	Learning Technology **publication of** **IEEE Computer Society** **Technical Committee on Learning Technology (TCLT)**	

Volume 7 Issue 3 | **ISSN 1438-0625** | **July 2005**

Guest editorial: Immersive Telelaboratories for engineering designer education 2

The Open Laboratory paradigm for engineering education via e-learning....................................13

On line perspiration: the laboratory experience via Web 16

Learning Objects design for supporting e-learning by Immersive Telelaboratory....................................19

Learning Evaluation in a Context of Immersive Telelaboratory....................................23

VLR: a novel approach for the effective learning of robotics on the Web....................................27

Open Standards for Higher Education in Robotics by Immersive Telelaboratories 30

Remote Laboratories on Mobile Robotics 33

Remote Identification and Control Experiments on a DC Motor....................................36

On Job training in immersive environments 40

ScorNet: a tool for remote experimentation on educational manipulators on the World Wide Web 43

***Regular article:* Adding content in Course Support Environments....................................46**

***Regular article:* Possibilities for Inclusion of Simulations in current Standards for e-learning....................................49**

Open Standards for Higher Education in Robotics by Immersive Telelaboratories

1. Introduction

Immersive telelaboratories for higher education in robotics involve new technological challenges. A telelaboratory is a laboratory where remote access to laboratory hardware is provided. A telelaboratory provides three separate access modes: *proximal*, *remote* and *simulation.* In particular, *immersive* telelaboratories aim to minimize the experience gaps between the three access modes, i.e. audiovisual experiences throughout simulation and web-based remote access modes should be very similar to the proximal one. Towards this end, we need quasi-real time communication, i.e. users ought to perceive changes to the laboratory environment as soon as they occur. Students and educators can access laboratory resources anytime and anywhere; more, institutions may share rare and expensive laboratory hardware, so that each institution may offer a broader range of lab experiences. The immersivity requirement is the key feature, making remote access almost as useful as the proximal one.

In this paper, we describe how an architecture of immersive telelaboratories based on open standards may act as a facilitator for educational purposes.

2. Architecture

A telelaboratory may be analysed in four components: the Lab Server, the Simulator, the Repository and the Web Server. Users (educators and students) connect to the telelaboratory via a Web Client, i.e. a browser.

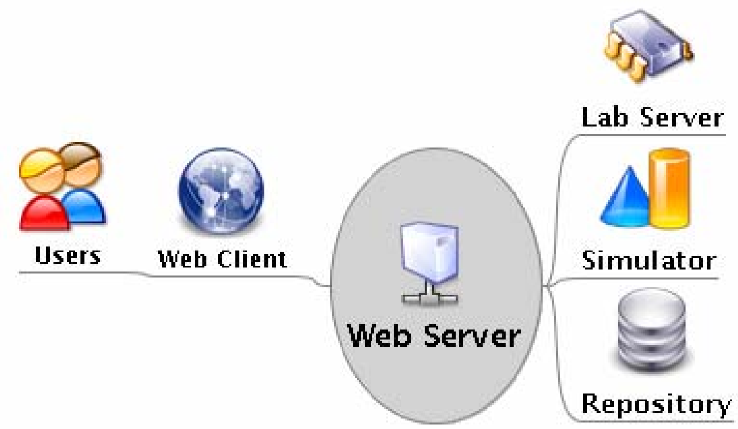

Figure 1. Architecture

The *Web Server* should comply with whatever protocol and language is used by each *Lab Server*, which controls the laboratory hardware. This concern can be dealt with adopting a plugin-based architecture: integration of different protocols should be a matter of developing an appropriate plugin module. The Web server also provides a convenient Learning Management System (LMS) to the *Repository*, which is a database of Learning Objects (IEEE 1484.12.1–2002) of experiments in advanced robotics, and their developments made by the students, ordered along quality indicators. Until now, far less attention has been given by now to open source LMS platforms (see Avgeriou et alt., 2003), although they may be easily customized for educational purposes. Finally, the *Simulator* is a a web application, i.e. a software module that simulates the lab's physical process as closely as possible in the web. The Simulator will be the main environment to construct and develop a variety of simulated experiments as Learning Objects.

3. Developing Software for Telepresence

The Web Server should be an effective information broker for users. Brokarage effectiveness may be divided in three requirements: accessibility, compatibility and immersivity.

Accessibility requirement. Web-based applications should deliver a good quality of interaction with the broadest range of hardware and software without relying on the presence of special software on the user's computer, other than a standard compliant browser (Zeldman, 2003). Moreover, standard web technologies respect accessibility special requirements for people with disabilities (W3C, 2002).

Compatibility requirement. Most modern web applications rely on JavaScript, as the client-side scripting language, which is able to manipulate the objects in the web pages without having them to be fully reloaded. This method permits to perform operations without information pull from the server, and avoiding unnecessary network round-trip delays, and recently it has been dubbed 'Ajax', i.e. 'Asynchronous Javascript and XML'. The Ajax method has become widespread, e.g. web services by Google (Gmail, google Suggest, Google Maps) follows this philosophy (Garrett, 2005). As it does not requires any additional plug-ins to common browsers, the Ajax method is well suitable for quasi-real time needs.

Immersivity requirement. Live video images of the laboratory hardware may be put on the web using the MJPEG protocol (Casey, 2004): many readily available surveillance cameras are able to generate such a video feed with no additional hardware or software, and every common browser can display an MJPEG feed.

We should also care providing an animated drawing or diagram of the lab state. This has significant advantages over a plain video image in terms of less bandwidth: we need only to transmit an array of numbers that represent the state of the system; in case of network delay, the lost frame may be interpolated in a precise way, by using a mathematical model of the physical process.

A third important point is that the same animated diagram can be used to display the state of the Simulator. This makes it possible to make a smooth transition from the simulated experience to the real experiments. We decided to use the Scalable Vector Graphics (SVG) standard, which is going to be almost universally available (Hicks, 2005), and it may also be coupled with client-side JavaScript to build animations, following the Ajax method.

4. Conclusions and further works

Work is underway to build a prototype of immersive telelaboratory for higher robotics education, as described above. Our further directions is to develop a framework for construction of experiments in advanced robotics as Learning Objects. We are going to extend an open source LMS with a software module *ad hoc*.

Acknowledgements

This work is partly supported by TIGER Project (Telepresence Instant Groupware for Higher Education in Robotics, FIRB RBAUM01BZMK), funded by MIUR, the Ministry of Education, University and Scientific and Technological Research, Italy.

References

Avgeriou, P., Papasalouros, A., Retalis, S., & Skordalakis, M. (2003). Towards a Pattern Language for Learning Management Systems. *Educational Technology & Society*, 6(2), 11-24.

Casey J., (2004), Mjpeg. Free Software Foundation, URL: http://directory.fsf.org/vid/view/mjpeg.html.

Garrett J.J., (2005), *Ajax: A New Approach to Web Applications*, URL: http://www.adaptivepath.com/ publications/essays/archives/000385.php. Published 2005, 18 February.

Hicks M. (2005), *Browsers Get Ready for Graphics Boost*, Publish.com. 12 April.

W3C (2002), *User Agent Accessibility Guidelines 1.0*, W3C Recommendation 17 December.

Zeldman J., (2003), *Designing With Web Standards*, New Riders.

Federico Gobbo
Insubria University
DICOM – Department of Informatics and Communication
Mazzini 5, Varese IT-2100 Italy
federico.gobbo@uninsubria.it

Matteo Vaccari
Insubria University
DICOM – Department of Informatics and Communication
Mazzini 5, Varese IT-2100 Italy
http://matteo.vaccari.name