

Tom 49(63), Fascicola 2, 2004

## A Toolkit for Internet Based Distance Laboratory Development

Sebastian V. Tiponut<sup>1</sup>

**Abstract** – This paper describes the design and implementation of a distance laboratory, suitable to be used in a large range of experiments with proper customization. Using the well-known client-server paradigm, this design tries to accomplish three goals: simplicity of design, future extensibility and ease of use. This implementation differs from other existing distance laboratory projects in the following way: Firstly, it can be interfaced with a wide range of experiments. Secondly, the core of this system has been designed using XML for a greater flexibility. Finally, it was written to allow further development and enhancements.

**Keywords:** distance learning, distance laboratory, Internet, education

### 1. INTRODUCTION

Experimentation is the cornerstone of the modern science, a way to convince others of the correctness of a particular scientific theory or hypothesis. Even if experimentation is of such importance, one may doubt the reasons for which students are reenacting experiments with already well known results. To enhance the learning techniques, provide more insight and consolidate the knowledge on a particular topic the best way is to let the students do the experiment themselves. Most of the time, theory is too dry and academical abstracts are just not enough to provide a thorough understanding of a certain matter. Thus, real life situations are ideal to supplement the theory, teaching the student how to interpret and correlate experimental results with a particular hypothesis. It is very important that, besides the experiment goals, the art of experimental designs be communicated to the student through the laboratory experience.

Distance laboratories are a new appearance in the field of distance learning. They are meant to provide a hands-on approach, allowing experiments to be carried out from remote locations rather than requiring the personal presence of the human operator. As with any new technique, many advantages and drawbacks have already been spotted. To be successful and to prove itself as a viable alternative to classic experiments, distance laboratories must offer more benefits than

deficiencies. Distance laboratories do not require synchronization between the instructor and the students; experiments can be accessed whenever the student needs them. Safety issues are also eliminated because dangerous equipment is not physically handled and unsafe experiments are performed remotely. Distance laboratories can also nullify the effects of space and time allowing access to experiments from a particle accelerator, nuclear research center or to obtain weather or seismographic data. Among the limitations that a distant laboratory may suffer are the lack of control of the experiment due to poor design of the user interface, need of technology standards for software and hardware and the intrinsic technical difficulties in designing such a complex system.

Different approaches to designing distance laboratories have been taken. National Instruments [12], the company which is producing the Labview software has designed a toolkit which, in conjunction with Labview software, can be used to create user interfaces meant to control distant experiments [2], [3], [4]. A few distance laboratories have already been deployed using this toolkit: Cyberlab from Stanford University, the Virtual Laser Laboratory from Dohlouise Laboratory or the Lecture Enhancement from Swiss Federal Institute of Technology. Another example, presented in [1], uses Java applets to create a graphical interface and custom client-server design to connect the interface to the experiment. [5] describes the remote access of a chemistry experiment using a Web interface to collect data, display and analyze the results.

This paper introduces a new design for such a distance laboratory. Instead of concentrating on implementations, we choose to detail the parts involved in this system, the communication protocol and the interfaces between them. This will help others to build similar laboratories - using other programming languages, on different platforms and targeting a broad range of experiments. Our approach boasts the benefits of a standardized system which can

<sup>1</sup> Inginer, S.C. Spidernet S.R.L. Calea Bogdanesti 6A  
Timișoara, e-mail seba\_tiponut@spidernet.co.ro

undergo several implementations while keeping information exchangeability and compatibility.

## II. GENERAL PRESENTATION OF THE SYSTEM

The proposed distance laboratory system consists of seven elements which are interacting using a set of predefined, well-known, interfaces. The core of this system is a client-server structure. The client is represented by a graphical user interface which communicates with the server using predefined network routines: its function is to display collected data. To mediate the concurrent access to the physical experiment of a large number of clients, the server part has been designed. It comprises of networking routines, decision code and a user-written driver whose purpose is to interface the server with various data acquisition systems which collect data from and control the experiment.

To make the graphical interface more adaptable to different types of experiments, a predefined, custom set of widgets has been designed. This widget set - which is supposed to be extended and even standardized - contains from simple primitives like buttons, frames to complex objects in form of gauges, 2D or 3D displays, sliders etc. In order to facilitate the construction of various graphical interfaces in a very simple and platform independent manner, the graphical interface is described in XML. An XML parser is then building the interface based on this file, instantiating the widgets from a library. One can associate an experiment with an XML description of a graphical interface, customizing the latter to suit experiment's needs.

One server can host multiple experiments, possibly different experiments. Because of that they must be controlled using distinct graphical interfaces. Thus a need arises to associate a particular experiment with the correspondent XML description of the graphical interface. This association was done using a very simple HTML-like page where, modeled after the World Wide Web. The path of the graphical interface description is associated with a hyperlink which, if clicked, will start the downloading process of that particular XML description. It also supports some primitive form of text formatting. A special browser is used to display the page downloaded from a separate server. The user can click on the hyperlink, have the XML description downloaded and then the graphical interface built from it. Once the graphical interface is generated the experiment may start. A general view of the system is depicted in Fig. 1.

As stated in the Introduction, this design emphasizes portability and standardization. Thus an implementation must comply with the specified protocols, file formats and interfaces. There are two communication protocols: the first one is mediating

the communication between the graphical user interface and the experiment; the second is a very simple file transfer protocol used to retrieve the page containing the list of the available experiments. The former protocol has double purpose: it transmits the commands issued from the graphical interface and it transmits back the data from the server.

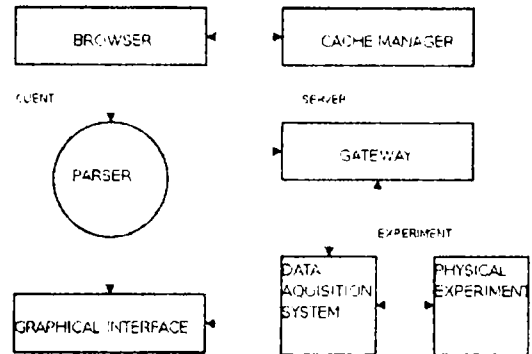


Fig. 1 General view of the system

The file format for the page listing the experiments along with a short description is an XML page which contains tags which specifies the location of the graphical interface description on the server. It specifies for each experiment the experiment identification and an arbitrarily long description of the experiment itself.

At the core of creating the user interface stays the XML file containing its descriptions. It is composed of tags, each tag describing a widget along with its characteristics. Some widgets rely upon the presence of others; thus, the XML parser is doing a check to assure no mistakes have been made during file editing. The library of widgets content will be described later on, in the section dedicated to implementation.

The physical experiment is connected to the server via a data acquisition system and a driver which is mediating between the two. Because the data acquisition system can be connected to the computer using a serial, parallel, I2C or other type of connection, the driver must be designed to handle these particularities. Another issue the driver must solve is the communication protocol with the data acquisition system.

## III. PRESENTING AN IMPLEMENTATION

The author has chosen to implement a distant laboratory using the design presented above. The programming language in which the whole distant laboratory was written is Python. It has been selected because it is a very powerful object oriented programming language, supporting well designed XML and general purpose libraries. Configuration

files, written in XML (for ease of parsing), have been designed for all the components.

The heart of the system is the parser which is generating the graphical interface by interpreting an XML file containing the interfaces' description, as presented above. This parser is a standalone program which is usually called within the browser once an XML file with the interface description has been downloaded; it can also be ran from console, passing the file to be parsed as its first argument. A minimal set of widgets have been designed to allow the construction of a graphical interface suitable to be used for basic experiments. Among the widgets included in this set are:

- a graphical XY display; one can customize the number of ticks and the start and stop values separately for each of the two axes;
- a sliding bar which can be used to input values; the start/stop values and the increment are customizable;
- an amplifier which multiplies the values outputted by the sliding bar with a certain, configurable, factor;
- buttons which can be used for a variety of operations;
- frames, used to group the other widgets together.

The server which is serving the user interface XML descriptions is called the *cache\_manager*. The server which is used by the user interface to mediate the communication with the hardware is called the gateway. Both servers have XML configuration file.

#### IV. THE PILOT EXPERIMENT

This is a simulated experiment which is supposed to test the distant laboratory whose implementation was presented above. A piece of software is meant to substitute the data acquisition system and the physical experiment. This is simulating the characteristic of the diode which can be obtained by applying an increasingly high voltage between the anode and the cathode of the silicon diode. The use inputs different voltage values which will be sent to server for evaluation. The simulation plugin will compute the associated current value and it will reply with pair of coordinates which represents a new point on the graphic.

After the graphic has been drawn it can be saved for later use. Saving a graphic is also saving the environment (the number of ticks and the values from both axis and the measurements units).

#### V. DISCUSSION

This paper presents the design of a distant laboratory along with a possible implementation. One of the

goals the author had in mind when conceiving the structure of this system was to make it versatile for a wide range of experiments. The widget library can be extended with new types of widgets which can be used to broaden the types of experiments for which this model can be used.

Another goal which was targeted was to make the system as portable as possible. That is why it was designed as an open project the interfaces and the protocols between the components are presented in detailed so that one can implement a compatible system or just a part of it. While the system could have been designed platform independent and distributed as a software bundle we rather wish to emphasize the benefits of a known protocol and an open design schematic.

Last but not least, the system was thought to be open to improvements and modifications while maintaining backward compatibility. This makes this design very different from existing implementation of distance laboratories which are - the vastness majority - close source and very hard to develop and to integrate.

The implementations decisions are also making this distance laboratory different from other distance laboratories. It is using XML as its core for both creating a graphical interface for a given experiment, communication between components and configuration files. The servers are designed such way that they can mediate the access of a large mass of users to the available experiments. Using the Python language for all the components offers some degree of portability. The system was conceived to be used on a Unix machine but with minimal changes it can be ran on other platforms as well.

#### REFERENCES

- [1] T. Chang, D. Hunt, "Web-Based Distance Experiments: Design and Implementation" 2000 *International Conference on Engineering Education*, Taiwan.
- [2] J. Henry, "LabVIEW Applications in Engineering Labs". *ASEE Conference Anaheim*, California, June, 1995.
- [3] J. Henry, "Controls Laboratory Teaching via The World Wide Web", *ASEE Paper*, June 1996.
- [4] J. Henry, "Internet Laboratory Server in Engineering Systems Laboratory", *Conference at M.I.T.*, June, 1997.
- [5] H. Shen, Z. Xu, V. Kristiansen, O. Strom, M. Shur, "Conducting Laboratory Experiments over the Internet", *IEEE Transaction on Education*, Vol. 42, pp. 180-185, August, 1999.
- [6] R. Berntzen, J. Strandman, T. Fjeldly, "Advanced Solutions for Performing Real Experiments over the Internet", *2001 International Conference on Engineering Education*, Norway.
- [7] B. Aktan, et. al. "Distance Learning Applied to Control Engineering Laboratories". *IEEE Transactions on Education*, v37, No. 3, pp. 320-326.
- [8] D. Knight, S. DeWeerth, "A Distance Learning Laboratory for Engineering Education"
- [9] J. A. del Alamo, et. al. "Educational Experiments with an Online Microelectronics Characterization Laboratory"
- [10] E. Bobkov, Y. Sheynin, "The Web-based Technology in Laboratories for Distance Learning and Training"
- [11] J. C. Piower, et. al. "Web-Based Educational Experiments"
- [12] [http://www.ni.com/academic/live\\_experiments.htm](http://www.ni.com/academic/live_experiments.htm)