Generic HCI and Group Awareness Collaboration for Remote Laboratory in Virtual World

J. Fayolle, C. Gravier, B. Jailly
DIOM Laboratory, Télécom Saint-Etienne, University Saint-Etienne
Université de Lyon, 42003 Saint-Etienne
25 rue du Dr. Annino 42023 Saint-Etienne
FRANCE
Jacques.Fayolle@telecom-st-etienne.fr http://www.telecom-st-etienne.fr

Abstract*: we present in this paper a remote lab that is generic, i.e independent to the controlled device thanks to a semantic web approach. This kind of approach is very interesting since it allows us to build a remote lab framework whi takes into account a semantic file as input and can deal with any kind of remote device. The graphical user interface and functionalities are described thanks to ontology. The proposed remote lab is also collaborative since it is prove that the collaboration takes an important place in the learning process. The collaboration is made in a virtual world called wonderland allowing sharing of information and audio channels between users' avatars*.

Keywords*: remote lab, virtual world, collaboration, group awareness, semantic web, ontology.*

I. Introduction

The main objective of remote laboratory is to share high technology equipments between universities and companies for different purposes like learning, demonstration before sales or even collaboration on projects. It concerns essentially research laboratories instruments which present common characteristics: high cost (purchase, maintenance), sensitive functioning, slow training and lots of surrounding constraints (temperature, dampness, dust, vibration). The cost induced by remote lab construction leads to eliminate classical devices for this paper since it is generally more difficult to build such lab than buy the needed instances of device. Different solutions have been explored:

- The individual purchase by a school means of course to spend a lot of money but also to rely on a qualified staff to work on the workbench.
- The equipment transportation between different schools implies several technical constraints and a qualified technician must follow the instrument.
- The students' displacement is the most used solution but the transfer of several people and their reception in a research laboratory is not so easy.
- The most promising way may be the distant monitoring of the instrument, which we call hereafter remote laboratory or remote lab.

The concept of remote lab is to make these equipments available for any kind of situations through a simple Internet connexion. We are interested in situations where teachers (or project manager) and students (or project members) are not in the same place and/or at the same time. Three different kind of laboratory are identified, according previous work [1]:
–Local laboratory are universities' lab in which student practice hands-on exercises.
–Virtual laboratory are simulations of an existing laboratory, trying to be as close as possible to the real device.
–Remote laboratory are local laboratory that are accessed via Internet, at different places but at the same time. It can be sum up as "Same time, different place" [2]. Those are real devices remotely controlled over Internet.
Nevertheless, it cannot be said that all laboratories endorsed the same properties. We argue local laboratory are the best kind of laboratory one can make, for learners' sake. This is because students are faced to the real device and have face-to-face tutorial assistance. Of course, local laboratories suffer from drawbacks: learners and teachers must be in the same place, at the same time, in a room sufficiently wide for the learning experience to occur. Those drawbacks lead computer scientists to propose alternative options, for learners/ teachers who cannot make it to the local laboratory.

In the following, a way to control such equipments through Internet connection is presented. A brief revue of remote lab approaches [3] shows that some of current proposals are pure http based. These are LAMP-based solutions, JSP coupled with MySQL Database [4] or other Internet technologies. We strongly believe rich clients are more suitable for such applications. Of course, they suffer from a higher development cost, but they are more precise in reproducing "felt-life" [5]: things that are moving, making particular noises, buttons resistance [6], etc.

The solution proposes in this paper is built according to the following constraints:
–Use of graphical user interface very close to the real one, in order to help the user to manipulate the remote lab as it should manipulate the real device
–Generic approach in order to not rebuild an entire software for different devices
–Integration of collaboration aspects, and especially group awareness, in order to help people to work together on the same device from different places.
More precisely it uses distributed and generic software architecture based on J2EE application server. J2EE is a java framework that has the ambition to centralize functionalities on server side thanks to java-objects. Next, fall into question the choice between virtual and remote laboratories. Several research works tend to demonstrate that online laboratory are "the ultimate solution" [7] and simulation is "the old way of doing things" [8]. Nevertheless, we have a subtle thought as we think simulations are intended to deliver laboratory facilities to the door of the students, when a remote laboratory is not a way to look at. Many drawbacks to simulations have already been related [7, 9]. We strongly believe in two arguments. Firstly, simulation cannot provide an authentic experience as, at its best, can only deliver partial representation of the reality of use of the instrument. Then, the limited functions set usually implemented, and the guided scenario associated to the manipulation, lead to an aseptic scenarii of use for learners. We state that it limits what we can call "learning walkthrough": learners' ability to try out things by curiosity. If the simulation does not foresee such tries, the response will not satisfy the learner and the system will miss opportunities of teaching.
Another key point of nterest is obviously the fastest HCI integration. Imagine a software that fits all the previous requirements. The key question would be "what happen if a new kind of HCI client is added in your architecture". Regarding the use case of eInstrumentation, the question transposed is "what happen if a new instrument (significantly different from others existing) has to be remote controlled". Indeed, the time of integration of new collaborative components is really important. There are many ways for a fast integration. Most of all need the user to enter all the characteristics of the device. Some tools could also be created to help the user in characterizing the HCI of the device. Our solution and the corresponding implementation is explained further.
In the remaining of this paper, we will expose the proposed framework, and then its embedded in virtual world in order to increase the group awareness and collaboration between users. We conclude the paper with a discussion on the way this kind of approach can be generalized to any kind of device and remote lab and any kind of virtual world.

II. Current Evolution of Remote Laboratory: Genericity and Collaboration

We started remotely laboratory researches in 2000 [10], based on a network analyzer we wanted online, thanks to a TCP-IP direct application, replaced in 2003 by a Corba framework [11] and in 2006 by a complete J2EE application server [12]. These evolutions are mainly due to the lack of interoperability in heterogeneous information systems. This successful pedagogical experience makes us think about putting another device online: an antenna workbench. Of course, unlike the network analyzer, the antenna workbench conveys mechanical experiences (moving antenna and starting/ stopping motors). The resulting GUI, however, are close to one another, because the GUI displays the same kind of widgets, whatever the device is (buttons, led, curves, objects moving, menus, etc.). Besides, we become aware we were about to reinvent the wheel each time we want another device online. One of the main constraints is therefore to propose a framework independent to the device.

A. On the genericity constraint

We need common description tools of a device user interface in one hand and the commands and protocols used by the dialog with the user, whatever the device is. Semantic web approaches and especially ontology is a solution to this problem. To be short, an ontology is a description of nature and composition of something. Mainly, the ontology is a vocabulary where are defined classes and properties, according to their field of application. Therefore, we could use ontology to describe the Graphical User Interface of the device and the functionalities it uses. With such an ontology (see fig. 1), we are able to conceptualize and instantiate the complete Graphic User Interface of any device simply through the interpretation of the semantic file describing the device.
This means the vocabulary is common to all appliances and is instanced once. Over this vocabulary, we produce a single file in OWL format, one per device. Using this ontology-based approach, we are able to describe in OWL format a network analyzer, an antenna workbench and we are about to dress the OWL file corresponding to an optic fiber stretcher, very different kind of devices but described thanks to the same vocabulary.

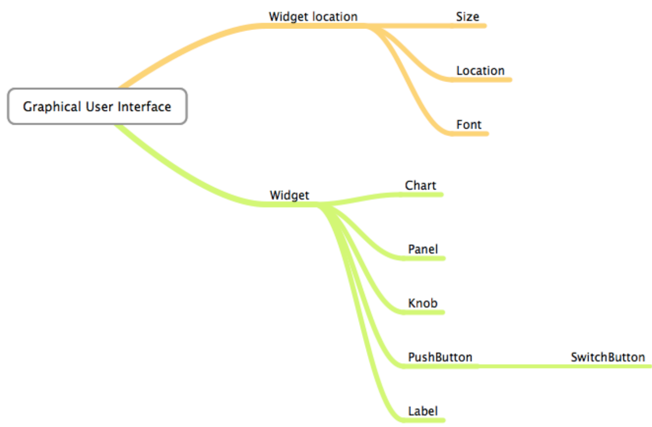

Fig 1: Ontology used for the description of the graphical user interface of a remote device.

B. HCI design

The second main point in a remote lab implementation concerned Human Computer Interaction. Our position is to isolate the HCI description from the platform because the HCI is related to the business logic of the application you want to become collaborative. The platform should only be the connector to the middleware underlying. As for the HCI, packaging it within an XML ontology allow to get fast deploy on new devices. Indeed, when a new instrument had to be accessed using a collaborative platform, the more pragmatic approach would be to write its ontology and then use the same generic client (same for all devices) to get it integrated within the platform. The ontology logic is explained in the previous section of this paper. To sum up, it can be said that the description of an instrument concerned both the widgets that are needed to get a representation of this device (position, size, ...) and the sequence of operations that can be performed when playing a sequence of widgets.

These are both the apparence and the usage associated to the instrument that are needed to get a correct representation of the device. The main issue regarding implementation we encounter is that the writing of an ontology is not very easy for novice users (and the platform aims at providing an easy and fast integration process). That's the reason why two other possibilities (rather than writing straight by hand the ontology) are given to our users:
(i) A HCI software that let the user draw the instrument using remote lab widgets and save it to ontology format (OWL).
(ii) Image processing that let ontology to be generated based on a picture of the device.

As we exploited this solution in our teaching, we understood how authenticity of the device displayed is important. Because students mostly learn from hands-on approaches how to use appliances, not how they work. As such, it is very important to be as real as possible, because students tend to be lost when put in front of the real thing. To achieve this goal we take pictures of the real devices and build the remote interface by replacing the real pictures at the place defined by the widget in the semantic files. The result is that the two interfaces (real and distant one) are very close (fig. 3)

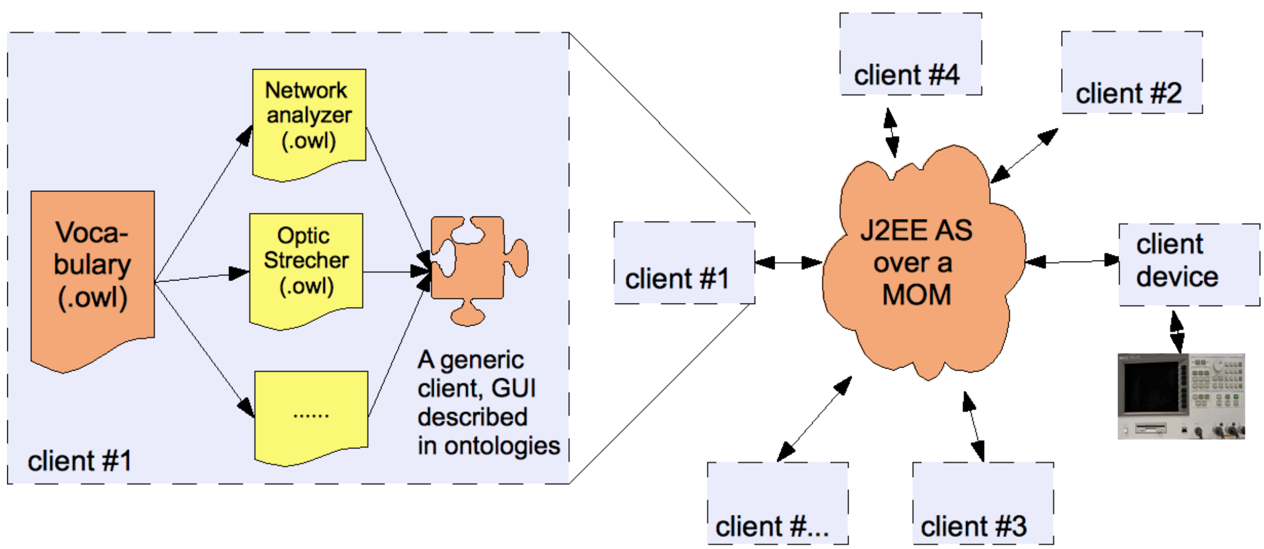

Fig 2: Use of semantic description file to reach the goal of remote lab genericity.

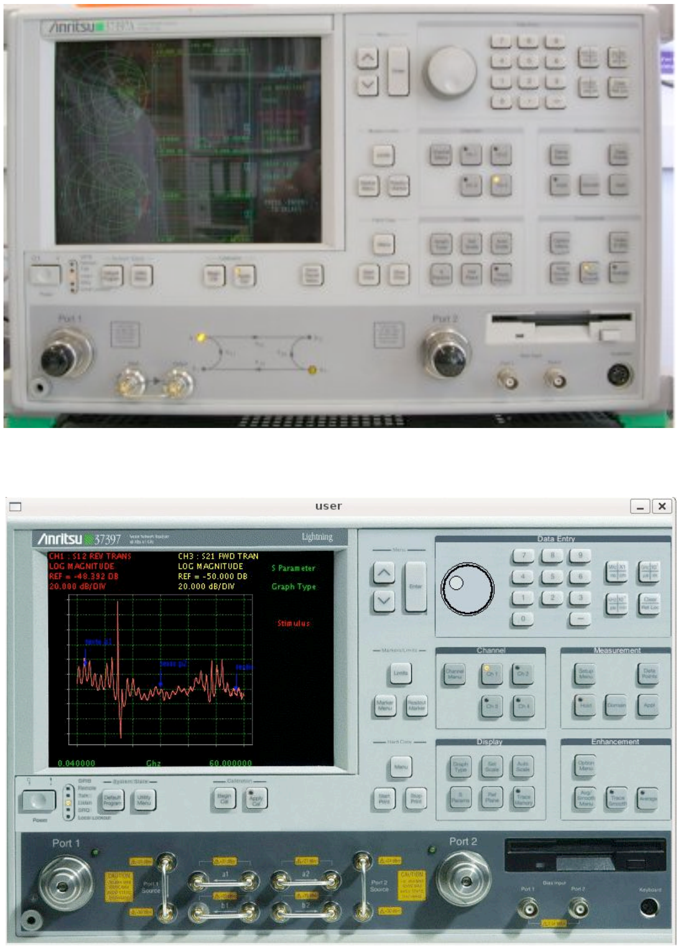

Fig. 3: real and remote interface of a device (network analyzer)

The figure 4 explain the whole architecture used in order to build the semantic file describing the GUI of the remote lab in a generic way.

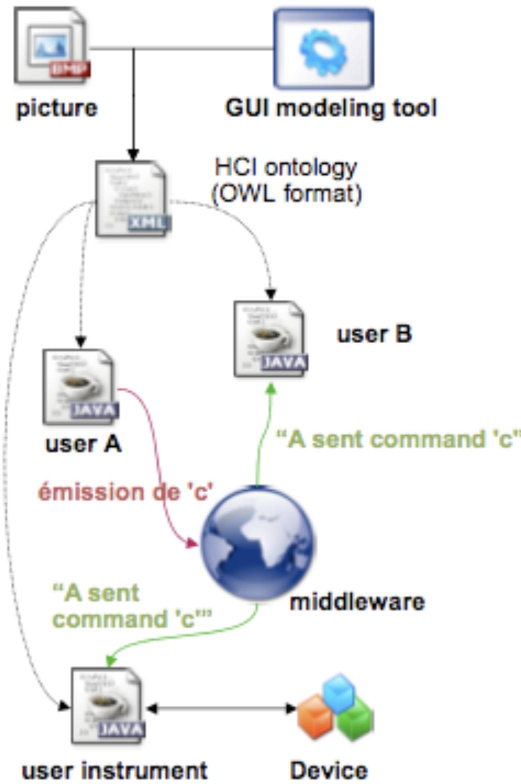

Figure 4 : Complete architecture of the remote lab framework.

Based on the same approach as the Graphical User Interface, we should describe the functionalities associated to a given widget of the remote lab in the OWL semantic file. Therefore, we propose to add some classes to the previous ontology in order to describe not only the GUI but also the functionalities. Moreover, one of our goal is to propose an adaptive GUI to the user's skills. To reach this goal, we add a field which tell if a given widget is related to an expert profile, a medium or a beginner profile. We are sure that this kind of description is not sufficient (a widget can be related to an expert profile is some scenarii and to a beginner one in others). The use of tools from artificial intelligence is probably a good way to qualify the level of use of a set of widgets (and also the skills of a given user). But, we argue that, at the end, we will need a tool to say : this is a beginner tool and it should not appear here since we are in expert mode, for example.

The following figure sum up the previous discussion. It describes the current ontology which includes not only the GUI description but also the functionalities and their level of use.

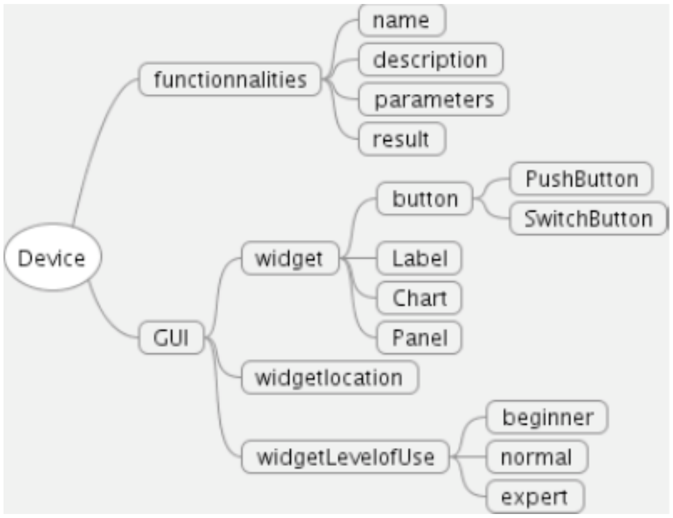

Figure 5 : Complete ontology, description of the GUI and the functionalities of the remote lab.

C. On the collaboration part: group awareness

Nevertheless, one could argue about collaborative sharing of a single resource and how this pseudo-concurrent access (as it is collaboration) is managed. What we want to do is to allow different users to collaborate on the same device and to permit to these users to understand what are the actions made by the others (what we call hereafter group awareness). The group awareness is very important to help students to acquire the desire skills for example [13,14,15]. In fact, computer supported collaborative work is defined for us as several users using a remote resource for a common objective. If the objective is not shared among all users, this is not collaborative work any more. That explains why we cannot apply already well known current resource sharing sequencing such as first in first out, round robin or what ever can be used in complete concurrent environment (by analogy of CPU time sharing). This is mainly because the role the user plays in the collaboration is a factor of orchestration. To achieve this goal we first propose to colorize the acknowledgment of user's action (the buttons pressed by Alice will be colorized in red, whereas Bob's actions will be in blue for example). Each action is broadcasted to every user thanks to a message-oriented middleware (MOM) implemented by our J2EE application server (JORAM over JONAS).

However, we think that this approach is not sufficient. Indeed, the group awareness resides on the actions made by the user but also on the location of each user (does it look at the device at this time or does it read another document), the oral communication, etc… This kind of collaboration is already done in virtual world like well-known second life. You can find many kind of tools in order to collaborate in virtual environments. The objective of the following section is therefore to wonder how we can take the best of remote lab and virtual world in order to propose real collaborative remote labs.

III. Collaborative Remote Lab Thanks to Virtual World Approaches

The use of virtual worlds is growing every day in different fields: entertainment, e-education, professional training, health, robotics, etc. ([16,17,18,19]). Virtual environments make users feel like they inhabit the Virtual Environment (usually referred as the "sense to being there"). It is therefore natural to think about coupling Virtual Environments and remote labs for matching socio-constructivism learning theories with the global challenge of Distance Collaborative Learning ([20,21]).

We can cite as example of such virtual world: second life, Sun Wonderland, opensims,... Each of these virtual worlds relies on the same concept for client-server architecture. On the client side, a heavy stand-alone client is involved, which is downloaded and executed by the end-user's computer, sometimes through one click deployment tools such as Java Web start. This application embeds a middleware client, which exchanges information with the server (usually TCP-based protocols) in order to allow the avatars' motions and discussions with other users. The collaboration itself is done through instant messaging, 3D motion in the virtual world and voice over IP channels. The framework is open and can embed some different tools such as a java-based multi agent for student collaboration and exchange (such as the one used in [22]). On the server side, the objective is to propose 3D description of the virtual world and to relay the different actions and place of user' avatars to each connected people. The description of the world itself is performed thanks to 3D models and tools such as blender, Google sketchup, auto3D.

In order to propose a remote lab such as the one described in above sections, we have to embed a remote lab client in one place of the virtual world. Since the proposed remote lab framework is based on J2EE framework, we choose the Sun wonderland virtual world (developed in java technology). This virtual world allows the sharing of 2D X-applications (just about any application that will run natively on a Linux system). With this type of shared application, one user can take control and edit a document in the world while others in proximity can watch. It is easy to pass control from one user to another. These applications, which were designed for single users, are handy for troubleshooting together in a terminal window, or working together on standard desktop applications like Open Office, or collaborative web browsing. We use this tool to embed our Java-based 2D client of our remote lab (see fig. 5).

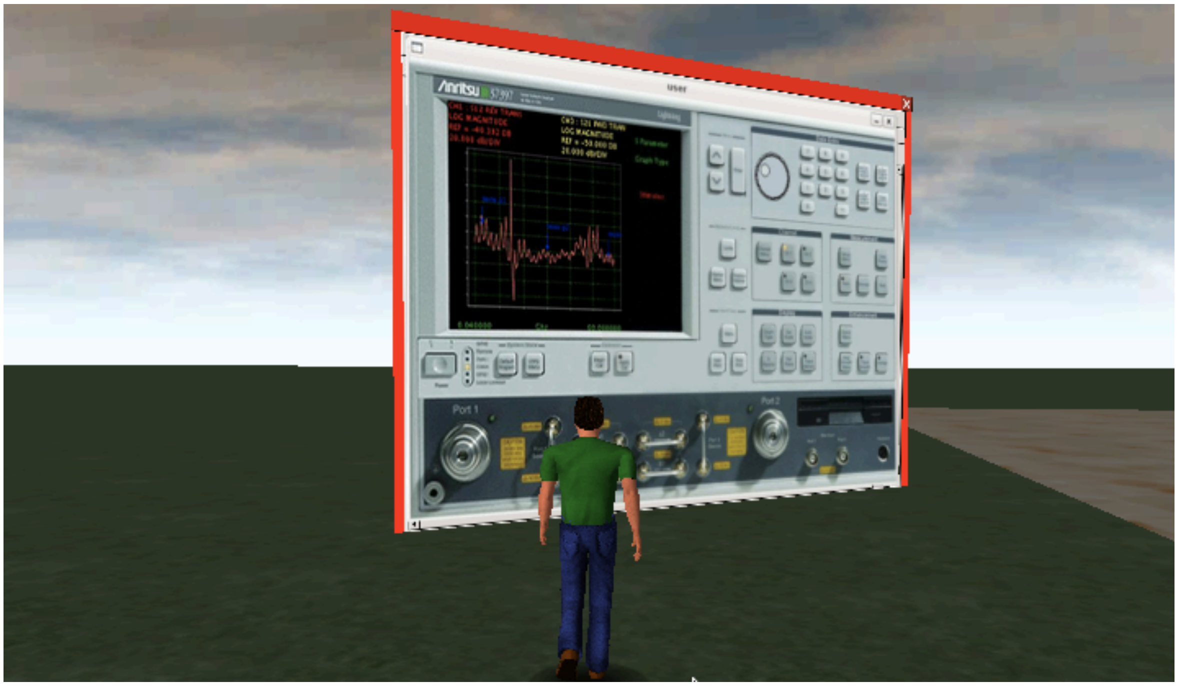

Fig 5: Use of a Network Analyzer remote lab thanks to OCELOT framework in Wonderland Virtual World

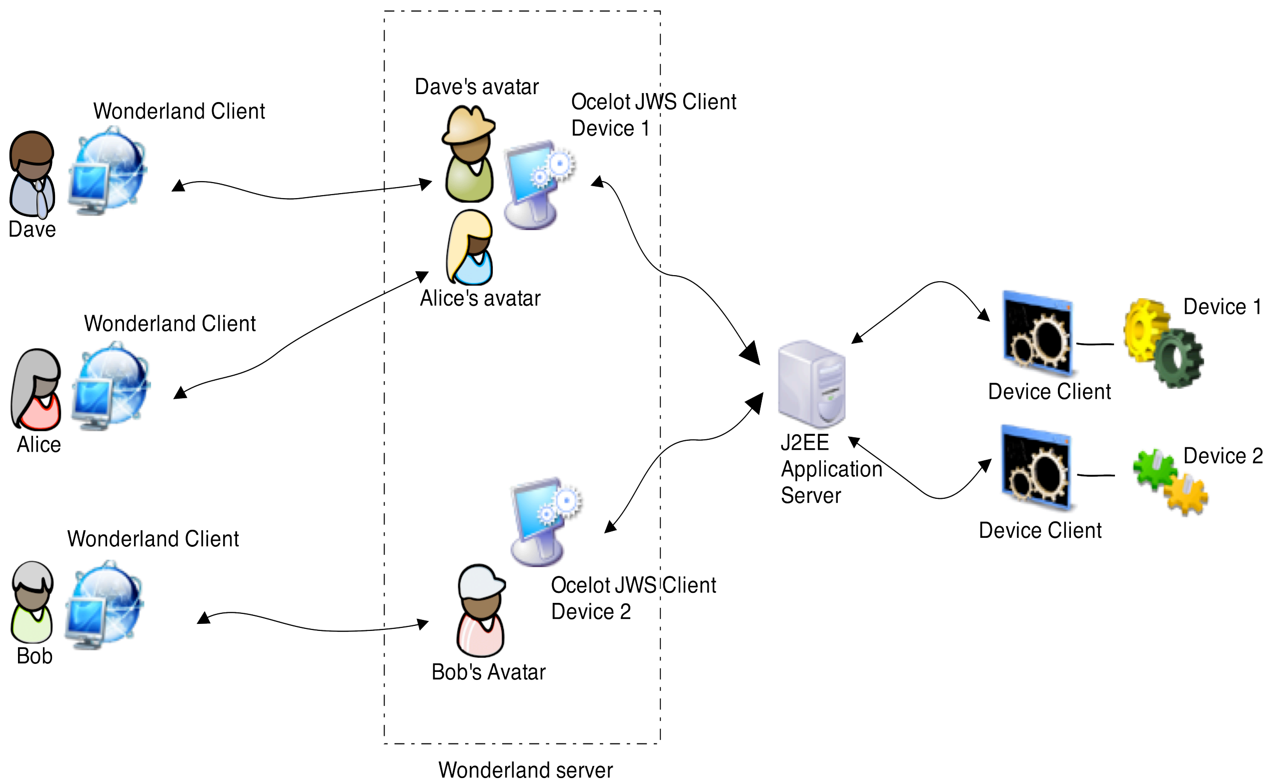

Figure 6 : Complete sketch of collaboration in the remote lab thanks to virtual world servers

Let's take a scenario for better understanding what are the role and aims of each software brick. Suppose that you have two devices, which are remotely controlled through the J2EE Remote lab framework (called OCELOT). Each of them gets its local software, which relays the command between the device and the network. Each device is described in a semantic file stored in J2EE server. Dave (the professor) and Alice (its student) have to conduct a practical session on the device 1, whereas Bob made his homework separately on device 2.

Without virtual world tools, Dave, Alice and Bob launch the OCELOT software as a Java Web start Application on their respective computers. These clients will load the semantic file corresponding to the device on which they want to work. The group awareness between Dave and Alice is made thanks to the JORAM messaging service which relays the command made by Dave on the Alice's screen (for example by tagging the button pushed by Dave in a specific color) as we explained in section 2.2. However, it is very difficult for Dave to know what Alice is doing (reading the course lesson, reading the practical session subject, or watching him doing a demonstration). If we put the JWS client in the wonderland world, the collaboration is made between the Dave and Alice avatars and therefore relies mainly on the wonderland tools: the colour acknowledgement of pushed button still remains in the JWS client but Dave can now see if Alice is watching to the remote device screen, reading the practical session subject (exposed a PDF on a virtual world for example) and discuss thanks to VoIP. It can also simplified the way competitive access to a remote lab is achieved since we can imagine that there is just one remote client correspond to a specific device in a virtual world and the spreading of the real client is made thanks to the wonderland client.

Another advantage of using collaboration tools of virtual world to bring collaboration in remote lab resides on the audio aspects. The spatial layout of the 3D world coupled with the immersive audio provides strong cognitive cues that enhance collaboration. For example, the juxtaposition of avatars in the world coupled with the volume and location of the voices allows people to intuit who they can talk to at any given time. The 3D space provides a natural way to organize multiple, simultaneous conversations. Likewise, the arrangement of the objects within the space provides conversational context. If other avatars are gathering near the entrance to a virtual conference room, it is a good guess that they are about to attend a meeting in that space. It is then natural to talk to those people about the content or timing of the meeting, just as you would if attending a physical meeting. In terms of data sharing, looking at objects together is a natural activity. With the 3D spatial cues, each person can get an immediate sense of what the other collaborators can and cannot see.

IV. Conclusion

We have proposed in this paper a generic and collaborative framework to share a remote lab in virtual world. A semantic approach based on ontology allows us to build a framework independent to the device in the remote lab. The semantic file devoted to each device describes its Graphical User Interface. The remote lab is very close to the real one in order to reproduce felt-life impressions. For collaboration aspects, the remote lab framework embeds a message-oriented middleware that broadcast each action of each user to everyone with a colour value acknowledgment. To reach a higher value of group awareness, we propose to put the remote lab in a virtual world similar to second life but based on java technology: sun wonderland virtual world. There is a number of advantages of using virtual worlds to create remote laboratories that allow immersive and highly interactive user experiences. From our study, it seems that among this advantages, there are:

–The audio relationship between users,

–The 3D location which allows visual collaboration between users,

–The fact that the remote lab is embedded in the virtual world induces different practices as we seen for serious gaming approaches,

–Collaboration between different tools (exposure of course documents as PDF walls for example).

In order to build this kind of systems, the Virtual World has to propose some gateways to other kind of servers, and for this part, the remote lab server has also to propose the correct gateway and interfaces in order to get its commands and results available in the Virtual World. Currently, it seems that the Java language is the better choice for all this architecture since it proposed normalized objects, 3D graphical representation engine, relationship with the materials.

References

[1] Tuttas and Wagner (2001), *Distributed Online Laboratories*. Proceedings of International Conference on Engineering Education, Oslo

[2] Jr and Sharda and J. Lucca (2005), *Computer-supported collaborative learning requiring immersive presence : An introduction*. Information Systems Frontier, Vol 7, 5–12

[3] Gravier, Fayolle, Bayard, Ates and Lardon (2008). *State of the art about remote laboratories paradigms - foundations of the ongoing mutations*. iJOE : International Journal of Online Engineering , 4 (1), 19-25.

[4] Faltin and Bhne and Tuttas and B. Wagner (2002), *Distributed Team Learning in an Internet-Assisted Laboratory.* Proceedings of International Conference on Engineering Education

[5] McCarthy and Wright (2004) *Putting felt-life at the centre of human-computer interaction.* Proceedings of In Reective HCI Workshop

[6] Guss (2003*), Interface metaphors and web-based learning*. Lecture Notes in Computer Science, Vol 2783 168–179

[7] Alhalabi, Hamza, Hsu, Romance (1998): *Virtual labs vs remote labs: between myth and reality*. Florida Higher Education Consortium (1998)

[8] Alhalabi, Hamza, Hsu and Marcovitz (2000): *Remote labs: an innovative leap in the world of distance education.* Proceedings of Information Systems, Analysis and Synthesis ISAS2000 (2000)

[9] B. Andò, S. Baglio, A. Beninato, S. La Malfa, N. Pitrone (2009) *International Journal of Education and Information Technologies*, Advanced Educational Tools in Measurement and Sensors: from remote monitoring systems to magnetic fluids. n°1, Vol. 3, pp 75-84

[10] Bayard, Fayolle, Sauviac and Noyel (2001), *Web Analyzer Project: real instrumentation over Internet Web*, CETSIS EEA 01

[11] Fayolle, Bayard, Sauviac and Noyel (2004*) A general and secure corba framework for distant control of instruments*, IFAC Workshop Internet Based Control Education IBCE'04 Grenoble 5-7 Septembre 2004

[12] Gravier, Fayolle, Lardon and Noyel, *A Distributed Online Laboratory System for Distant Learning*, IEEE/ACM The International Conference on Signal-Image Technology and Internet-based systems (SITIS'2006), Hammamet , Tunisie 18-21 Decembre 2006

[13] Kamariah Abu Bakar, Ahmad Fauzi Mohd Ayub, Rohani Ahmad Tarmizi (2010) , *International Journal of Education and Information Technologies,* Utilization of Computer Technology in Learning Transformation, n°2, Vol 4., pp 91-99

[14] I. Šimonová, P. Poulová (2010), *International Journal of Education and Information Technologies,* Electronic education in the Czech Republic in 2003-10: development and students´attitudes, n°4, Vol. 4, pp 214-223

[15] M. Tutunea, R.V. Rus, V. Toader (2009) *International Journal of Education and Information Technologies*, Traditional Education vs. E-learning in the vision of Romanian business students n°1, Vol. 3, pp 46-55

[16] Callaghan, McCusker, Losada and Harkin (2009). *Engineering Education Island: Teaching Engineering in Virtual Worlds*. ITALICS, Innovation in Teaching And Learning in Information and Computer Sciences , 8 (3), 1-18.

[17] Castronova (2005). *Synthetic Worlds: The Business and Culture of Online Games*. University of Chicago Press.

[18] Diener, Windsor and Bodily (2009). *Design and development of clinical simulations in second life.* EDUCAUSE Australasia Conference .

[19] Prattichizzo, D. (2009*). Robotics in Second Life.* IEEE Robotics and Automation Magazine , 16 (1), 99-102.

[20] Domingues, Otmane, Davesne, Mallem and Benchikh (2009). *A Distributed Architecture for Collaborative Teleoperation using Virtual Reality and Web Platforms*. 6th annual IEEE Consumer Communications & Networking Conference (IEEE CCNC 2009) .

[21] Onyesolu, M. (2009). *Virtual Reality Laboratories : An Ideal Solution to the Problems Facing Laboratory Setup and Management.* Proc. of the World Congress on Engineering and computer science .

[22] Asma Moubaiddin and Nadim Obeid (2008), *International Journal of Education and Information Technologies*, Towards A Multi-Agent Based Model of Argumentation and Dialogue for E-Learning, n°3, Vol. 2, pp 175-183