Virtual Laboratory Instruments and Simulations Remotely Controlled via the Internet

Miroslav Joler* and C. G. Christodoulou
Department of Electrical and Computer Engineering
University of New Mexico
Albuquerque, NM 87131
www.eece.unm.edu

Abstract

One of the problems of long distance education in engineering is performing laboratory experiments without actually being in the lab. That is perhaps the biggest limitation on the quality of any engineering program that wants to go the long-distance way. Most programs depend on simulations. Although this is possible, the student does not get the opportunity to get a very practical experience during the process. In this paper, we demonstrate how a student, from anywhere in the world, can operate a laboratory equipment such as a network or spectrum analyzer, residing in a remote laboratory, via a regular internet browser. This new approach allows several students, from different parts in the world, to run an experiment together, simultaneously. It also, can allow the sharing of equipment and resources among universities and other laboratories without the physical movement of equipment or researchers.

Implementation

Figure 1, depicts the concept of a Virtual Instrument (VI), controlling a real analyzer residing away from the virtual instrument (the replica). Since, the "student cannot come to the Lab", the Lab will come to the student. The aim here is not simply to monitor and control measurements from a distant location, but to achieve this remote control without, if possible, any special hardware or software. Hence, it must be operated on some type of software that is common to all today's PCs, in a network environment. Consequently, some of today's standard Internet browsers were found as a suitable solution. They are most convenient to perform this task because they are installed on every computer nowadays.

The client's (user's) browser is connected to an interactive HTML page on a server, built as a Java applet [1,5], which then becomes the client's interface for controlling an instrument. The final configuration of this communication scheme is shown in Figure 2. A host computer encompasses three units: an interactive web page(s) (Java applet(s)), a Web Server and a Data Server. The Web Server receives requests from a remote client computer for the interactive web page and serves the client with that web page (applet) if the client is eligible for that (security restrictions apply which is going to be mentioned in the next section). At the same time, by its Data Server part, the server communicates with the

instrument to be remotely controlled. The Data Server is in fact a virtual instrument that communicates with the real instrument. Connectivity between the host computer and the real instrument can be local (physical) or the instrument may even be connected to some other computer in which case the communication is handled through the network. Once the user receives an applet on his/her web browser window and a connection with the host Web Server gets established after some handshaking, the overall data transfer is managed by the Web Server. Commands and parameter values received from the client are through the virtual instrument transferred to the real instrument while measured data are being sent from the real instrument back to the client, analogously. Our Web Server [6] can serve a number of students simultaneously and the students can request different applets to control various instruments or processes at the same time.

Results

Using the communication scheme described above, a system was configured to remotely control a network analyzer. In our case, the analyzer has been connected physically to the server. To run the tests, the clients were connected to wired and wireless networks. They were connected to the analyzer via an HTML page set up to measure S-parameters of a sample microstrip antenna, as shown in Figure 3.
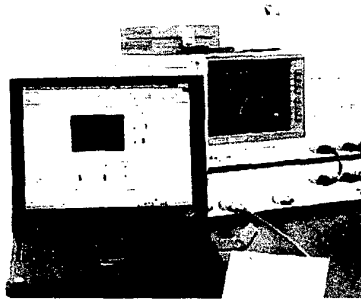

Figure 3   Remotely controlled S-parameters measurement from a wirelessly networked notebook

In addition to the instrument related functions, the interactive page, also, included a chat box to enable real-time discussion during the measurement. That way, all users can communicate during the measurement session in a practical and cost-effective way. Concerning any security aspects of the incoming connections towards the server and its parts, there is a way to check them out programmatically both in the Web Server and the Data Server part. We can arbitrarily assign an ID to each prospective applet to ensure that an applet can be connected only to its virtual instrument peer. This is especially important in the

case of repeating several processes simultaneously. If the ID does not match, the connection is denied.

Again, all connections and data transfers are managed by the server automatically. For example, if any user at any moment changes the value of any parameter on his applet, the server, in addition to sending that new value to the analyzer, also takes care of updating the applets of all the other users being connected to that applet. That enables all users (students) to follow any change made by their colleagues at the same time. Moreover, each user can open more parallel processes (simulations or instrument controls) on his/her computer as well.

Discussion and Conclusions

It was shown how a network analyzer can be controlled remotely using simple web browsers for educational purposes. Numerous users can, simultaneously, run multiple instruments or simulations. This definitely leads us to the *Virtual Laboratory* concept. Students can do lab in a more flexible environment where they can practice from distant locations, at their convenience, and with extended durations, unlike classic and strict schedules. Expensive equipment, usually inconvenient to be manually run by a large number of users, and untrained students, can now be offered in safer and more controllable way by use of its virtual replica - virtual instrument as an interface between a user and a real instrument. On the other hand, engineers, scientists, and project teams can easily share their test data, organize virtual meetings and workshops. People can share the instruments worldwide by putting their corresponding virtual instruments accessible on the Web.

References

[1] National Instruments Corporation™: LabVIEW™ 5.1 software and user manuals, 2000

[2] National Instruments Corporation™: ComponentWorks™, DataSocket Server 2.1 (445) software and documentation, 1998

[3] National Instruments Corporation™: ComponentWorks™, DataSocket Server Manager 2.0 (19), 1998

[4] Microsoft Corporation™: Microsoft Visual Basic 6.0, 1998

[5] Nacimiento Software Corporation™: AppletVIEW™ 1.1.15 software and user manual, 1999

[6] Nacimiento Software Corporation™: AppletReplicator™ 1.1.15 software and user manual, 1999

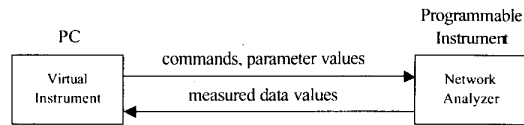[7] Hewlett Packard Corporation™: Network Analyzer 8753D™ manuals
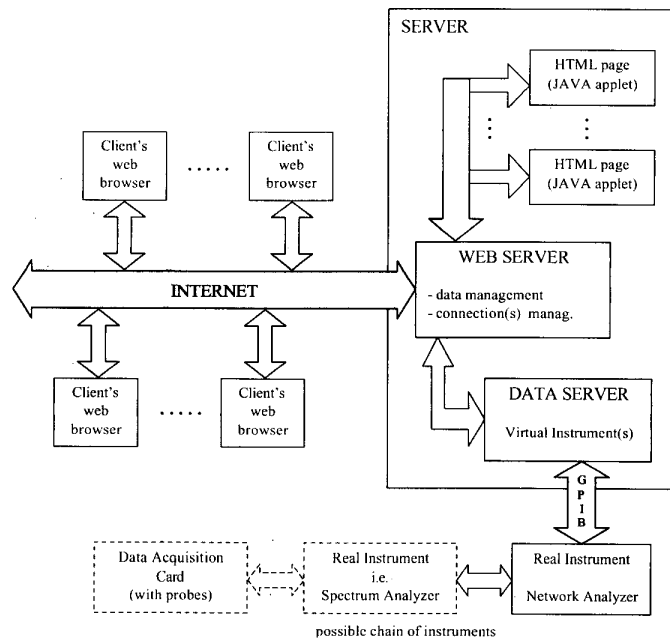
Figure 1   Virtual instrument controls the real instrument

Figure 2   Block diagram of the system