Integrating Mixed-Reality Remote Experiments Into Virtual Learning Environments Using Interchangeable Components

Frederico Menine Schaf and Carlos Eduardo Pereira

***Abstract*—This paper presents a proposal to integrate mixed-reality remote experiments into virtual learning environments (VLEs) using the concept of interchangeable components, which can represent either real or virtual devices or software in industrial automation systems. Combinations of real and virtual technical plants and automation systems are used in different learning scenarios for teaching control and automation concepts. Configurations of interchangeable components can be dynamically created via a VLE by configuring database parameters. The proposed system includes a remote Web interface that follows a thin client strategy and is designed to be compatible with Web browsers, including basic Java support. As the architecture that supports the integration of virtual and real components is located on the server side, remote students/users are only concerned with the experiment and do not need to be aware of the system that provides this integration and flexibility. In the current version, interchangeable components are integrated via an Object Linking and Embedding for Process Control interface, which is a widely adopted standard in the control and automation area. The proposed approach also provides practical and theoretical support for experiments within a collaborative virtual environment. This paper includes a description of four experiments developed using the proposed environment and concepts.**

***Index Terms*—Collaborative work, computer-aided engineering, educational technology, learning systems.**

I. Introduction

ENGINEERING education has become a crucial aspect for most countries since it has been recognized that skilled engineers are one of the main components for the development of innovative products and services, as well as for the optimization of production processes to ensure high productivity and quality. Considering education on control and automation systems, a key issue is the reduction of the gap between classical theoretical courses and real industrial practice. Hence, it is important to allow students to operate with devices, systems, and techniques that are as close as possible to those that they will confront in industrial settings. Unfortunately, reproducing a real industrial plant in an academic environment is not an easy task. Industrial equipment is, in general, very expensive (both in terms of acquisition and maintenance costs). Furthermore, safety constraints should also be taken into account. Such factors restrict the use of real industrial devices in academic laboratories, which, in general, are then structured as small-scale experiments with little connection to industrial reality. Within this context, an industrial laboratory facility that is available via the Web and, therefore, accessible at flexible times to a larger number of individuals helps to improve the overall cost effectiveness of such solution.

The Internet growth has brought new paradigms and possibilities in technological education. In particular, it allows the remote use of experimental facilities that can be used to illustrate concepts that are handled in a classroom and serves as an enabling and powerful technology for distance teaching. Through its worldwide connectivity, the Internet also allows having learning materials available to a much larger audience of students. Web-accessible laboratories with remote experiments have become an attractive economical solution for the increasing number of students [1]. They represent a "second-best-of-being-there" (SBBT) [2] solution for students and laboratories with expensive equipment. Following this trend, many institutions around the world have been engaged in the development of Web-based experimental settings. Systems aiming at teaching and research in several different areas have been proposed, such as digital process control [3], proportional–integral–derivative (PID) control [4], embedded communication systems [5], supervisory control [6], robot and other systems teleoperation [7]–[9], and real-time video and voice applications. Mostly, these experiments utilize customized devices and software to make small-scale textbook-like experiments remotely available.

However, our experience has shown that the availability of remote experiments is not a sufficient condition to ensure success in the learning process of control and automation engineers. Remote laboratory experiments offered as "stand-alone" settings, without connection to adequate learning materials (explaining the topics that are to be learned in the experiment), usually lead students to the use of a "trial-and-error" strategy, which has a lower learning impact than originally expected. Moreover, remote laboratories are available 24/7 for a large audience of students, increasing the demand in the number of faculty members and tutors that are necessary to provide online guidance to students. To alleviate these problems, remote experiments can be integrated into virtual learning

Manuscript received August 15, 2007; revised June 18, 2009. First published June 30, 2009; current version published November 6, 2009. This work was supported in part by the Brazilian research agencies Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES), Financiadora de Estudos e Projetos (FINEP), and Conselho Nacional de Pesquisa (CNPq), by the German exchange agency Deutscher Akademischer Austauschdienst (DAAD), and by the Serviço Nacional de Aprendizagem Industrial (SENAI) in Brazil.

The authors are with the Federal University of Rio Grande do Sul (UFRGS), Porto Alegre 90040-060, Brazil (e-mail: frederico.schaf@daad.alumni.de; cpereira@ece.ufrgs.br).

Digital Object Identifier 10.1109/TIE.2009.2026369

environments (VLEs) [10]–[12] that manage and provide guidance via learning materials before, during, and after the experimentation. This paper proposes such an integrated learning environment on which mixed-reality laboratory experiments and student guidance tools are combined for control and automation education. Mixed-reality experiments [13], on which simulated components can be combined to real equipment, are used to illustrate different learning situations according to the knowledge level of remote students.

The proposed environment has been developed with the context of the RExNet Consortium [14]. This Alfa II financed project, funded by the European Community, had mainly three goals: 1) to share; 2) to harmonize; and 3) to spread current skills on remote experimentation. The first goal directly addresses the essence of the ALFA program (financial support), namely, it calls for the cooperation among the consortium partners. Harmonization is a direct consequence of having universities from distinct countries with different languages and cultures. Each university participating in the RExNet project acts as a disseminating party within its own country, i.e., spreading access to remote experiments to other surrounding universities. Results of the RExNet project can be seen in [10], [12], and [15]–[20].

This paper is divided as follows. Section II describes Distance Education Environment of the Automation, Robotics and Control Systems Group (GCAR-EAD), a VLE for control and automation engineering education developed at our university. Sections III and IV describe experiment scenarios, which exploit different combinations of real and simulated automation-system modules. Section V compares the GCAR-EAD with other similar systems presented in the literature. Section VI discusses the results achieved when using the GCAR-EAD in undergraduate and graduate courses. Finally, in Section VII, conclusions are drawn and future work is discussed.

II. GCAR-EAD

Our experience in applying remote experiments for control and automation started in 2002 with the construction of a remote laboratory with a Foundation Fieldbus (FF) pilot plant [20]. That system enabled students to learn PID tuning control techniques and industrial communication protocols by working with a hypertext-based learning material and by remotely accessing the pilot plant.

A. FF Pilot Plant

The FF pilot plant consists of three interconnected tanks, four ac pumps (two of which are driven by vector-control drivers), and five control valves to control the liquid flow in the system. It is, therefore, possible to configure many different single-variable and multivariable control loops, involving tank level and liquid flow as process variables, with various choices of control inputs. All sensors and actuators in the plant are networked smart devices, which communicate using the FF protocol.

The use of a highly innovative industrial-oriented [21] technology brings a very strong perspective to research and teaching. Practical interest is raised in the students, as the experiments become visually and conceptually closer to the professional practice. Also, a wider variety of concepts can be illustrated, and several levels of difficulty and complexity/challenge can be explored.

The pilot-plant architecture [20] is based on Object Linking and Embedding for Process Control (OPC) and Web services technologies to provide remote accessibility. Tools like the Elipse SCADA[1] software (supervisory system) and the Syscon software and PCI302 FF network card[2] (SMAR commercial products) were employed in the development and operation of the industrial pilot plant. The FF devices are responsible for the control that is not affected by Internet connection delays.

The communication between the supervisory and the Web server is based on a client/server structure, which allows any browser (client) to access the screens of the supervisory system (server). The plant monitoring (visualization) is implemented by means of Java Applets. The user interacts with the browser, sending parameters to the plant via PHP–MySQL functions. The information is processed by the supervisory system in the server, and an answer is sent back to the user, closing the cycle.

The experiment accessibility is controlled through an access control tool that has the following objectives: user validation, experiment scheduling, monitoring a session's duration, parameterization of experiments, and experiment initialization.

B. GCAR-EAD: Embedding Remote Laboratories Into VLEs

Experiences using the FF pilot plant showed that due to the fact that the learning material was "loosely coupled" with the remote experiment, students were not able to identify which topics to review in case they could not get the proposed experiments adequately done.

To overcome those drawbacks, a system called GCAR-EAD was proposed, which supports mixed-reality remote experimentation. The GCAR-EAD has a more complex architecture (see Fig. 1) that additionally integrates learning-material manager (also called VLEs), educational materials, remote experiments with mixed reality [13], [22], [23], interchangeable components strategy [10], experiment analysis, and simple student guidance tools. The proposed architecture has five main modules: a learning-material manager, a student guidance system (or guide), experiment booking, experiment analysis (or analyzer), and experiment manager/interface. Each of these modules is responsible for controlling a specific functionality of the GCAR-EAD. The interaction with each module is transparent, i.e., students interact directly only with the VLE interface.

A central database is the main communication channel among modules (see Fig. 1). In the sequence, each of the modules is further described.

1) Learning-Material Manager: This module contains all learning materials. All users are identified via username/password, and, depending on the users' category (in case of students, their knowledge level), distinct operations are allowed when accessing an available learning material. Distinct learning modes are supported: active learning [24],

[1] Elipse SCADA. See http://www.elipse.com.br.

[2] SMAR PCI302; FF Process Control Interface. See https://www.smar.com/PDFs/Catalogues/PCI302CE.PDF.

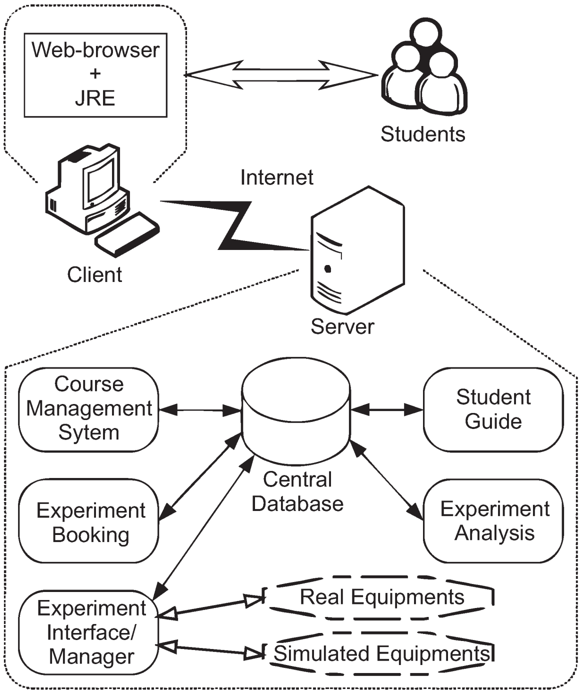


Fig. 1. GCAR-EAD high-level architecture.

[25], distributed learning [26], and team learning [27]. Active-learning skills are justified since, via environment interactions, students can "self-learn" (or self-teach). Distributed-learning skill is obviously linked to the spatial flexibility characteristic offered by VLEs' Web accessibility. The most important skill is, however, related to collaborative interaction, i.e., student teams (or users, in general) may work together, increasing the knowledge transfer in a common environment. In traditional courses, it is assumed that teachers were the only source of knowledge, and they centralized all courses' information. The collaborative learning skill is mostly associated with the social constructionist pedagogic line [28].

The learning-material manager plays the most meaningful role in the GCAR-EAD architecture since this system was developed to supply an educational (learning) environment attached to remote experiments for the students.

2) Experiment Booking Module: This module is responsible for controlling students' access to experiments. Since real experiments are not replicable with no cost, booking systems are necessary to organize the use of the real equipment (or entire real experiments) by students. Only signed VLE users can book/access experiments and select the available time slots for running their experiments in a calendar-type Web page.

3) Experiment Interface/Manager Module: The experiment manager provides a link among the remote experiments and the VLE and must ensure that the right remote experiment interface is available according to VLE setup parameters. That means that the experiment manager receives from the VLE a reference to the experiment to be executed and "constructs" the experiment, providing also a Java Applet interface for data visualization.

This module is also responsible for implementing the interchangeable components strategy [10] by linking and combining real and virtual components in a learning scenario.

4) Experiment Analysis: The experiment analysis module comprises tools to evaluate the results of a conducted experiment and determine some metrics derived from the experiment results (for instance, in the PID tuning experiment, usual metrics are overshoot, rise time, settling time, etc.). The experiment data are supplied by the experiment manager and the VLE in a form of reports or are directly stored in the central database.

Experiment feedback characteristics are also stored in the central database and are available for further visualization and/or manipulation by the VLE and other modules.

5) Student Guidance Module: The last module of the GCAR-EAD architecture is responsible for providing student guidance, which means that it receives as input the metrics generated by the experiment analysis module and has to determine whether students have met the goals defined by tutors/teachers. If not, this module has to indicate learning materials to be reviewed by the students. Therefore, this module acts as a simple tutoring system integrated in the VLE. The guidance must take into account all information stored in the database related to the student (student level, previous experiments performed, and visited learning materials). The tutor can also analyze reports (history data) of completed quizzes and VLE-proposed activities like assignments.

C. VLE Integration With Mixed Reality Supporting Interchangeable Components

While the remote access of real laboratory equipment has several advantages, there are also some issues to be considered for teaching control and automation concepts.

1) The number of students/student groups working simultaneously is equal to the number of physical experiments available.
2) Real systems with a slow dynamic lead to long waiting times.
3) Interlocking systems have to be carefully developed to avoid that students may damage components via improper actuation.

Such drawbacks can be overcome by employing simulated components. Simulations, although sometimes unrealistic, have some advantages that can be explored in different learning scenarios. One of the main advantages of using executable simulation models is that they can be easily replicated. Students can then simultaneously use multiple copies (replicas) of the same simulation simultaneously. The simulation replicas, instead of real experiments, do not imply on more equipment because replicas only consume more computer memory and processor power. Another advantage of using simulation is that students can speed up slow dynamic systems for quick visualization using simulation models (e.g., heating a real tank can take hours, while simulation speeds up to seconds). Safety concerns involving simulation-variable limits are not as important as in real experiments since the models cannot be damaged.

By analyzing the pros and cons of real versus simulated experiments, one can see that, in some sense, they are complementary so that a combination of both possibilities seems interesting. The strategy called *interchangeable components* has been developed to allow the combination of both real and virtual components [10]. This supports the definition of a variety of learning scenarios. For instance, real automation equipment such as programmable logic controllers could be remotely programmed by students, and these controllers would then control a simulated industrial plant. Additionally, simulated plants can be used to evaluate the robustness of control algorithms when the technical plant presents an unexpected behavior. On the other hand, simulated automation systems can be useful in showing step-by-step execution of industrial controllers.

Based on our experience in teaching control and automation courses, three different learning scenarios can be identified.

1) *Fully simulated*. This kind of experiment setup illustrates an experiment abstraction where simple and ideal simulation models (without perturbations and other real-world characteristics) are employed. Although simulation models are not necessarily simple or without perturbations, for didactic issues, the implementation of simple models is more adequate in early stages of experiment learning process.
2) *Mixed simulated/real components scenarios*. This configuration can be used, for instance, in the interaction between a simulated controller and a real plant to elucidate how acquired data from the real plant vary from the ideal model, and this can cause instability in the controller programmed logic; consequently, some precautions must be addressed in the simulated equipment to treat that instability. When dealing with a real controller and a simulated plant (hardware-in-the-loop), some problems also occur in the delay of the control logic since the controller cannot instantaneously process the acquired data (commonly, the controller cycle time is responsible for this delay).
3) *Scenario with real components*. This experiment scenario is the typical implementation of remote laboratories where SBBT [2] is implemented, and students can perform experimentation using real components and observe how theory applies into practical applications. Here, nonlinear behavior, perturbations, physical constraints, communication delays, etc., affect the experiment, and all these “real-life limitations” can be visualized.

D. VLE Integration With Tutoring Systems

The proposed VLE integration with tutoring systems is responsible for every GCAR-EAD interaction feedback. Tutoring systems are dependent on several other tools or modules. Each architecture module stores information in the central database contributing to the tutoring-system feedback compilation.

Basically, an integrated tutoring system gives two kinds of feedback: 1) allows remote experiment configuration according to the user (student) level, i.e., students with no previously recorded interaction with the experiment should start with basic experiments, whereas more advanced students can directly go to more complex experiments; and 2) infers, based on the assessment of the experiment performed by a student, the required learning material to be worked.

The first type of feedback compilation searches in the central database for previously performed experiments and visited learning materials for a given student or student group. Based on these data, it “decides” which type of learning scenario can be accessed. The second feedback type also takes into account the reports generated by the experiment analyzer to suggest a specific didactic material to the student.

The experiment analyzer plays a meaningful role in the experiment-driven tutoring-system feedback. There are two proposed types of experiment analyses: 1) For dynamic experiments, the result of the analyses (“evaluation”) is mostly computed offline, i.e., after the experiment has been concluded, control metrics like overshoot and rise time are calculated. 2) On the other hand, discrete experiment based on logic control can be evaluated in execution time since the digital I/Os can be tested while the experiment is running. The first type is called postruntime analysis, whereas the other is called runtime analysis, but both produce reports that are stored in the central database.

The postruntime-analysis tool compares the results of the experiments performed by students with the requirements specified in the experiment assignment. The runtime analysis includes some testing codes in the experiments to verify if specific conditions are satisfied. Although the evaluation is made in the execution time, the suggestion is also postexperiment. Stored experiment messages (experiment reports) are read by the tutoring system, and learning-material suggestion is done in case the experiment’s goals were not reached.

III. GCAR-EAD Implementation

This section will describe some applications developed using the GCAR-EAD architecture presented in this paper. Most of the applications were developed within the scope of the RExNet Project [14].

The learning-material manager module offers a user-friendly interface with the free open-source course-management system called Moodle [28]. Moodle is a widely used software package designed with social constructionist pedagogical principles to help educators create effective online learning communities with many collaborative learning tools.

OPC-Data Access (OPC-DA) standardized interface is employed to allow configuration of scenarios without building four different and distinct experiments (which include three learning scenarios). The OPC interface provides a common, simple, and reusable interface to the interchangeable components strategy implementation. Basically, the same interface is used for students’ interaction with real and simulated components (and, therefore, they are considered “interchangeable”). Each of the components (real/simulated plant and controllers) must have an OPC interface, so that automatic selection of the interchangeable components can occur; consequently, configuration of scenarios can be automatized. Therefore, OPC works as a “glue” among different applications (in this case, components).

The experiment booking, student guidance, and experiment analysis modules are implemented in PHP code. A SCADA software, named Elipse SCADA, is used as the experiment manager and is responsible for providing connections with the real/simulated equipment and for supplying Java Applets as experiment visualization interfaces.

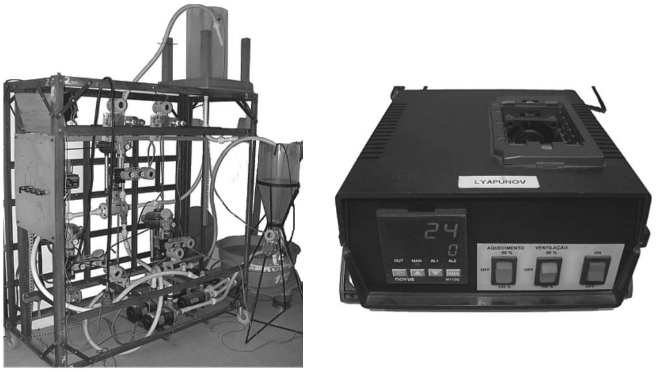
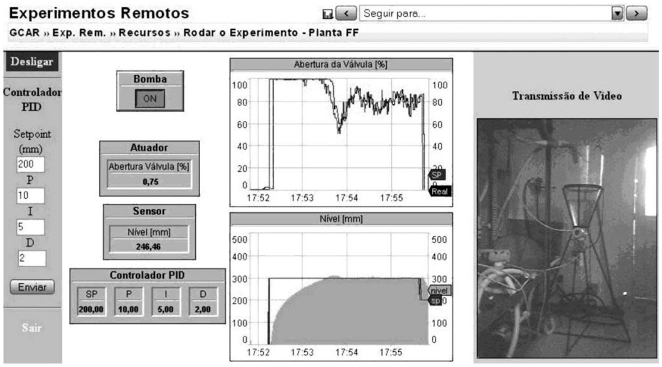


Fig. 2. (Left) FF pilot plant experiment. (Right) Thermal plant experiment.


Experimentos Remotos

Seguir para...

GCAR » Exp. Rem. » Recursos » Rodar o Experimento - Planta FF

Desligar

Controlador PID

Setpoint (mm): 200

P: 10

I: 5

D: 2

Enviar

Sair

Bomba: ON

Atuador — Abertura Válvula [%]: 0,75

Sensor — Nível [mm]: 246,46

Controlador PID

SP	P	I	D
200,00	10,00	5,00	2,00

Fig. 3. Snapshot of the computer screen viewed by remote students when performing the experiment on the FF pilot plant in GCAR-EAD environment. Note that the water level could also be monitored by an online video feedback.

All developed remote experiments (case-studies implementations) follow common software architecture: Apache Web-server software, MySQL as a database interface manager, OPC-DA for experiment-level communication interface, and the ISaGRAF as a simulation tool.

IV. Developed Remote Experiments

Four major courses for control and automation education were developed in the GCAR-EAD environment: 1) PID controllers; 2) IEC 61131 Standard for programming industrial automation applications; 3) FF communication protocol; and 4) remote experiment course. Other related courses are in development, e.g., a complete course on control systems theory.

The PID course contains several learning materials related to the design of PID controllers, such as PID tuning methods, stability analysis, and system performance analysis. This course makes extensive use of remote experiments, aiming to provide students with some hands-on examples.

The second course introduces the IEC 61131 Standard, a widely adopted standard for programming industrial applications. Students are required to program industrial applications using the IEC 61131 languages (sequential flowcharts, ladder diagrams, instruction list, function block diagrams, and structured text).

The FF communication-protocol course introduces concepts of industrial communication protocols in general and includes a specific discussion on the FF protocol.

The last course contains didactic materials referring to the construction and use of the remote experiments developed and integrated in the GCAR-EAD.

A. Enhanced FF Pilot Plant

The first experiment was developed using an FF pilot plant [10], [20], which has been used as a remote laboratory for several years for teaching PID controller theory, in particular, PID parameter tuning (see Fig. 2). The original remote laboratory was integrated into the GCAR-EAD (see Fig. 3) by including a new learning material on PID control theory as well as incorporating an experiment analysis tool to calculate control metrics from the results of the performed experiments. In this experiment, students have to control the water level in two tanks by acting on pumps and valves.

The interchangeable components strategy is implemented in using the script language of the SCADA software (Elipse). Moodle-embedded PHP programs write scenario parameters so that the scripts configure the experiment, setting up links and connections between selected components.

B. Thermal Plant

The second experiment uses a simple thermal plant [5] built with a PID industrial controller and simple electronic equipment to illustrate temperature-control techniques and the use of industrial controllers (see Fig. 2). The experiment consists of a thermal resistance and a PTC100 thermal sensor connected to an industrial controller N1100 from the Brazilian company Novus. This controller manipulates the electric current that flows in the resistance via pulsewidth-modulation switching and, thus, affecting the temperature of the resistance. This industrial controller offers a PID control technique. Students must, based on open-loop trails, identify and control the temperature (in closed loop) of the experiment by setting PID parameters given temperature set points.

C. Mixed-Reality Electropneumatics Workbench

The third experiment, used in vocational training, was developed in collaboration with researchers from the University of Bremen and the University of Berlin in Germany [13], members of the RExNet consortium. In this case, the so-called deriveSERVER, developed in Germany, is integrated via an OPC interface to the GCAR-EAD environment [12].

The system provides mixed-reality experiments through the use of "hyperbonds," which allow tight coupling between physical and virtual phenomena [13]. Physical signals such as air pressure and electric potential are converted into binary logic information and vice versa. Hyperbonds are based on the bond-graph theory [29], which provides a unified view on different systems using the notion of effort and flow.

The system software architecture is entirely based on client–server modules allowing great modularity and mobility (system distribution). The hyperbond software interface was modified from the original German project also currently supporting, besides original hardware connectivity, OPC and parallel/serial communication interfaces.

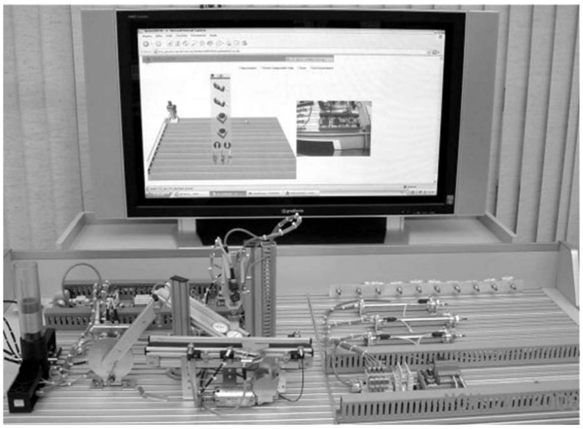


Fig. 4. Mixed-reality workbench for mechatronics vocational training at SENAI-RS.

Owing to the client–server architecture of the hyperbond to the experiment manager, i.e., Real Object Manager, students can use local hardware attached to a simple parallel PC port to interact with the experiment and also to local OPC servers with any kind of compatible equipment.

The new hyperbond interfaces offer remote hardware or simulator interaction using any of the interfaces, i.e., students can have their own hardware/software (which are compliant with the new interfaces) remotely interacting with the system.

Didactic materials related to this experiment, in Portuguese and English versions, are divided into four major courses: 1) devices manual; 2) use of the deriveSERVER system; 3) experiment guide; and 4) basic electropneumatics.

To focus the system utilization, task-oriented experiment scenarios were developed so that students have a well-specified goal (see Fig. 4).

D. Simulated Bottle Production Plant

The fourth experiment is a virtual experiment, whose behavior is modeled in the ISaGRAF software using IEC 61131 programming languages. This experiment provides a very didactic and reusable experiment that can be combined with the other experiments presented and has a built-in analysis tool, integrated into the simulation model, which can check the behavior of automation systems developed by the students that control the process.

V. Related Works

Although there are several interesting related works published in the literature, we could not identify any solution integrating all concepts incorporated into GCAR-EAD.

Similar FF pilot-plant experiments were encountered, such as Coupled Tank Apparatus WebLab [30] and Automatic Control Telelab (ACT) [31].

Several projects have employed and tested remote-experiment networks like LabNet [32], CyberLab [33], Remote Wiring and Measurement Laboratory [34], digital-signal-processing-based Remote Control Laboratory [35], Digital Electronics Education and Design Suite (DEEDS) [36], and Network Control System Laboratory [37].

The Solar Energy e-Learning Lab from Michaelides *et al.* [11] has an integrated learning system with several learning materials and "quizzes" to identify student-understanding level. First, the student must pass several theory tests so that the system grants remote-experimentation access. Despite these qualities, the system also does not offer experiment feedback.

The Remote Lab System [38] is a much more elaborate e-learning environment, where a framework that addresses bandwidth efficiency for the remote monitoring and control of course laboratories is employed. This framework provides instructor-friendly remote monitoring of the laboratory, effective evaluation, and grading methodology, and, different from others, also supports effective help from the teacher. The teachers' help connection allows audio transfer and remote controlling of the student's computer, which helps in case of doubt resolution.

The Distance Internet-based Embedded System Experimental Laboratory (DIESEL) [39] is a learning-environment system for remote experimentation that allows students to access embedded hardware ranging from microcontrollers to real-time operating systems. This implementation also offers autonomous configuration of the experiment with intelligent user support, i.e., provides a semiautomated intelligent helping tool based on a case-based reasoning system that is able to monitor and infer context-aware assistance to users on demand. Therefore, this system highlights the research pointed to autonomous help or user support that a remote-learning environment can offer.

The Power Electronics E-Learning Platform proposed in [40] consists of a reconfigurable power-electronics testbed, a Web-based distance laboratory, and a user interactive e-learning platform. The reconfigurable power-electronics testbed can be configured via a Web-based interface to construct a wide variety of converters and inverters. This way, this system offers much more flexible experiments.

Another excellent example of flexible experiment configuration is the deriveSERVER system proposed in [13]. Unlike others, this employs mixed-reality techniques and is integrated in the VLE with collaborative and distributed learning methods. This solution, however, has no specific experiment goals or experiment learning feedback. Thus, the user (student) neither is guided nor receives any analyzed results of the performed experiment.

The ACT proposed in [31] offers not only controller parameterizations but also MATLAB Simulink models to describe and characterize the controller logic. This interesting approach is very useful in the experiment configuration. Thus, students can design their own experiment controller in a much more flexible way.

VI. Experiences Using GCAR-EAD

The GCAR-EAD is currently being used in undergraduate electrical engineering lectures at the Federal University of Rio Grande do Sul (UFRGS). So far, experience in adopting the system can be considered as very successful, either in terms of comparing the performance of students using the tool with others that are not using it or based on students' evaluation. A custom quiz was developed to evaluate system qualities and faults according to this class of students. Results indicate that the majority of the students have enjoyed the environment and, in particular, the possibility of combining both real and virtual experiments. The quiz also indicates that the time flexibility and the collaborative environment are the most important characteristics according to the students' opinion. Thus, obtained results have been very positive. In particular, one can see that students' motivation has increased. Logging data analysis shows that whereas some students access the remote experiments late at night, others prefer to work early in the morning; that means each one can define his or her preferable working/learning time. The system is being continuously "tested" and improved with many suggestions.

As an example, the use of the GCAR-EAD system in an undergraduate lecture on Control Systems in the second semester of 2007, which has proposed a blended learning method, combining remote virtual lessons with traditional classes, has considerably increased in comparison with those in the previous years (around 90% of students approved, 23% more than in the previous semester with the same instructor).

VII. Conclusion and Future Work

This paper has presented a virtual Web-accessible learning architecture. The GCAR-EAD environment was developed based on the proposed architecture, which allows an integration of mixed-reality experiments with VLEs, introduces the concept of interchangeable components, includes experiment analysis tools, and provides student guidance through the learning material.

Whereas the proposed environment has proven to be very useful for control and automation education, there are still some challenges to be faced. The synchronization in the timing behavior of the virtual and real equipment is dependent on the communication delays in the network infrastructure. In the current implementation, this delay is around 2 s for the whole communication between the client and the end actuators. While this is not meaningful for technical plants with slow dynamics (as is the case in the selected experiments), it has to be improved. Of course, there is a tradeoff in having geographically distributed applications and the higher communication times that are required.

The proposed environment can also be used for collaborative engineering since experiments can be distributed into several sites, and several students (users) can interact using the same environment.

Although the work of Ma and Nickerson [41] presents a literature review with skeptic research on the long-term debate of technology as the enabler of student's learning, it is widely believed that collaborative experiences enabled by remote experiments associated with virtual environments are powerful drivers of cognitive processes and can significantly enhance learning efficiency. The benefits of collaborative learning are widely researched and advocated throughout the literature [42]. Regardless of the varying theoretical emphasis in different approaches on collaborative learning (e.g., social constructivism), research clearly indicates that, in many (not all) cases, students learn more effectively through collaborative interaction with others. This motivates the preparation of remote laboratories for collaborative learning and their use in distributed teaching scenarios with simulation tools, hands-on laboratories, and practical workshops.

There is a strong demand for research that seeks to create such "blended learning," where collaborative remote laboratories can play a significant role. Emphasis on collaboration adds new technical requirements to the design of remote laboratories.

Mixed-reality concepts support learning environments with remote laboratories and distributed workspaces. The bidirectional telecooperation functionality allows students to use the Internet for collaborative engineering. The presented environment allows groups of students/technicians (or even employees) at remote locations to take part in the same training using the same equipment (either simulated or real). The users are able to work in a collaborative way to solve problems and explore solutions to proposed problems. This kind of interaction provides a systematic support for skilled workers and engineers. The presented research can be used as an appropriate and cost-effective tool to support collaborative engineering.

The hyperbond concept allows integration of different remote equipment or simulators providing common environments to remote collaboration of experiments. This is especially important to cost-effective remote experimentation since real hardware must not be duplicated.

While software may be designed to achieve closer social ties or specific deliverables, it is hard to support collaboration without also enabling relationships to form and to support a social interaction without some kind of shared coauthored works. For this purpose, a new environment interface is under development. This new interface will employ play ethic (methods) applied to work-turning activities that employ computers for a more comfortable experience. This is commonly referred to as gamelike interface. A 3-D representation is also in study aiming to display more realistic (virtual) *worlds* to its users. In this manner, future works indicate new trends to the GCAR-EAD that will be further developed to become a complete computer-supported collaborative environment.

Acknowledgment

The authors would like to thank their RExNet partners for their fruitful collaboration.

References

[1] M. Cooper, "The challenge of practical work in a euniversity—Real, virtual and remote experiments," in *Proc. Inform. Soc. Technol. Conf.*, 2000, pp. 34–40.

[2] B. Atkan, C. Bohus, L. A. Crowl, and M. H. Shor, "Distance learning applied to control engineering laboratories," *IEEE Trans. Educ.*, vol. 39, no. 3, pp. 320–326, Aug. 1996.

[3] L. L. Bello, O. Mirabella, A. Raucea, and L. Capetta, "ENEL Pilot: From a research testbed to a virtual educational laboratory," *IEEE Trans. Ind. Electron.*, vol. 56, no. 12, pp. 4844–4853, Dec. 2009.

[4] C. Batur, Q. Ma, K. Larson, and N. Kettenbauer, "Remote tuning of a PID position controller via Internet," in *Proc. Amer. Control Conf.*, 2000, pp. 4403–4406.

[5] C. Schmid and A. Ali, "A web-based system for control engineering education," in *Proc. Amer. Control Conf.*, 2000, vol. 5, pp. 3463–3467.

[6] J.-S. Lee and P.-L. Hsu, "Remote supervisory control of the human-in-the-loop system by using Petri nets and Java," *IEEE Trans. Ind. Electron.*, vol. 50, no. 3, pp. 431–439, Jun. 2003.

[7] R. Marín, P. J. Sanz, P. Nebot, and R. A. Wirz, "Multimodal interface to control a robot arm via the web: A case study on remote

programming," *IEEE Trans. Ind. Electron.*, vol. 52, no. 6, pp. 1506–1520, Dec. 2005.

[8] L. Huijun and S. Aiguo, "Virtual-environment modeling and correction for force-reflecting teleoperation with time delay," *IEEE Trans. Ind. Electron.*, vol. 54, no. 2, pp. 1227–1233, Apr. 2007.

[9] A. Balestrino, A. Caiti, and E. Crisostomi, "From remote experiments to web-based learning objects: An advanced telelaboratory for robotics and control systems," *IEEE Trans. Ind. Electron.*, vol. 56, no. 12, pp. 4817–4825, Dec. 2009.

[10] F. M. Schaf and C. E. Pereira, "PID controller remote tuning experiment with learning environment integration," in *Proc. 12th IFAC Symp. Inform. Control Probl. Manuf.*, 2006, vol. 2, pp. 753–758.

[11] I. Michaelides, P. Eleftheriou, and D. Müller, "A remotely accessible solar energy laboratory—A distributed learning experience," in *Proc. Int. Conf. Remote Eng. Virtual Instrum.*, 2004.

[12] F. M. Schaf, C. E. Pereira, A. C. Assis, C. L. Reichert, F. Campana, and I. Krahkeche, "Collaborative learning environment using distributed mixed reality experiment for teaching mechatronics," in *Proc. 8th IFAC Symp. Cost Oriented Autom.*, 2007.

[13] F. W. Bruns and H.-H. Erbe, "Mixed reality with hyper-bonds: A means for remote labs," in *Proc. IFAC 11th Symp. Inform. Control Probl. Manuf.*, 2004, pp. 55–68.

[14] G. R. Alves, J. M. Ferreira, D. Müller, H.-H. Erbe, N. Hine, J. B. M. Alves, C. E. Pereira, L. Chiang, O. Herrera, and L. E. Sucar, "Remote experimentation network—Yielding an inter-university peer-to-peer e-service," in *Proc. 10th IEEE Int. Conf. Emerging Technol. Factory Autom.*, 2005, pp. 1023–1030.

[15] J. B. Silva, G. R. Alves, and J. B. Alves, "Remote experimentation integrating research, education and industrial application," in *Proc. 8th IFAC Symp. Cost Oriented Autom.*, 2007.

[16] Y. Hernández, J. Noguez, and L. E. Sucar, "Incorporating an affective model to an intelligent tutor for mobile robotics," in *Proc. 36th ASEE/IEEE Frontiers Educ. Conf.*, 2006, pp. T2F–22–T2F–27.

[17] H.-H. Erbe and D. Müller, "Distributed work environments for collaborative engineering," in *Proc. 7th Int. Conf. Inform. Technol. Balanced Autom. Syst. Manuf. Serv.*, 2006, pp. 4–6.

[18] O. Herrera, G. R. Alves, D. Fuller, and R. G. Aldunate, "Remote labs experiments: Opening possibilities for distance learning engineering fields," in *Proc. 19th IFIP World Comput. Congr.*, 2006, pp. 321–325.

[19] J. M. M. Ferreira, A. M. Cardoso, and M. G. D. Gericota, "An integrated framework to support remote IEEE 1149.1/1149.4 design for test experiments," *Int. J. Online Eng.*, vol. 2, 2006.

[20] R. P. Zeilmann, J. M. G. Silva, A. S. Bazanella, and C. E. Pereira, "Web-based control experiment on a foundation fieldbus pilot plant," in *Proc. IFAC Int. Conf. Fieldbus Syst. Appl.*, 2003, pp. 325–330.

[21] L. Costas-Pérez, D. Lago, J. Fariña, and J. J. Rodriguez-Andina, "Optimization of an industrial sensor and data acquisition laboratory through time sharing and remote access," *IEEE Trans. Ind. Electron.*, vol. 55, no. 6, pp. 2397–2404, Jun. 2008.

[22] P. Milgram, "Merging real and virtual worlds," in *Proc. IMAGINA*, 1995, pp. 218–230.

[23] M. Billinghurst and H. Kato, "Collaborative mixed reality," in *Proc. Int. Symp. Mixed Reality*, 1999, pp. 261–284.

[24] K. Watson, "Utilization of active and cooperative learning in EE courses: Three classes and the results," in *Proc. Conf. Frontiers Educ.*, 1995, vol. 2, pp. 1–6.

[25] K. A. Smith, "The craft of teaching cooperative learning: An active learning strategy," in *Proc. Conf. Frontiers Educ.*, 1989, pp. 188–193.

[26] M. Auer, A. Pester, D. Ursutiu, and C. Samoila, "Distributed virtual and remote labs in engineering," in *Proc. IEEE Int. Conf. Ind. Technol.*, 2003, vol. 2, pp. 1208–1213.

[27] N. Faltin, A. Böhne, J. Tuttas, and B. Wagner, "Distributed team-learning in an Internet-assisted laboratory," in *Proc. Int. Conf. Eng. Educ.*, Manchester, U.K., 2002, pp. 1–5.

[28] M. Dougiamas and P. Taylor, "Moodle: Using learning communities to create an open source course management system," in *Proc. World Conf. Educational Multimedia, Hypermedia Telecommun.*, 1995, pp. 171–178.

[29] H. M. Paynter, *Analysis and Design of Engineering Systems*. Cambridge, MA: MIT Press, 1961.

[30] C. C. Ko, B. M. Chen, J. Chen, Y. Zhuang, and K. C. Tan, "Development of a web-based control experiment for a coupled tank apparatus," *IEEE Trans. Educ.*, vol. 44, no. 1, pp. 76–86, Feb. 2001.

[31] M. Casini, D. Prattichizzo, and A. Vicino, "The automatic control telelab: A Web-based technology for distance learning," *IEEE Control Syst. Mag.*, vol. 24, no. 3, pp. 36–44, Jun. 2004.

[32] F. Davoli, P. Maryni, M. Perrando, and S. A. Zappatore, "General framework for networked multimedia applications enabling access to laboratory equipment: The LabNet project experience," in *Proc. IEEE Int. Conf. Inform. Technol.: Coding Comput.*, 2001, pp. 359–365.

[33] R. Haugom, O. K. Solbjørg, K. Y. Pettersen, and T. I. Eikaas, "A simulation game for nonlinear control theory education," *Model. Identif. Control*, vol. 28, no. 2, pp. 45–51, Apr. 2007.

[34] J. A. Asumandu, R. Tanner, J. Fitzmaurice, M. Kelly, H. Ogunleye, J. Belter, and C. H. Song, "A web-based electrical and electronics remote wiring and measurement laboratory (RwmLAB) instrument," *IEEE Trans. Instrum. Meas.*, vol. 54, no. 1, pp. 38–44, Feb. 2005.

[35] D. Hercog, B. Gergic, S. Uran, and K. Jezernik, "A DSP-based remote control laboratory," *IEEE Trans. Instrum. Meas.*, vol. 54, no. 6, pp. 3057–3068, Dec. 2007.

[36] G. Donzellini and D. Ponta, "A simulation environment for e-learning in digital design," *IEEE Trans. Ind. Electron.*, vol. 54, no. 6, pp. 3078–3085, Dec. 2007.

[37] C. Lazar and S. Carari, "A remote-control engineering laboratory," *IEEE Trans. Ind. Electron.*, vol. 55, no. 6, pp. 2368–2375, Jun. 2008.

[38] A. Mittal, C. Gupta, and A. Gupta, "Addressing the bandwidth efficiency, control, and evaluation issues in software remote laboratory," *IEEE Trans. Ind. Electron.*, vol. 55, no. 6, pp. 2326–2333, Jun. 2008.

[39] M. J. Callaghan, J. Harkin, T. M. McGinnity, and L. P. Maguire, "Intelligent user support in autonomous remote experimentation environments," *IEEE Trans. Ind. Electron.*, vol. 55, no. 6, pp. 2355–2367, Jun. 2008.

[40] S.-C. Wang and Y.-H. Liu, "Software-reconfigurable e-learning platform for power electronics courses," *IEEE Trans. Ind. Electron.*, vol. 55, no. 6, pp. 2416–2424, Jun. 2008.

[41] J. Ma and J. V. Nickerson, "Hands-on, simulated, and remote laboratories: A comparative literature review," *ACM Comput. Surv.*, vol. 38, no. 3, pp. 1–24, Sep. 2006.

[42] E. Lehtinen, "Computer supported collaborative learning: An approach to powerful learning environments," in *Unraveling Basic Components and Dimensions of Powerful Learning Environments*. Amsterdam, The Netherlands: Elsevier, 2003, pp. 35–53.

Frederico Menine Schaf was born in Alegrete, Brazil, in 1979. He received the B.S. degree in electrical engineering in 2002 from the Universidade Federal de Santa Maria, Santa Maria, Brazil, and the M.Sc. degree in electrical engineering in 2006, with emphasis on industrial automation and instrumentation, from the Federal University of Rio Grande do Sul (UFRGS), Porto Alegre, Brazil, where he is currently working toward the Ph.D. degree.

He is staying as a Guest Researcher with the Universität Bremen, Bremen, Germany. He developed several applications, tools, and didactic materials for Web-accessible experiments within the scope of the European Alfa II project called RExNet. His main research interests include mixed reality, remote handling, multiagents systems, collaborative learning and work environments, Web and learning technologies, and modeling and simulation that are always focused on the field of engineering education and training, where he has several publications.

Carlos Eduardo Pereira was born in Porto Alegre, Brazil, in 1965. He received the B.S. degree in electrical engineering and the M.Sc. degree in computer science from the Federal University of Rio Grande do Sul (UFRGS), Porto Alegre, in 1987 and 1990, respectively, and the Dr.-Ing. degree in electrical engineering from the University of Stuttgart, Stuttgart, Germany, in 1995.

He is currently an Associate Professor with the Electrical Engineering Department, UFRGS. His research focuses on methodologies and tool support for the development of distributed real-time embedded systems, with special emphasis on industrial automation applications and the use of distributed objects over industrial communication protocols.

Dr. Pereira is the Chair of the International Federation of Automatic Control (IFAC) Technical Committee on Manufacturing Plant Control (TC 5.1). He has acted as a member of International Program Committees for several conferences in the field of industrial automation, manufacturing, industrial protocols, and real-time distributed object computing. He is also an Associate Editor of the journal *Control Engineering Practice*. He is currently the Chair of the Brazilian Automation Society, the IFAC's national member organization in Brazil.