HAL open science

Open Architecture for Signal Processing Lab Distance Learning.

M.D. Steriu, Franck Luthon

To cite this version:

M.D. Steriu, Franck Luthon. Open Architecture for Signal Processing Lab Distance Learning.. 4th IEEE Digital Signal Processing Education Workshop, Sep 2006, Jackson Lake Lodge, United States. pp.305-310. hal-00404176

HAL Id: hal-00404176

https://hal.science/hal-00404176v1

Submitted on 18 Mar 2014

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

OPEN ARCHITECTURE FOR SIGNAL PROCESSING LAB DISTANCE LEARNING

*Mihai-Dan Steriu**

Politehnica University of Bucharest
Faculty of Electronics & Telecom
Dept. Electronic Circuits & Devices
Bd. Iuliu Maniu, Bucharest, Romania
dansteriu@yahoo.com

Franck Luthon

University of Pau and Adour River
Computer Science Dept., LIUPPA Lab
IUT Informatique, Château Neuf, Place Paul Bert
64100 Bayonne, France
Franck.Luthon@univ-pau.fr

ABSTRACT

We present a global architecture (software and hardware) for the remote implementation of practical laboratory works in the field of electronics, automation or signal processing.

The students can not only do simulations and virtual instrumentation, but they can also do real-world experiments in connection with the physics of the phenomenon under study, through observation or control. This is done remotely with TCP/IP network connection.

The client-server architecture is open in the sense that it is based on the plug-in concept, each pulg-in corresponding to a different lab experiment (e.g. automatic plot of Bode curves of linear systems, signal acquisition and numerical filtering...)

An operational prototype is implemented with LabVIEW: remote control of instruments (through the GPIB network), of data acquisition boards (DAQ, PCI or USB) and of switching matrices for electrical connections and power on/off, management of multi-user connection, multi-session, multi-experiment.

The advantage of such an architecture is the decrease of financial costs (sharing of expensive measurement devices) and the sharing of pedagogical means for really distant practical lab work (not only magistral courses or tutorial exercises).

***Index Terms*—** Distant learning, Lab work, LabVIEW, EEE engineering, Cybernetics education, Signal processing

1. INTRODUCTION

This work describes the design and implementation of a generic architecture (software and hardware) dedicated to electronics and signal processing lab distance learning. The laboratory is accessible via TCP/IP protocol network.

All technology resources in the nature (energy, materials and information) are taken into account in the learning ojectives of engineering education by remote control or monitoring (observation) [1].

*This work was supported financially by the University of Pau, with an invited professor position for three months in Bayonne, France.

The client-server architecture is based on the plug-in concept, one plug-in for each experiment available on the server side (Fig. 1).

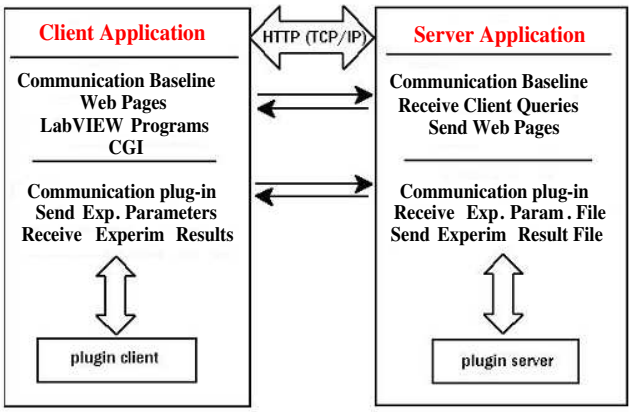


Fig. 1. Client-server open architecture : the client uploads experimental parameters and downloads results. The server executes client queries, manages user accesses and experiment lists for an optimal use of the hardware system

The student at home can not only do simulations of signals and virtual instrumentation as with J-DSP [2] (interactive online simulations for signal processing), but also do real (practical) measurements in order to characterize physical circuits and systems remotely.

Five types of client plug-ins are available for:

- experiments,
- simulations,
- web pages (start web browser with URL address),
- demos (for illustration during on-line courses) and
- others (e.g. help for report editing and production of deliverables, advanced representation and visualisation of data, quizzes).

As regards server plug-in, a template is also available that helps any authorized tutor to plug in easily a new experiment inside the menu (Fig. 2).

Fig. 2. Template plug-in

Applications are numerous: electronics lab [3], data acquisition and signal processing lab, image processing lab, control system and automation, supervision of controllable / programmable logic devices (PLD), standard communication interfaces (serial, parallel, USB).

2. SOFTWARE ARCHITECTURE OVERVIEW

2.1. Server application description

The main part is the G web server that works with different programs like in Fig. 3:

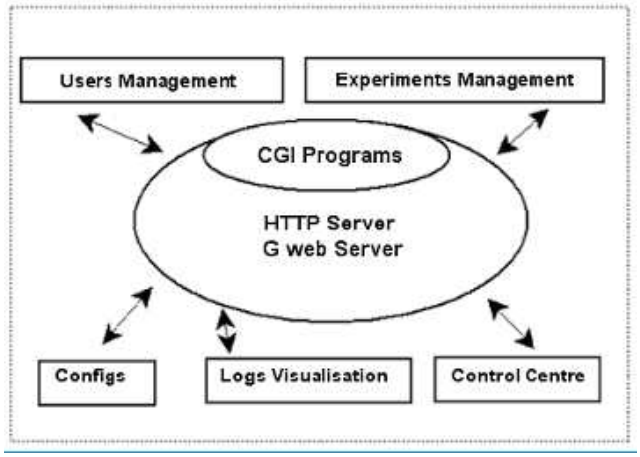


Fig. 3. Server application structure

The simplified actions of the server are presented in Fig. 4.

We use a G web server (HTTP server) with Internet Developper Toolkit and communicate through CGI.

The management of users access (multi-session, multi-user, multi-experiment) is handled by the control center on the server side, that is located near the real devices, systems and power sources. The server manages the user query list so that the hardware system is used in an optimal way. The main functions of the control center are (Fig. 5):

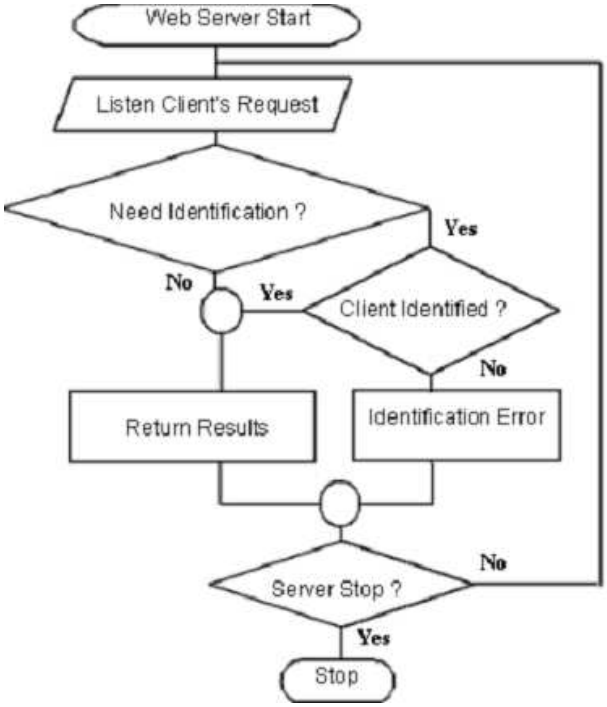


Fig. 4. Flowchart of server actions

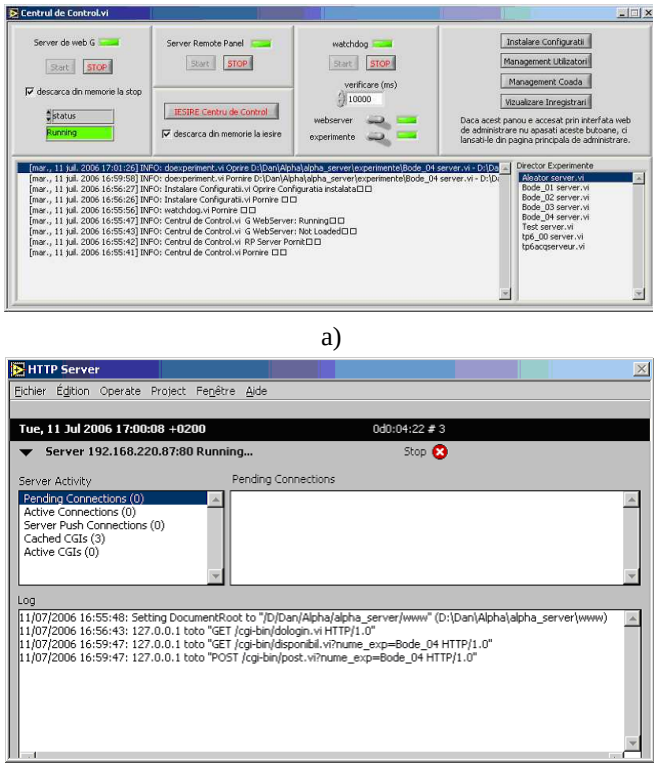
- start webserver,
- start remote panel server,
- start watchdog,
- install the experiment configuration,
- manage users (login, password),
- manage the list of pluggable experiments,
- visualization of logs, history, statistics of use,
- manage experiment queue,
- remote administration.

2.2. Client application description

A CDROM containing all the pedagogical content and the runtime version of the application is furnished to each remote client (student at home). The client application ("Virtual Laboratory", see Fig. 6) consists in an HMI for enabling connection and selection of learning mode (simulation vs. real experiment, off-line vs. online) and parameter setting.

The activity diagram of the client application is shown in Fig. 7.

The client sends to the server (uploads) the choices about the experiment, with configuration parameters. After execution of the experiment by the server, the remote client downloads the results of measurements. Then the student does



a)

b)

Fig. 5. Control center on server side: a) Left: webserver start ; Middle: watchdog start ; Right and bottom: configuration and management ; b) HTTP server running

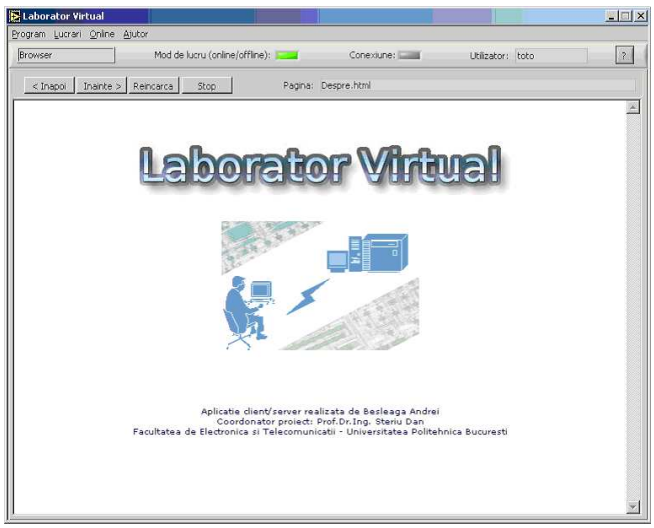


Fig. 6. Remote Laboratory Client Application

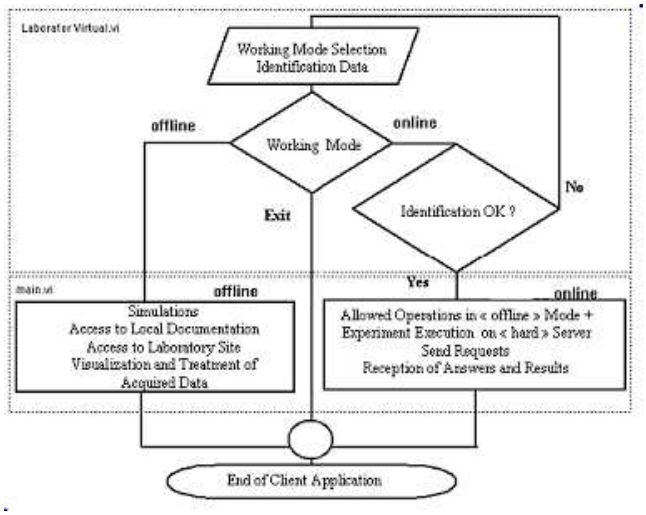


Fig. 7. Flowchart of client application

analysis and interpretation of experimental data in connection with the theory and models in agreement with learning objectives: plot curves, validate theoretical models, and generation of the report that will be evaluated by the teacher.

2.3. Server plug-ins

To make easier the programming of the server plug-ins, we propose a structure which permits to work in two situations: **`Local`** (the settings for experiments are made on the server) and **`Client`** (the settings come from a remote client, Fig. 8). Also it is possible to obtain the results from a real experiment (switch on **`Manip`**) or from a simulation module (switch on **`Sim`**). Indeed it is easier to program and verify a server plug-in working without a client connection.

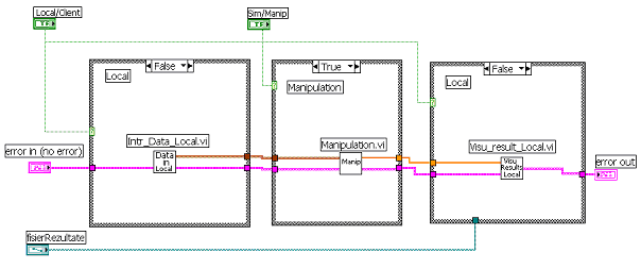


Fig. 8. Server plug-in diagram in verification phase of the experiment: Local=False (*i.e.* remote client) and Manip=True (*i.e.* real experiment, not a simulation)

`Intr_Data_Local.vi` is used for setting the experiment locally, **`Visuresult_Local.vi`** , to visualize the results locally and **`Manipulation.vi`** to command the apparatus.

The current working situation (request by client → real experiment on server → results to client) is presented in Fig. 9.

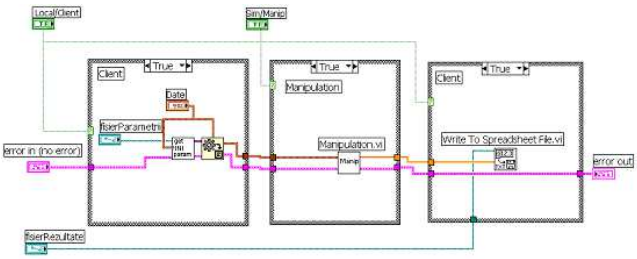


Fig. 9. Server plug-in diagram in the current working situation

Also, it is possible to answer to a client's request with the results of a simulation, using precise mathematical models of the devices. It is very useful when there is maintenance running on the equipments.

3. PROTOTYPE IMPLEMENTATION

The prototype is based on LabVIEW which is a graphical programming environment for measurement and testing. It offers easy to use programming by wiring graphical objects togheter and human-friendly interfaces, yielding fast prototyping cycle in the development of applications [4].

The application runs on PC with Windows system, Internet access, web browser and local GPIB bus (IEEE 488 standard for instrumentation). The measurement devices (waveform generator, programmable power supply, oscillscope, multimeter, data acquisition card) and the switching matrices (multiplexing for connection configuration and power on/off) are controlled and run remotely by the student connected through Internet to the server that is located at our institute (Fig. 10).

4. APPLICATION EXAMPLE AND RESULTS

An operationnal multi-user, multi-experiment prototype was implemented and tested successfully: 5 to 30 simultaneous connections, plug-ins validated for various experiments among which data acquistion with signal filtering, and automatic plot of Bode curves (gain and phase) to characterize the transfer function of a linear system (electronic circuit, see diagram schematics on Fig. 11).

The measurement setup for Bode plots is shown in Fig. 12. The estimation of the phase requires the use of an oscillscope for the measurement of the time delay between the input and output signals. Note that such a device is much more complicated to program than a simple multimeter. That is one reason why we choosed to implement this experiment, in addition to the fact that the study of the transfer function of 2nd order

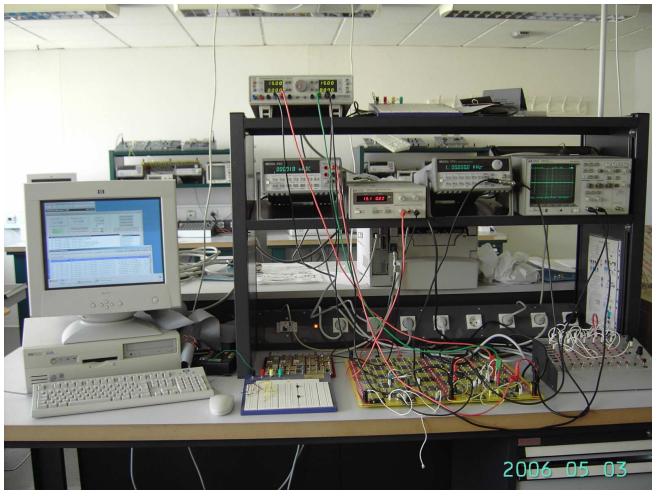


Fig. 10. Experimental setup in server room: server PC on the left, GPIB instruments on top and connection board with DUT (device under test) on the table

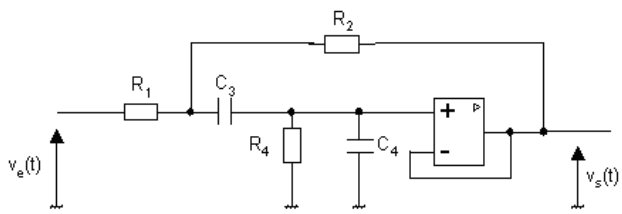


Fig. 11. Bandpass Sallen-Key 2nd order active filter to be characterized

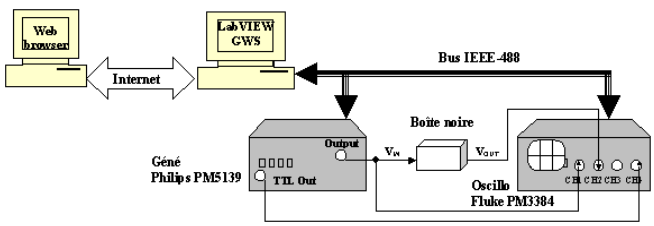
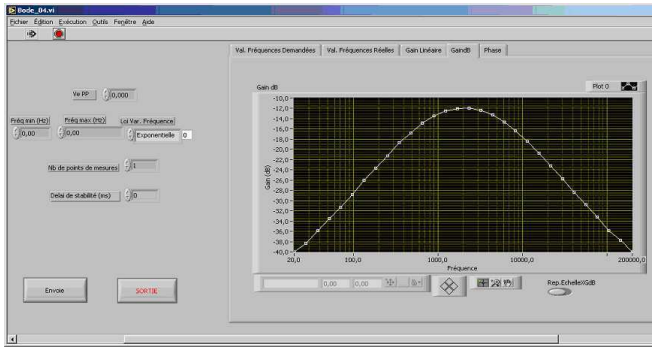


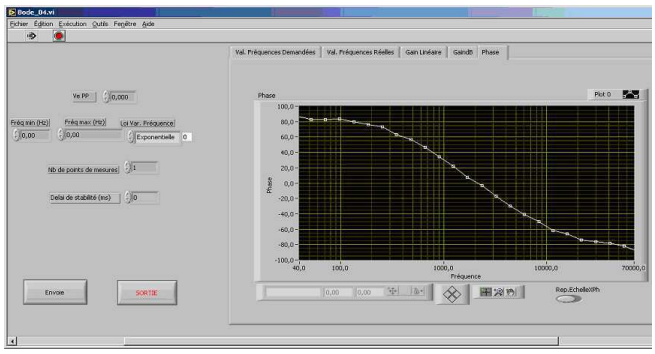
Fig. 12. Instrument setup for gain-phase Bode plot

filters or systems is of major interest for EEE students at the undergraduate level.

After connection, the student chooses the experiment (Bode plot) and the parameter settings (minimal and maximal frequencies f_{min}, f_{max}, input voltage V_{in}, number of samples N_s, display mode (linear, dB, decade). The watchdog allows to inform the client of data ready available for download (Fig. 13).



a)



b)

Fig. 13. Download of results with display on the client side: a) Bode curve of the Sallen-Key 2nd order filter: Gain in dB; b) Plot of the Phase variation of the filter

5. DISCUSSION AND FUTURE WORK

The advantages of such an architecture are the following: reduction of financial costs for the institution (one saves money because a single item of each expensive equipment in the field of electronics and automation can be used for many students). It enables distance learning courses that encompass the entire learning process, including practical work, with 24 hours $\times 7$ days access.

Other advantages are speed-up in measurements, reduction of repetitive tasks and errors, leaving more time to the student for interpretation, unlimited access to experiments, on-line demos, easy development and maintenance by team work, etc.

This open architecture is extensible (thanks to plug-ins), scalable (LAN, WAN, etc). It allows the open distribution of training material since any tutor can build a new plug-in and add it into the menu.

In addition to our two current experiments (Bode curve plotting and Data Acqusition and Filtering) taht are already plugged in the prototype, we intend to plug new experiments that we have already developped separately [3]:

- charaterization of an audio LF power amplifier (Fig. 14),

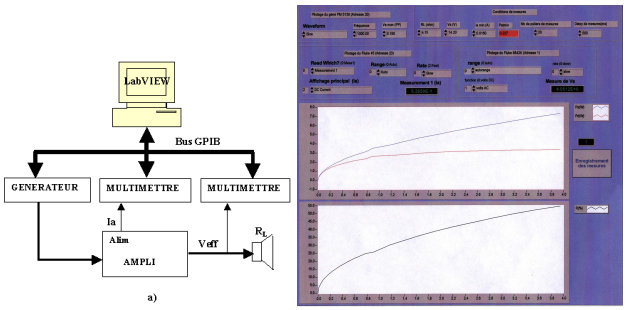
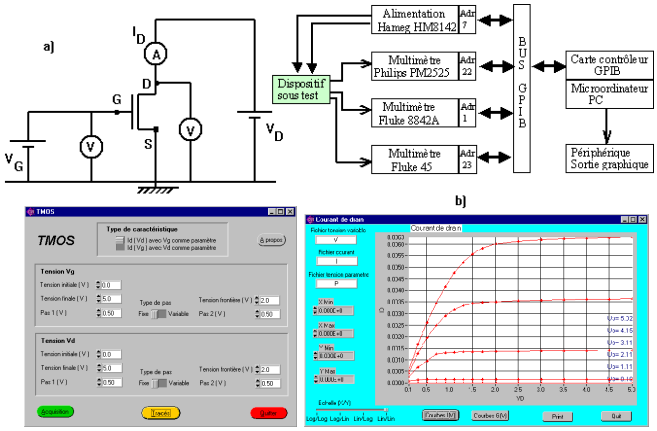


Fig. 14. Characterization of a LF audio power amplifier: a) Experimental setup ; b) Graphical user interface that displays the power and efficiency curves

- characterization of a MOS transistor (Fig. 15)



c) d)

Fig. 15. TMOS characterization: a) Device under test ; b) Measurement setup ; c) Front pannel ; d) Graphical user interface with display of curves of drain current vs voltage

Presently, the prototype is validated from a technical point of view (remote real experimentation and network communication) but we still have to calibrate the learning objectives and to evaluate the prototype with many more students (to assess the pros and cons of real experimentation vs. simple simulation as regards learning goals).

For experiments requiring real-time (for example live video showing in real-time the physical phenomenon in motion or action), we are currently investigating what choices have to be changed in the proposed architecture (web control by CGI vs. ActiveX vs. Java ; DataSocket vs. VI Server vs. TCP VIs).

Future work will also concern the evaluation process (student evaluation form, quizzes, constructivist approach [5] etc), and the use of the new high speed standard HS-488. Comparison and better positioning with respect to other interesting methods and works [6, 7, 8, 9, 10, 11, 12] is also under investigation.

6. ACKNOWLEGMENT

The authors thank Thierry Nodenot and Philippe Lopistéguy from University of Pau, Bayonne, France, for fruitful discussion and references.

7. REFERENCES

[1] L. D. Feisel and A. J. Rosa, "The role of the laboratory in undergraduate engineering education," *Journal of Engineering Education*, pp. 121–130, Jan. 2005.

[2] A. Spanias, "J-DSP Java Digital Signal Processing and On-line Laboratory," www.eas.asu.edu/∼ midle/jdsp/, 2006.

[3] F. Luthon, "Instrumentation programmée pour TP d'électronique distant," in *Actes de la Conférence FuturVIEW 2003*, Futuroscope, Poitiers, 12-13 Juin 2003, pp. 263–268.

[4] J. Travis, "Distance learning solutions guide," Tech. Rep., LabVIEW National Instruments, 2000, ni.com/academic/distance_learning.htm.

[5] J.M. Ferreira and D. Mueller, "The MARVEL EU project: A social constructivist approach to remote experimentation," 2004, http://www.marvel.uni-bremen.de/.

[6] D. Müller and J. M. Ferreira, "Online labs and the MARVEL experience," *Int. J. Online Engineering*, 2005, www.i-joe.org.

[7] J. Garcia-Zubia, D. Lopez de Ipina, and P. Orduna, "Towards a canonical software architecture for multi-device weblabs," in *IECON*, 2005.

[8] Javier Garcia-Zubia, Diego Lopez de Ipiña, and Pablo Orduña, "Evolving towards better architectures for remote laboratories: a practical case," *International Journal of Online Engineering (iJOE)*, vol. 1, no. 2, 2005.

[9] T. Bouhadada, M.T. Laskri, T. Ghrib, and A. Chouaf, "SAPIENCE: A simulator of interactive pedagogical activities for distance learning environment," in *Proc. Conf. INFORSID*, Hammamet, Tunisie, June 2006, pp. 1–16, Hermès Science Pub.

[10] N. Masseux, "Modélisation d'une interaction didactique distante individuelle synchrone (IDIS)," *Revue Sciences et Technologies de l'Information et de la Communication pour l'Education et la Formation (sticef.org)*, vol. 10, Nov. 2003.

[11] C. Bonivento, L. Gentili, L. Marconi, and L. Rappini, "A web-based laboratory for control engineering education," in *Proceedings of 2nd International Workshop on Tele-education using Virtual Laboratories*, Sherbrooke, Canada, August 8-9 2002.

[12] C. Chiculita and L. Frangu, "A web remote control laboratory," in *Proceedings of 6th Multiconference on Systemic, Cybernetics and Informatics*, Orlando, Florida, July 14-18 2002.