Practical experiences on building structured remote and virtual laboratories from the student’s point of view

Rafael Pastor, Daniel Sanchez, Nourdine Aliane, Roberto Hernandez, Antonio Robles-Gómez, Agustín Caminero, Salvador Ros, Gabriel Díaz, Manuel Castro

Abstract – Nowadays, the use of remote laboratories is a common feature in order to get students involved in practical experiences. This is particularly important in distance learning environment. The regular practice is to develop a particular laboratory, required by the student competence’s curricula. Usually, this involves a lack of methodology or a standard procedure so, new developments must be done to include new laboratories, additionally so many “manual” procedures are required. RELATED (REmote LABoratory Extended) it’s used to solve these issues. In this paper, a practical experience of the structured development process of a virtual laboratory, using RELATED framework, will be presented from the point of view of students. This virtual laboratory is a simple signal generator. This signal generator will be used to generate a reference position for the height of the ball of an electromagnetic levitation system in a control experiment. Finally, responses of surveys made by students will be presented, in order to get satisfaction results of using/development of virtual/remote laboratories using RELATED.

Index Terms - remote experimenting, virtual and remote laboratories, software development, services and facilities, practical experience

II. INTRODUCTION

Acquiring qualitative knowledge is one of the most important things in control engineering education [1-3]. Giving students a taste of real situations, the behavior of measure instruments and how does the control action work, is a vital aspect nowadays. All these things can be done with laboratory experiments although it can be tricky with a large amount of students or in distance learning environment, as it is proposed by UNED (National Distance University of Spain) in its official degrees.

The idea is for the students to be able to perform real experiments [4-5], in real time, on real equipment, but over the Internet. This idea is in fact a widely discussed issue. In this general context there are many works, but all these remote and virtual laboratories are punctual efforts of different research groups. The use of the software and hardware of different universities is not contemplated in any case, not taking advantage of the work previously carried out by others. This is to say, a lack of methodology or a standard for the construction of networks of virtual/remote laboratories based on previous developments. Also, services and facilities for all of these laboratories are not present.

RELATED framework [6] proposes a structured methodology of remote/virtual labs development and, also, provides common facilities as user management, booking, basic visualization (trend graphs and direct interaction using interactive variables), data logging and experimental session’s control. A RLAB (Remote LABoratory) system is defined using a formal specification (which is LEDML, based on XML). The main component is an experiment, which is defined on the laboratory XML specification. To integrate a laboratory as a RLAB system is only necessary to develop local access to laboratory equipment using Java technology and the “module” paradigm. These modules, which are run-able entities, are started by the RELATED facilities in order to get/set data from/to the laboratory equipment. This data will be sent over Internet to the RELATED client too.

As users need graphical interfaces in order to understand and operate with the remote/virtual labs, RELATED framework provides the concept of “view”, which is a coded entity with the purpose of showing widgets and other GUI components. RELATED also provides built-in graphical user interface facilities, defined in a basic way, as it will be shown in the next sections.

In the next section, the fundamentals of RELATED will be shown. Also, it will be shown how a laboratory can be described in terms of its components. This section will show the development cycle and the facilities which are automatically part of the framework. In Section III, the magnetic levitation system will be presented in order to apply the development cycle using RELATED methodology (Shown in section IV). In Section V, student’s development procedure and results are explained and showed, and how their results are used as the building of a remote experiment using the remote levitation lab. In Section VI, results of a survey performed by students are detailed. Finally, conclusions will be presented in section VII.
II. RELATED BASICS

RELATED provides a structured way of developing remote and virtual laboratories, focusing in lab definition and built-in facilities. To achieve this objective, these are the steps which are defined in the methodology:

A. Defining laboratory structure

Structuring the lab is one of the main tasks a developer/teacher must do [7]. There are three main components in RELATED which are necessary to describe the laboratory:

a) Modules. They are the basic part of development and provide virtual/physical access to lab data. They are “black boxes” used to describe the component as a set of input/output variables. This paradigm allows the interconnection of modules, being possible to flow data between a module and others. Developers must provide read/write function to get/set data for these variables.

b) Views. They provide the visual information to the final user (students, teachers, etc.). GUI can be included in these views. These views use data from modules to update the experiments state, and it is possible to send to modules values changed in the (Graphical User Interface).

c) Experiments. They define the behavior expected from the lab, so one or more experiments can be defined on the lab. Basically, it is possible to define an experiment by stating the set of modules and views that will be used in the experiment. Any convenient combination of modules and views can be used as an experiment.

In order to get a full description of lab’s components, a description file based on XML must be defined. In Fig. 1, the levitator example (explained in the next sections) is shown. Fig. 2, 3 & 4 show detailed components definition for a module, view and experiment in the remote lab.

B. Developing coded entities

Once the labs components and views are defined, a full implementation of modules and views must be coded in order to assign the attribute/tag <implementation> to these entities. To “pack” the modules for RELATED runtime use, it’s mandatory to create a jar file with a main class implementing the IRLABModule java interface, which define several restrictions needed to integrate the module with RELATED.

C. Publish laboratory

With the XML definition (laboratory structure) and implemented modules and views, the next step is to “publish” the lab (and its experiments) in the RELATED Web Server. For doing this and application is provided (RLAB Publish Application) in order to validate the XML file and publish the laboratory. In Fig. 5, the RLAB Publish Application shows how the XML for levitator system has been correctly parsed and published on RELATED Web [8].

Once the laboratory is registered on main server, the lab is available to registered users, who have authorization, in the RELATED Web Server. Managers of laboratories can add permissions of use to the registered users in the RELATED web server, using authorization system associated to every laboratory published.

add permissions of use to the registered users in the RELATED web server, using authorization system associated to every laboratory published.

D. Run Experiments

The experiments can be started through the RELATED Web Server simply clicking on the experiment name. A JWS (Java Web Start) application will be downloaded and, if the Internet navigator is properly configured, the login control facility will start automatically (Fig. 7). User login data...
must be filled in order to show the experiment control panel (see Fig. 8).

The experiment control panel provides automatically some facilities as:

- **Booking slot time**: For every experiment, a slot time is assigned to avoid multiple concurrent users. It is represented on the upper left corner of the experiment control panel, using a three color state clock. Green represents free time for using experiment, yellow indicates less than 10% of the booking time and red color is used to indicate the finishing of slot time.

- **Experiment time**: Values of experiment starting time and actual time are presented to users in order to know time evolution for its experiment.

- **Graphical trends**: In order to add this feature, there is the possibility of adding special `<paint>` tags in the experiment definition. It is possible to group several variables into the same tag, which is useful for related variables. The graphical trends are presented in the lower left part of the Experiment Control Panel.

- **Interactive variables**: Usually a view is used to change some variables in one or more modules but, in simple experiments it is possible to define a set of variables which can be modified directly from the Experiment Control Panel. There is a special tag (`<interactives>`) for doing this. The interactive variables are usually presented on the lower right corner of the Experiment Control Panel.

Once the experiment control panel is shown, users have to press the “Start experiment” button in order to run it. After pressing the “Start experiment” this button changes for a “Stop experiment” button. In order to stop the experiment, users need to press this “Stop experiment” button.

III. LEVITATOR ON-SITE LABORATORY

Quanser [9] has on-site didactical equipment called MAGLEV (MAGnetic LEVitation system), see Fig. 9(a). This equipment provides a magnetic field in order to get an steel ball levitating in air. The MAGLEV system consists in a rectangular enclosure which has three distinct sections.

First, there is a solenoid coil with a steel core, which provides the magnetic field. In second place, there is a chamber, with white walls, where the magnetic ball suspension takes place. And, in third place, there is a bottom place where a photo-sensitive transistor (with conditioning circuitry) measures ball position, also it has offset and gain potentiometers for proper calibration. There is a cylindrical post inside the middle chamber, which is empty in the inside, where the phototransistor is situated.

The MAGLEV system provides two experiments to be carried out: coil current control and ball position control. Both of them are based on the assumption that students will run these experiments in the same “location” as the equipment is laid. It’s necessary to do some modifications in order to get a real remote lab.

The next section will show how it is the MAGLEV reconfigured and the control software is reused using the RELATED framework.

![Figure 5. RLAB Publish Application](image)

![Figure 6. RELATED Web Server: Laboratory list for an user.](image)

![Figure 7. User login in the client application.](image)

![Figure 8. Experiment Control Panel](image)

IV. REMOTE LAB FOR LEVITATOR

In order to reconfigure the on-site lab hardware, the first step is to avoid manual procedures associated with the on-site experiences. The main problem occurs when the ball falls outside the post. The solution is to surround the post with a plastic cylinder, in this way the ball never falls out. The next step is to define MAGLEV laboratory and to
reconfigure software procedures, as it is explained in the next sub-sections.

A. Laboratory specification

They are needed two modules for the two types of control experiences.

a) **PI_MODULE**: Coil current control experience which has several variables like coil current, voltage, command signal, and so. It is based in a PI (Proportional-Integral) controller, which parameters are named Kc_p and Kp_i.

b) **PIV_MODULE**: This is the ball position control experience, and the controller it’s a bit more complicated than the anterior module. It’s based in a feed forward plus velocity controller, and it has also the PI current loop control. Like the previous one, it has all the typical variables like coil current and so, and the specific ones of the controller (Kff_b, Kp_b, Ki_b and Kv_b).

Also, two auxiliary modules are used: one associate to a signal generator for setpoints (SG_MODULE) and another associated to the video streaming (VIDEO SERVER module) which allows getting a real time image of the MAGLEV system. Additionally, to get a more detailed view of the remote MAGLEV system, two “views” are defined: one to represent the MAGLEV system (see Fig. 11(b)) and another to view a real time image of the steel ball.

Finally, a set of experiments can be defined using these last modules and views:

a) **Current control**: This experiment represents the coil current control experiences and its definition includes the use of PI_MODULE & VIDEO SERVER modules and the two views (ImageViewer and MAGLEV Virtual View).

b) **Ball position**: In this case, the modules which are used are PIV_MODULE and VIDEO SERVER module.

c) **Ball position (generator)**: Similar to the above experiment, but also include the SG_MODULE module in order to use the signal generator for the ball position.

B. Modules development

QUANSER provides QUARC software, which are composed of several modules, including some based on Matlab/Simulink models. As the Simulink models are implemented for on-site experiments, they need some modifications. The model needs to be modified with network specific blocks which allow get/set information from the model (see Fig. 10).

C. Views development

There are two views defined in the XML file. One of them is the virtual view [10] for MAGLEV system (see Fig. 9 (b)). The other view is associated to the video streaming of images from a network camera. Really, the view is supported by the use of the VIDEO SERVER module.

D. Final steps

Once the development steps are done, the final step is to publish the MAGLEV laboratory in the RELATED Web Server (as it was mentioned before, in section II).

V. STUDENT’S VIRTUAL/REMOTE LAB.

The practical experience which must be carried on by students has two different parts: development of a virtual signal generator and integration of the virtual lab with the remote lab (levitator). Students are enrolled in the subject called Distributed Systems on the Computer Science Degree at the National Spanish Distance University (UNED).

A. Development of the virtual lab.

In this part, first, students must familiarize with the publish process of a RELATED lab. For this objective it is used a predefined lab with its code available for the students (a simple random generator of 4 values). Next, they must develop a module for the signal generator and a simple view to change type, frequency and amplitude of signal generator. The development process is based on the procedures explained in Section II, in a similar way but more simple than the levitator system development. Fig. 11 details the developed view by student and the output computed in the REALTED client application.

Once the virtual lab is developed, a module component is available for reusing in other laboratories (module corresponding to signal generator). So, the next step in the practical experience is its integration with a remote lab (the MAGLEV system).
B. Integrating the virtual lab as a remote lab.

In order to create and publish the student’s remote laboratory, the only task which must be done is the modification of the virtual lab XML specification (all the code is developed). This modification includes the definition of an experiment (tag marked in green in Fig. 12) called “Remote experimentation”. It is defined in a similar way that the virtual lab developed in the previous section, but the experiment must include references to the corresponding module in the remote lab (called MAGLEV). These references are located in the experiment definition:

1) First reference corresponds with the inclusion of a <run> tag which defines the remote execution of the PIV Module of MAGLEV system (this module get/set access to the data generated by the real equipment). It is marked in red in Fig. 12.

2) The second reference must be included in order to connect the output of the signal generator module with the set point for ball height defined in the remote module. This is done by using the <out> tag inside the <run> tag associated to the inclusion of the signal generator module in the experiment (to execute during the experiment). It is marked in blue in Fig. 12.

No more development must be done, so students simply need to publish the new remote lab. In Fig. 13 it is shown the views associated to the remote experiment running. As part of the experiment, a real time video image is presented to the user (represented as a module/view, and developed for the MAGLEV system). Again, reusing the code provides an efficient way of development of virtual and remote laboratories.

VI. STUDENT’S OPINION

Once the practice experience with virtual/remote laboratory is over, students must perform a survey used to get development/services satisfaction information. This survey is mandatory and it has 16 questions, classified in 4 main areas (questions/areas/ratings are shown in Table 1). The survey was performed by 78 students from 188 enrollment students in the subject. Due to the university’s regulations, students can deliver the practical homework in two dates (February 2012 and September 2012), so only the first date surveys are included in this paper.

<table>
<thead>
<tr>
<th>Question (scored 1-5)</th>
<th>Code</th>
<th>Average rating</th>
</tr>
</thead>
<tbody>
<tr>
<td>Overall Satisfaction (A1)</td>
<td></td>
<td>4.3</td>
</tr>
<tr>
<td>It was interesting to develop the virtual/remote labs?</td>
<td>Q1</td>
<td>4.5</td>
</tr>
<tr>
<td>Would you use this system in other subjects in the degree?</td>
<td>Q2</td>
<td>4.1</td>
</tr>
<tr>
<td>Would you recommend the use of RELATED in a professional environment?</td>
<td>Q3</td>
<td>3.5</td>
</tr>
<tr>
<td>What's your overall satisfaction rate about the practical experience developed?</td>
<td>Q4</td>
<td>4.8</td>
</tr>
<tr>
<td>Development/Services facilities (A2)</td>
<td></td>
<td>4.0</td>
</tr>
<tr>
<td>Consider the process of developing a virtual lab with RELATED, do you think it has been easy?</td>
<td>Q5</td>
<td>3.6</td>
</tr>
<tr>
<td>Consider the publish process of laboratories, do you think is a simple procedure?</td>
<td>Q6</td>
<td>5.0</td>
</tr>
<tr>
<td>Consider the development facilities provided by RELATED, do you think the learning process of these facilities is low?</td>
<td>Q7</td>
<td>3.9</td>
</tr>
<tr>
<td>Consider the documentation about development/services facilities, do you think is enough to make the practical experience?</td>
<td>Q8</td>
<td>2.9</td>
</tr>
<tr>
<td>Consider the services set provided by RELATED (data logging, booking, finding and accessing to laboratories), do you think that are appropriated to the virtual/remote labs development?</td>
<td>Q9</td>
<td>3.8</td>
</tr>
<tr>
<td>The client runtime is intuitive and easy to learn</td>
<td>Q10</td>
<td>4.3</td>
</tr>
<tr>
<td>Consider the used elements on views providing a visual feedback, do you feel that you are using a real on-site laboratory?</td>
<td>Q11</td>
<td>4.6</td>
</tr>
</tbody>
</table>

Reliability/Quality of software (A3) 3.5

As it is shown in Table 1, the overall satisfaction and curricula adequacy ratings is over 4 points, so students describes RELATED as a useful tool for the learning process. Also, from ratings data can be concluded that reliability/quality of software must be improved, specially the development documentation for a RELATED laboratory.
VII. CONCLUSIONS

RELATED provides a standardized way of developing remote/virtual laboratories and useful services and facilities (users management, booking system, etc.). The clean separation between data and presentation (modules and views) has several advantages, but the main is the software reusing (code develop for this lab or others). In the case of the MAGLEV system, on-site practical experience software is reused and also, previous developed modules/views (for example, the video server and image viewer) are included with no extra development cost.

With these features, students can develop their own virtual and remote laboratories, using the structured methodology of RELATED. The “module” paradigm allows to students develop their own code, and integrate with “real” experiments simply creating/modifying the XML specifications which defines the behavior of systems/experiments. Survey’s results from students indicate that the approach is easily adopted with minimum effort and it will be useful on practical experiences with other subjects.

ACKNOWLEDGMENT

The authors wish to thank the Spanish Ministry of Science and Innovation and the National Plan R&D TIN2008-06083-C01/TSI and TIN2008-06083-C03/TSI “s-Labs: Integration of open services for remote and distributed virtual laboratories, reusable and safe”. Also authors would like to acknowledge the support of the Project 142788-2008-BG-LEONARDO-LMP mPSS – mobile Performance Support for Vocational Education and Training Project and IPLECS Project – Internet-based Performance-centered Learning Environment for Curricula Support Project ERASMUS 141944-LLP-2008-1-ES-ERASMUS-ECDSP. Finally, the authors want to acknowledge the support provided by e-Madrid Project, S2009/TIC-1650, “Investigación y Desarrollo de tecnologías para el e-learning en la Comunidad de Madrid”.

REFERENCES


AUTHOR INFORMATION

Rafael Pastor, Senior lecturer at the National Spanish Distance University (UNED). IEEE Member, rpastor@scc.uned.es

Daniel Sánchez, Researcher at the National Spanish Distance University (UNED), dsanchez@scc.uned.es

Nourdine Aliane, Senior lecturer at the European University of Madrid (UEM), nourdine.aliane@uem.es

Roberto Hernández, Senior lecturer at the National Spanish Distance University (UNED). IEEE Senior Member. roberto@scc.uned.es

Antonio Robles-Gómez, Assistant Lecturer at the National Spanish Distance University (UNED). arobles@scc.uned.es

Agustín Caminero, Assistant Lecturer at the National Spanish Distance University (UNED), accamineros@scc.uned.es.

Salvador Ros, Senior lecturer at the National Spanish Distance University (UNED). IEEE Senior Member. sros@scc.uned.es

Gabriel Díaz, Associate lecturer at the National Spanish Distance University (UNED). IEEE Member, gdiaz@ieee.uned.es.

Manuel Castro, Professor at the Spanish University for Distance Education, IEEE Senior Member. mcastro@ieee.uned.es.