID, title, description, and button to move the Modify/Delete Lab Page. The Modify/Delete Lab Page receives the request from the Lab Info Page to display the information for a specific lab ID. The data in the *labfile* is changed based on a modify or delete request. When an instructor sends the add new lab request with title, description, and URL of LabVIEW Server, the title, description and URL will be stored in *labfile*. The User Info Page reads the data from *userfile* and display the user ID and the user status and an instructor can choose the modify or delete button. The User Info Page will check whether there is the same user ID in *userfile* when it receives the modify-user or delete-user request. If there is the same user ID, the information of users will be changed from *userfile* based on the request type. If not, it will send an error message to the browser of the instructor.

When an instructor sends the add-user request with user ID, password and status, the Add New User Page checks whether the user ID is at the list of user IDs in *userfile*. If there is a matching user ID in the list, an error message will be sent to the instructor. If not, the user ID, password and status will be stored in *userfile*.

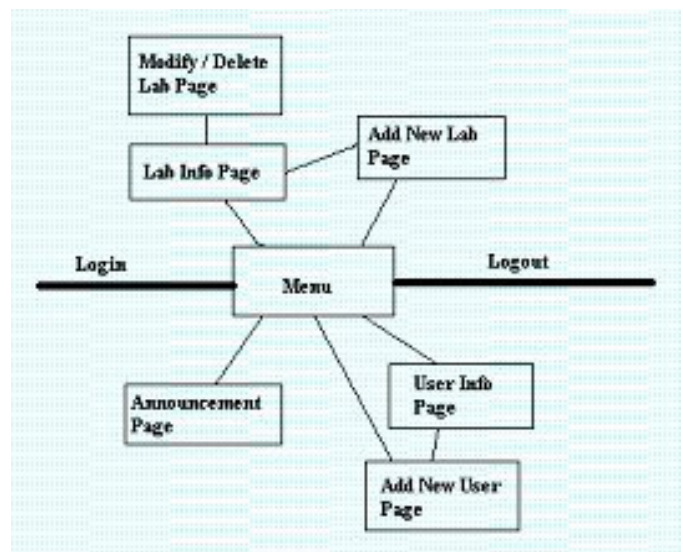


FIG. 3 TRANSITION DIAGRAM OF THE INSTRUCTOR PAGES

The Announcement Page reads the data from *announcementfile* and displays the announcement. When an instructor has changed the announcement and click the submit button, the changed announcement is changed in *announcementfile*.

The Student Page reads the lab information from the *labfile* and the announcement from the *announcementfile*. Using the data from *labfile* and *announcementfile*, the Student Page displays the announcement, lab ID, title, description, and button to request the lab experiment.

TESTING THE VIRTUAL LAB

The performance of the Virtual Lab is being examined by testing each function to verify its proper operation. For instance, when the session is over, the Web application allows users to access other Web pages. Users can reenter the system only from the Login Page. All users, including manager and instructors, access the system from the Login Page. Users must have proper user IDs and passwords to access the experiments. After a successful login, a user is able to see the list of available laboratory experiments and a description of each. Announcements may also be displayed at the top of this page. When students click the button of an experiment, they can see the graphical user interface of the LabVIEW application in a new pop-up window. In the case if our prototype, students can see the information for the Temperature System.

This system supports multi-user access. This means that all students who access this application see the same Front Panel. When a student wishes to control the lab experiment through the Front Panel, he or she must request control permission. If there is someone who already has control permission, other students can not have permission until control permission is released.

To request control permission, a student clicks the right button of the mouse on the Front Panel and chooses the

Request Control of VI. After a student is granted control, he or she can start the application by clicking the arrow at the top and can have access to all control functions. While the student manipulates some functions available on the temperature system, other students can only watch the changes on the front panel. Students release control of the experiment when they have finished their experiment by selecting the Release Control of VI.

The instructor can control the view of Student Page, give the student access permission by assigning user IDs and passwords, and make announcements. The instructor can also add the link of new experiments and delete links for unavailable ones. Instructor pages include Lab Info Page, Add New Lab Page, and Modify/Delete Lab Page.

The Lab Info Page allows the instructor to see the list of labs and the current announcement. The view of Lab Info Page is very similar with the view of Student Page. The instructor can add new lab information to the lab list by filling in *Title* prompt, *Description* prompt and the *URL* prompt to identify the LabVIEW Server computer on the Add New Lab Page.

The instructor can delete or modify the lab information by selecting the lab number button at the Change Info list of the Lab Info Page. This makes a user move to the Modify/Delete Lab Page. The User Info Page shows the list of user IDs and the status of the user. This page also allows the instructor to modify the user information and to delete the user. The instructor adds a new user by entering the user ID, password and status at the three prompts labeled *User ID*, *Password* and *Status*, respectively on the Add New User Page. There is a text area labeled *Announcement* and a submit button on the Announcement Page. The instructor adds an announcement by simply typing at the announcement prompt and clicking on submit. The User Info Page, Add New user Page, and Announcement Page are similar in design to other instructor pages.

CONCLUSION

This paper presented the design and development of a virtual laboratory environment and a prototype experiment that allows students to perform experiments from remote locations using a web browser. The system serves as a communication tool between students and instructors, allows the secure access of students to experiments, and controls the view of Web pages by users. A front panel in LabVIEW displays the results and allows storage of acquired data for later processing. The implementation of a digital video camera that offers real time

images on the published front panel was also described. Employing the video camera to monitor experiments enhances the learning process. In the current experiment, for example, students can actually see the LED in the lab turning on when the measured temperature goes over a high temperature limit. The system is functional using either Netscape or Internet Explorer.

ACKNOWLEDGEMENT

This project is funded in part by the Air Force Research Office, award # F49620-01-1-0557.

REFERENCES

- [1] I. Gustavsson, "A remote laboratory for electrical experiments," *Proceedings of the 2002 ASEE Annual Conference and Exposition*, June 16-19, 2002, Montreal, CA.
- [2] M. Karweit, "Enhanced learning through virtual laboratory," *Proceedings of the 2002 ASEE Annual Conference and Exposition*, June 16-19, 2002, Montreal, CA.
- [3] M. Budhu, "A civil engineering materials course with a virtual laboratory," *Proceedings of the 2002 ASEE Annual Conference and Exposition*, June 16-19, 2002, Montreal, CA.
- [4] E. Eckhoff, V. Eller, S. Watkins, and R. Hall, "Interactive virtual laboratory for experience with a smart bridge test," *Proceedings of the 2002 ASEE Annual Conference and Exposition*, June 16-19, 2002, Montreal, CA.
- [5] M. Uskova, M. Bunch, D. Ludy, and V. Uskov, "Developing online educational materials for computer applications undergraduate courses," *Proceedings of the 31st ASEE/IEEE Frontiers in Education Conference*, Oct. 10-13, 2001, Reno, NV.
- [6] A. Striegel, "Distance education and its impact on computer engineering laboratories," *Proceedings of the 31st ASEE/IEEE Frontiers in Education Conference*, Oct. 10-13, 2001, Reno, NV.
- [7] C. Nippert, "Evaluating student performance in online laboratories," *Proceedings of the 2002 ASEE Annual Conference and Exposition*, June 16-19, 2002, Montreal, CA.
- [8] M. Naghedolfeizi, S. Arora, and S. Garcia, "Survey of LabVIEW technologies for building Web/Internet-enabled experimental setups," *Proceedings of the 2002 ASEE Annual Conference and Exposition*, June 16-19, 2002 Montreal, CA.
- [9] N. Swain, J. Anderson, M. Swain, and R. Korrapati, "State-space analysis of linear, time-invariant control systems using virtual instruments," *Proceedings of the 2001 ASEE Annual Conference Proceedings*, June 2001, Albuquerque, NM.
- [10] L. Sokoloff, "LabVIEW implementation of ON/OFF controller," *Proceedings of the 1999 ASEE Annual Conference Proceedings*, June 1999, Charlotte, NC.
- [11] R. Bachnak and C. Steidley, "An interdisciplinary laboratory for computer science and engineering technology," *Journal of Computing in Small Colleges*, Vol. 17, No. 5, April 2002, pp. 186-192.
- [12] J. C. Meloni, *PHP Fast & Easy Web Development*, 2nd Edition, Learning Express, 2002.