Electronic Laboratories: ICTT@Lab experimentation

Hcène Benmohamed, Arnaud Lelevé, Patrick Prévot

Electronic Laboratories: ICTT@Lab experimentation

Hcene BENMOHAMED, Arnaud LELEVE, Patrick PREVOT[1]
ICTT Laboratory - INSA de LYON
Bât. L. de Vinci, 21 avenue Jean Capelle,
69621 Villeurbanne Cedex, France

Phone: +33 (0)4.72.43.60.47/ Fax: +33 (0)4.72.43.79.92

ABSTRACT

During last decade, Internet and related web technologies development enabled the arising of e-learning services and made distant learning a reality. As traditional face to face classroom became virtual classroom through Internet, traditional laboratories found their equivalent in e-Laboratories. These e-Labs enable learners to train on remote real or virtual systems. They represent important components in e-learning environments, especially in scientific and technical disciplines.
Up to now, e-learning environments didn't integrate features allowing remote practical works with the same opportunities and facilities of edition, content use and reuse, as the other e-learning components. In previous papers, we proposed a software environment coming to extend traditional features of standard LMS in order to enable the execution of e-Lab sessions. This solution covers the whole edition chain, from scenario creation for a class of apparatus, to real-time manipulation by learners of a corresponding apparatus. In this paper we describe an experimentation of this e-Lab solution.

Keywords: remote laboratories, virtual laboratories, microcomputer in education, web-based training, distance learning, e-learning.

I. INTRODUCTION

For a few years, numerous educational institutions have incorporated the Information and Communication Sciences and Technologies into their educative systems. This choice brought new tools which contribute to a continuous improvement of educational practices. So, e-learning solutions, which were at first essentially based on abstract education (online courses, virtual classrooms, e-projects, role-playing, ...), gradually opened themselves to real practical activities such as Remote Laboratories [1]. This trend answers to a recognized need for such activities and enables real experimentation in scientific and technical disciplines.
This paper presents a quick survey on Electronic Laboratories and recalls the work done for 4 years: initial problematic, our approach, the proposition of a complete life cycle for E-Lab pedagogic content and the generic framework we developed. This framework is depicted here with a sample of apparatus, used for experimentation in automation discipline. The last and main part of this paper describes experimentations we conducted last summer.

II. ELECTRONIC LABORATORIES

Generally speaking, we distinguish between Electronic laboratories (**E-Labs**), either Remote Laboratories (**R-Labs**) [2] [3] or Virtual Laboratories (**V-Labs**) [4] [5]. R-Labs (sometimes called *"web-based control"*), offer remote access to real laboratory equipment and instruments. V-Labs are based on simulations of real systems or phenomena often supplied by web-based simulations using generic commercial software such as *Matlab+Simulink* ® [6] [7] or *LabView* ® [8].
A few years ago, first solutions focused on very specific academic needs (water level regulation in automatic control discipline, simulation of microprocessor functioning in electronics, …). As needs for different applications progressively grew up, second generation E-Labs supplied general solutions which could be applied into a given discipline (as example, [8] [9] for robotics or [10] for chemistry). Current third generation E-Labs propose even more generic architectures such as [3]. From another point of view, first approaches to this problem focused on technical solutions to make apparatus teleoperation easier [11]. When educational content began to be linked to technical solutions, we could regret that contents (scenarios) were merged with containers (platform) in such a way that no evolution was possible without reprogramming the whole. Hopefully, a few solutions, such as *Emersion* project [12], had a modular approach.
In the meantime, V-Labs became more mature. Many authoring tools and generic simulators now coexist such as MARS [13], SimQuest [14] and EJS [15]. Starting from this statement, we started four years ago to study a generic E-Lab architecture [16].

Simulation versus lab works
Simulation is used when the manipulated system is virtual, based on a model run by computer. Simulations return results whose closeness to reality depends on the model complexity. Moreover, the way results are presented has not the same impact according to their production (in the meaning of a theatre production). For instance, try to learn how to play billiards on a computer screen with a keyboard (such as old 2D billiards games, filmed from the top of the table) or dived in a 3D virtual world with Virtual Reality Equipment and haptic interface: with the same ball movement modelling behind the scene, better commandability (command ability) and observability make the training more effective.
From our point of view, simulation is linked to virtual laboratories and is complementary in this aspect with remote laboratories. As a whole, they are essential when the simulated system is not reachable by learners: microscopic or macroscopic phenomena, destructive process … In fact, we think that, when it is possible, it

[1] {first name}.{last name}@insa-lyon.fr

is recommended to merge virtual and remote laboratories. Indeed, professional plane pilots have been using simulators made of physical simulators (a real scale cockpit with same equipment as in planes) merged with a computerized simulation of the rest of the plane and the environment. One of the important impacts of featuring a simulation facility in a remote lab is that learners can train on the simulator before testing on the real system. Therefore they can, on one hand, closely compare reality versus theory and, on the other hand, share the real system with other learners in a finer schedule, which may allow having more populated remote lab sessions than local ones. However, this presupposes an efficiently defined time sharing protocol but results in an increase of productivity of the system (more learners per hour).

III. GENERIC SCENARIOS FOR E-LABS

Main problematic

Imagine an author writing an E-lab scenario whose pedagogical objectives are to learn how to compute P.I.D. parameters for a thermal system. This author uses a dedicated authoring tool for this task.

Many tutors are certainly interested to reuse this scenario without having to rewrite one from scratch. But each tutor owns different apparatuses and software to run E-Lab scenarios. Every necessary functionality is featured by these different apparatuses, for this specific use, but with different components and interfaces. Hence, tutors have to manually translate such scenarios into a specific format for their E-Lab platform and to adapt it to their equipment. Moreover, they have to do it each time, for each scenario and apparatus.

There was the same situation when every word processor had to incorporate every printer specification to be able to print on any customer printer from the market. This situation prevents any sharing or re-use of scenarios between authors and tutors whereas, nowadays, conceptual E-Learning contents are able to be spread and shared through Learning Content Management Systems (LCMS) and re-used on compatible Learning Management Systems (LMS).

Proposed concept

The aim of this concept is to provide, in one hand a way to edit generic E-Lab scenarios and, in the other hand, a standard generic E-Lab platform to host any apparatus to be used with any relevant scenario.

The key concept is a middleware called ElaMS (Electronic Laboratory Management System) which plays the role of driver manager according printer analogy. It permits E-Lab scenario sharing as for any standard pedagogical content. In this context, an author writes a standard generic scenario which is linked to a class (a *template*) of apparatuses (motor, oven, robot, optic system …) which, in its turn, is represented by a set of available standard functionalities. This scenario can be run on a standard (in our case, IMS-LD compliant) LMS and its activities appeal to specific functionalities of a fictive apparatus. When a tutor wants to execute it on its platform, he just has to run an automated *scenario adaptation script* which transforms this generic scenario into an *apparatus specific scenario* by changing its generic functionality calls into real function calls towards his apparatus.

This life cycle is depicted in figure 1. It consists of four main steps:

1. installation (by a platform manager) of a new apparatus on the ELaMS platform; this apparatus must belong to a known apparatus class.
2. Creation (by authors) of generic pedagogic scenarios on an authoring tool; each scenario is linked to an apparatus class.
3. Adaptation (by tutors) of a generic pedagogic scenario to a specific apparatus and
4. use (by tutors and learners) of these scenarios on this specific apparatus.

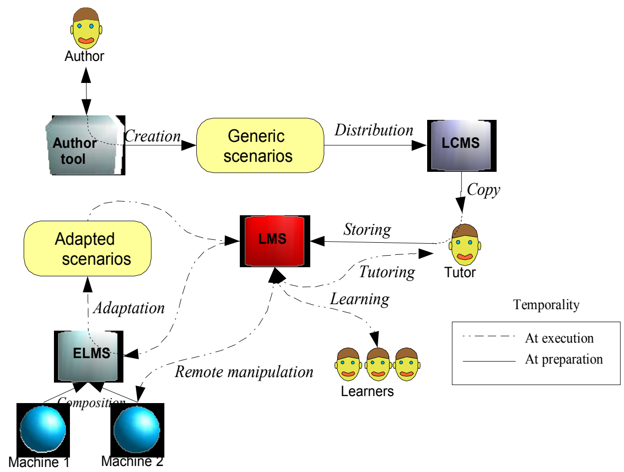

Fig. 1*: Typical proposed life cycle for e-lab scenarios*

Benefits

This solution enables to share a same scenario between numerous similar apparatuses. There is no need to have specific software to run scenarios: commercial and open source IMS-LD compliant LMS already exist. As scenarios are written in this standard, they benefit from common e-learning content management (LCMS) tools. For instance, a classical IMS-LD authoring tool may be sufficient to edit an E-Lab scenario, even if not very ergonomic for this specific use.

Overcome difficulties

Three difficulties appeared in this strategy. The first one comes from the need for a mechanism to automatically describe and gather similar real or virtual apparatuses. Second one resided in the choice of an e-learning standard, rich enough to embody an e-lab scenario and also able to code links to external objects (information and methods), such as an interface software with an apparatus. Last one was to find a way to extend an existing authoring tool to provide authors with the necessary ergonomics for specific work with real (in R-Labs) or virtual (in V-Labs) apparatus.

Provided solutions

To represent sets of similar apparatuses (same classic functionalities but different manufacture), we appealed to an OWL ontology based system, described in the following sections.

The second one resided in the choice of an e-learning standard, rich enough to embody an e-lab scenario and also able to code links to external objects (information and methods), such as an interface software with an apparatus. Our study led us to IMS-LD specification.

The last one was to find a way to extend an existing authoring tool to provide authors with the necessary ergonomy for specific work with real (in R-Labs) or virtual (in V-Labs) apparatus. We still work on this aspect.

Global architecture

The architecture corresponding to our proposition features five fundamental parts, as shown in figure 2. Two of them are generic (used by other educational contents):

- a general IMS-LD compliant authoring tool, such as Reload[2] editor, in order to edit generic IMS-LD scenarios from a pedagogic point of view without dealing with teleoperation aspects and
- an IMS-LD compliant LMS (Moodle[3] with Coppercore[4]) to run in real time scenarios for learners and tutors.

The others are specific for E-Labs:

- a specific authoring tool to edit resources in generic scenarios: once the pedagogic part is written, this tools helps authors to associate a generic scenario to an apparatus template and to select which functionalities to provide to actors according to their activities. The merging of this tool with the previous general authoring tool is envisaged to enhance ergonomy.

- An E-Lab management middleware, called ELaMS (Electronic Laboratory Management System) which features three families of functions:
  - managing a series of apparatuses. At set-up, a manager associates a new apparatus to a corresponding template. Next, he links every generic functionality known in the associated template to URLs, which will be used during scenario play, to call each real apparatus functionality (re-initialization, parameter change, run/stop, ...).
  - Adapting generic scenarios into specific ones to allow their use for a specific apparatus. Before adapting, a compatibility test is performed. As an apparatus does not have to furnish every functionality declared in its corresponding template, there may be some functionalities unavailable on some apparatuses. The first task of the test consists in checking whether there exists apparatuses corresponding to the scenario template on local platform. Second step consists in checking whether every functionality required by a given scenario are locally available. Afterwards, adaptation consists in replacing URLs pointing towards generic functionalities in a template into URLs pointing towards a real web server which is explained in the following item.
  - During E-Lab sessions, redirecting functionnality calls sent by LMS (automatic actions programmed in scenarios to parameter the apparatus according to current activity, for instance) and actors (to teleoperate the apparatus) to the corresponding apparatus. In fact, ElaMS features a web server serving theses URL and acts as a *driver* for an apparatus: it plays the role of interface between (LMS+actors) and (apparatus). It is envisaged to introduce an intermediary stage to deal these calls between several compatible apparatuses according to their vacancy when several groups play together with same apparatuses.

- An Ontology Management Server, called OntoServ which plays the role of generic interface between previous specific authoring tools and ElaMS to give information about templates and their functionalities, for instance:
  - *What are the available templates ?*
  - *What are the components and corresponding functionalities for a given template ?*
  - *What are the functionalities associated to a given component for a given template ?*
  - ...
  - This server is unique and public so that any authoring tool or ElaMS can communicate with it through XML-RPC requests. Ontologies are by now edited with Protege[5] software but a tool to create new templates should be available to help user in this task.

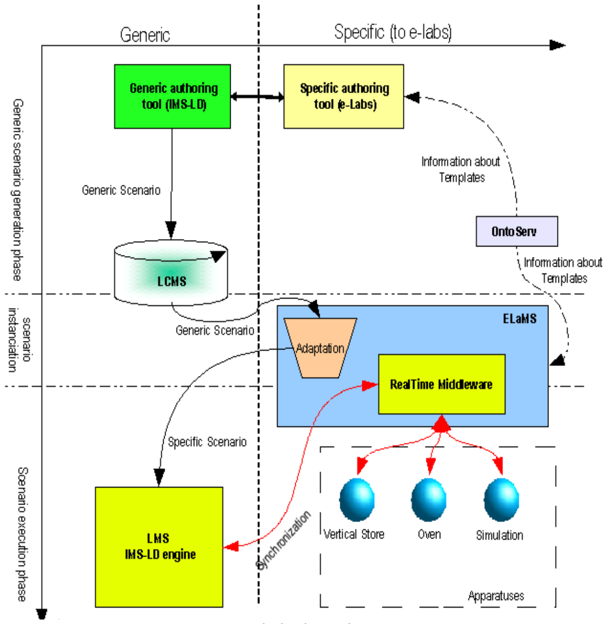

Fig. 2: Global architecture

IV. EXPERIMENTATION

In this section, we describe our first experimentation, at small scale, of the current platform in automation discipline.

Main objectives

The main goals of this experimentation were to validate previously presented solutions (for more details, see [1] and [17]).

More specifically, we wanted to answer some questions:

- *What is the opinion of learners and tutors about e-Labs in a real situation ?*
- *Does the proposed model answer real e-Labs needs ?*
- *What are the important aspects to keep in remote configuration ?*
- *What are the limits and the gaps of our approach ?*
- *What are the points to be improved ?*

Remote system description

The physical setup used to validate our research is a vertical store (see fig. 3) which is used in local laboratory training for *Industrial Engineering Department* students at National Institute of the Applied Sciences (INSA) of *Lyon*, in France. It features a tower enclosing a loop which drives nacelles capable of holding small specific pieces. Learners manually put or remove pieces from nacelles through a door on the right side in order to simulate a real stocking use. An industrial Programmable Logical Controller (PLC) controls the loop motor through several sensors and a raw Human-Machine Interface (HMI).

Fig. 3: Local automation setup: the initial system.

---

[2] http://www.reload.ac.uk/
[3] http://www.moodle.org/
[4] http://coppercore.sourceforge.net/
[5] http://protege.stanford.edu/

For remote access need, some adaptations of this initial system were necessary. It has become an experimentation platform to test our developments while still being used for local laboratory sessions. The first extension was the addition of a robotic arm (see fig. 4) to (un)load the store at distance, as learners cannot directly do it at distance. This system is controlled by a second similar PLC.
The second extension consisted in making both PLCs reachable by any host from Internet. This was easily done since every modern PLC features Ethernet communication ports. In our case, we added an Ethernet communication module per PLC; these specific *Schneider* modules feature an integrated web server which is initially programmed for remote servicing and able to be customized for other purposes such as remote laboratory. In our case, these servers ship applets to be run to get real time state of both systems and apply orders as if these orders came from the local HMI.

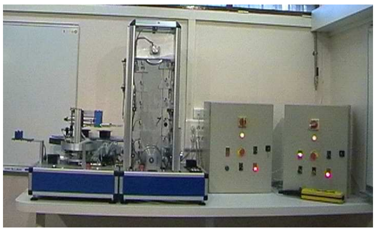

Fig 4*: Our automation TEST platform*

Interaction with the physical setup
At distance, learners have the possibility to remotely control the physical setup in two manners (using VRML 3D reconstruction or by remote PLCs programming) and to visualize it also in two manners (a VRML 3D reconstruction and a webcam). A 3D VRML representation run by the *Blaxxon*® plugin and linked in real time with the store PLC by a *JAVA* applet, provides a virtual view which user can orientate as he wishes. Two other applets propose a 2D reconstruction of both loader and vertical store HMIs (figure 5 illustrates 3D view and vertical store HMI).

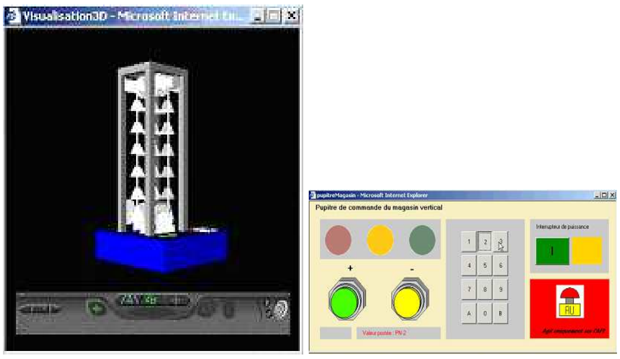

Fig. 5*: VRML 3D reconstruction for remote visualization and remote control board*

The PLC programming reveals one of the difficulties in remote laboratory building: making industrial software be used by several learners on remote workstations whose configurations are not under our control. In our case, we set up a *Windows TSE (Terminal Server Edition)* server which deals *PL7Pro* to learners and tutor workstations. This software is run on the server and the display is carried to the client through network. Therefore, each client can work on either Linux or Windows platform and it only requires to use a free TSE client software to do so.

Pedagogical content and scenario execution
The objective of the e-Lab session was to program a sequential system using the industrial software *Schneider PL7Pro*. Through this e-Lab session learners learn how to use *Sequential Flow Charts* (*SFC*) and the *ladder* language to program an application. Learners also discover how a PLC works. The E-lab session needs prerequisites: to best understand the concepts of this e-lab scenario, it is necessary to know about SFC and binary coding.

Figure 7 shows samples of proposed activities. It consists of a set of miscellaneous activities. We distinguish two types of activities proposed to learners: activities without possibility of manipulation (activities description, courses, exercises, downloadable documents …), and activities with possibility of manipulation (see the functioning of the vertical store via the webcam; interact with the vertical store via the VRML model …). Each of these activities is associated to an environment including pedagogical objects and/or tools (in the IMS-LD sense) necessary for the realization of the activity.

Learner evaluation
In addition to real-time observation, learners had to produce a home e-lab report. This report was a simple word document explaining the work realized during e-leab session. Learners had to return this word processed document after a few days by email.

Human actors/ materials organisation
Two different experimentations were conducted in an interval of ten days. The scheduled duration was two hours for each session. Human actors who participated to the experimentation were:

- a tutor,
- a group of learners as potential users of the future e-Lab system and
- a passive observer for the experimentation needs.

Also, for this need, all computers (especially, those used by users) were Multimedia PC, connected to Internet and equipped with a webcam and a microphone,

Figure 6 depicts human and material organisation. Next, follows detailed description of the technical environment of each site.

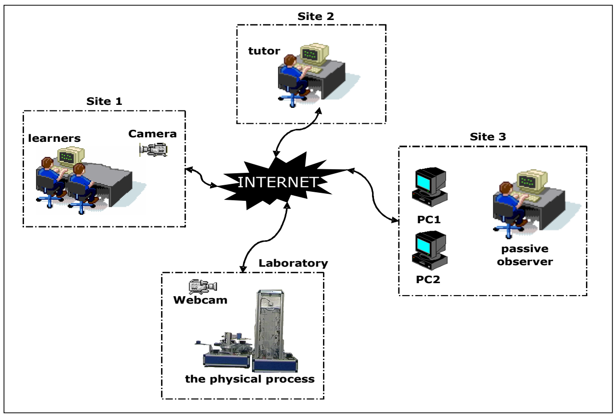

Fig. 6*: Technical environment for experiment*

- Learners:
  as in presence situation, learners work in groups. Two groups participated to the experimentation (one group per experiment). The learners were students at the Industrial Engineering department of INSA Lyon. Some of them followed the same session in a local lab in first year. Thus, these last ones could tell us "the differences between the local lab and the remote ones".

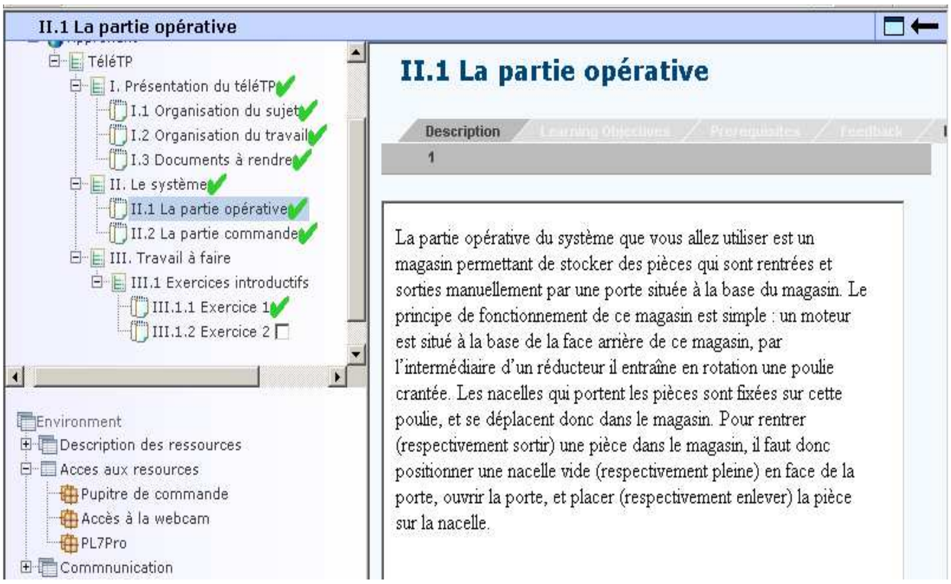

***Fig.** 7: Screen-shot of the learner runtime environment*

- Tutor:
  the tutor is lecturer in automation discipline in the same department as learners. He has a good experience and knowledge in computer science and multimedia. Besides, he is tutor for the same local lab session in first year of engineer cycle. This enables to have a point of view from a person who conducts both local and remote laboratories.
- A passive observer:
  this person checked the good progress of the experiment. There was no direct contact between this person and other users (tutor and learners). For the experimentation need, the learners' screen was backed up and recorded on the passive observer PC via VNC software. Moreover, the passive observer could watch learners' screen in real-time.

For hardware, two computers were used in the experimentation:

- First computer (PC1) included:
  - an application server for remote PLC programming. We set up a Windows TSE server (Terminal Server Edition) which dealt Modicon® PL7Pro (an industrial software for PLC programming) to learner and tutor workstations.
  - ELaMS application, for apparatus management system as described in the architecture section.

- Second computer (PC 2) included :
  - Moodle, as LMS platform,
  - Coppercore as IMS-LD runtime environment, integrated into Moodle platform. Both jointly provided an LMS compliant IMS-LD.

Measurement devices

For the post experimentation needs, learners and tutor screens were recorded using *River Past Screen Recorder pro 6.1* ® for post analysis. Screens were recorded in *DivX* format to spend minimum space onto the hard disk.
Learners were filmed with a Digital Video camera for post analysis. This video has been mixed with the screen shots to obtain a global vision of the experiment progress.
At the end of the experimentation, a questionnaire was given to learners. Also, a semi formal interview was organized in order to get learners' opinions.

Remote learner desktop control

A remote desktop control was necessary so that tutors can take control of learners' desktop when they are stuck. Professional software exist (*Symantec PCAnywhere®, Microsoft NetMeeting®, ExpertCity GotoMyPC, AT&T Cambridge Laboratory Virtual Network Computer VNC …*) and are the best way to implement this complex functionality. For the experimentation, we used *VNC.*

Human communication

For human communication, two webcams were installed respectively on tutor's computer and on learners' computer. In our case we have used the *Microsoft MSN 7.0* software. Out of proposed functionalities, this software enables audio/video communication, text chat, black-board and file exchange.

First results

This experimentation was done at a small scale. Other experimentations at a larger scale are programmed with other tutors, learners, apparatus, and disciplines in order to definitively validate our e-Labs model. We could however draw some interesting conclusions from video recordings and post-experimentation interviews.

For tutors and learners:

- human communication, especially audio communication, must be of a high quality.
- Learners cannot stand more open windows at the same time on the same screen. The use of a second screen could be envisaged for remote apparatus evolution (live video), for example,
- The restitution of the experimentation environment (video+sound in our case) is an important point for telepresence quality.
- A remotely driven camera is not a gadget; the ability to zoom on specific running parts in a debugging process is essential and increases observability level.
- Security is an important aspect: an operator has to remain near the apparatus(es) because of legal statements; one physical person has to turn on/off power and be able to push on emergency button in case of risk.

For tutors:
- remote screen control software allows tutors to efficiently assist and help learners;
- working with more than one group and apparatus at the same time assumes using a specific environment to help tutor in this task;
- using an intelligent tutoring system can considerably reduce the role of the tutor, but his presence remains indispensable, so that learners can, at any time, ask questions and for assistance.

For learners:
- fast response time from apparatuses, tutor assistance and group working help learners to remain motivated.
- 3D VRML reconstruction is an interesting idea to interact apparatus.
- Teamwork allows faster learning process, even if there is often an ascendancy of one of the learners on the others during manipulations.

As a conclusion, tutor and learners expressed their satisfaction with the proposed e-Lab solution. These results could certainly be generalized for other disciplines. This small scale experimentation will be followed by bigger ones with other tutors, learners, apparatus, and disciplines in order to definitively validate our e-Labs model.

V. CONCLUSION

This paper outlines first experimentation results obtained on our current research about a generic architecture for E-Labs. This architecture integrates e-Labs in an existing e-learning environment and permits to reuse traditional functions provided by modern E-Learning platforms: authoring, scenario playing, daily administration communication, evaluation tools, ... while focusing on E-Lab specific needs. Thanks to templates representing classes of compatible apparatuses, an E-Lab scenario is no more dedicated to one specific apparatus but it can be edited and reused by other tutors on other apparatuses as for any standard pedagogic content.
Other experimentations with other tutors, learners, apparatuses and disciplines are programmed to definitively validate the proposed life-cycle and architecture and some improvements concerning the sharing of apparatuses between simultaneous learners are currently studied.

REFERENCES

[1] Benmohamed H., Lelevé A. and Prévot P., "Remote laboratory: new technology and standard based architecture", *1st International Conference on Information & Communication Technologies: from Theory to Applications (ICTTA'04)*, Damascus, Syria, April 19-23, 2004.

[2] Berntzen R., Strandman J.O., Fjeldly T.A. and Shur M. S. "Advanced solutions for performing real experiments over the Internet", *International Conference on Engineering Education*, Oslo, Norway, August 6-10, 2001.

[3] Saad M., Saliah-Hassane H., Hassan H. and El-Guetioui Z., Cheriet M., "A Synchronous Remote Accessing Control Laboratory on the Internet", *International Conference on Engineering Education*, Oslo, Norway, August 6-10, 2001.

[4] Beier K.P., "Web-Based Virtual Reality in Design and Manufacturing Applications", *1st International EuroConference on Computer Applications and Information Technology in the Maritime Industries*, Potsdam, Germany, March 29 – April 4, 2000.

[5] Chang C. C. "A Web-Based Interactive Environment Based on Virtual Reality Simulation for Constructive Learning", *World Conference on Educational Multimedia, Hypermedia and Telecommunications (ED-MEDIA)*, Montreal, Canada, 26 June–1 July, 2000, pp. 191-194.

[6] Hassan C., Tuschak R., Vajk I., Bars R., Hetthessy J. and al., "A New WEB/MATLAB Based System in Control Education", *14th World Congress of IFAC (International Federation of Automation and Control (IFAC'99)*, Beijing, China, July 5-9, 1999, pp. 241- 246.

[7] Tilbury D., Luntz J. and Messner W. "Controls Education on the WWW: Tutorials for MATLAB and SIMULINK", *American Control Conference*, Philadelphia, June 24-26, 1998, pp. 1304-1308.

[8] Marín R. and Sanz P.J. "Grasping Determination Experiments within the UJI Robotics Telelab". *Journal of Robotic Systems 22 (4)*, pp. 203-216, 2005.

[9] Yu L., Tsui P.W., Zhou Q. and Hu H., "A Web-based Telerobotic System for Research and Education at Essex", *IEEE/ASME International Conference on Advanced Intelligent Mechatronics Proceedings*, Como, Italy, July 8-12, 2001.

[10] Girault I., D'Ham C., Caix-Cecillon C. and Bettega H., "Apprentissages en chimie par des expérimentations pilotées à distance", *actes de la conférence Environnements Informatiques pour l'Apprentissage Humain (EIAH'2003)*, Strasbourg, France, April 15-17, 2003.

[11] Chen S. H., Chen R., Ramakrishnan V., Hu S. Y., Zhuang Y., Ko C.C. and Chen B. M., "Development of Remote Laboratory Experimentation through Internet", *Symposium on robotics and control*, Hong Kong, China, 2-3 July, 1999, pp. 756-760.

[12] Gillet D. and Fakas G. "eMersion: a new paradigm for web-based training in engineering education", *International Conference on Engineering Education (ICEE'01)*, Oslo, Norway, August 6-10, 2001.

[13] Pernin J.P., "M.A.R.S. Un modèle opérationnel de conception de simulations pédagogique", *PhD*, Joseph Fourier university, Grenoble, France, 1996, 272 p.

[14] Buitrago G.C., "Simulation et Contrôle Pédagogique : Architectures Logicielles Réutilisable", *PhD*, Joseph Fourier, Grenoble university, 1999, 269 p.

[15] Sánchez J., Dormido S., Esquembre F. and Pastor R., "Interactive learning of control concepts using easy java simulations", *2nd IFAC Workshop on Internet Based Control Education (IBCE'04)*, Grenoble, France, September 5-7, 2004.

[16] Lelevé A., Meyer C. and Prévot P., "Télé-TP: premiers pas vers une modélisation", *Symposium on Technology of Information and Communication in education for engineering and industry*, Lyon, France, November 13-15, 2002, pp. 203-211.

[17] Benmohamed H., Lelevé A. and Prévot P. "Generic framework for remote laboratory integration", *6th proceedings of the International Conference on Information Technology Based Higher Education and Training (ITHET'05)*, Santo Domingo, Dominican Republic, July 7-9, 2005.