Towards New Multiplatform Hybrid Online Laboratory Models

Luis Rodriguez-Gil, Javier García-Zubia, *Senior Member, IEEE*, Pablo Orduña, *Member, IEEE*, and Diego López-de-Ipiña

Abstract—Online laboratories have traditionally been split between virtual labs, with simulated components; and remote labs, with real components. The former tend to provide less realism but to be easily scalable and less expensive to maintain, while the latter are fully real but tend to require a higher maintenance effort and be more error-prone. This technical paper describes an architecture for hybrid labs merging the two approaches, in which virtual and real components interact with each other. The goal is to leverage the advantages of each type of lab. The architecture is fully web-based and multiplatform, which is in line with the industry and the remote laboratory community trends. Only recently has this become technically feasible for graphic-intensive laboratories due to previous limitations in browser-based graphical technologies. This architecture relies on the recent HTML5 and WebGL standards to overcome these limitations, and makes use of the Unity technology. To ensure that the proposed architecture is suitable, we set requirements based on the literature, we compare it with other approaches, and we examine its scope, strengths, and weaknesses. Additionally, we illustrate it with a concrete hybrid lab and we evaluate its benefits and potential through educational experiments.

Index Terms—Remote laboratories, virtual environments, architectures, web-based architectures, hybrid laboratories

1 Introduction

Online laboratories have traditionally been classified in two large and distinct groups: virtual and remote ones [1], [2], [3]. The former provide access to a simulation while the latter provide access to real equipment over the Internet. Educators have long debated over how those laboratories compare to hands-on ones, and over which type of lab is more effective. Though no consensus has been reached, research suggests that certain virtual laboratories can be as effective as hands-on laboratories [4] and that some types of laboratories are more or less adequate depending on the set learning objectives [5]. Today, in practise, the line between *virtual* and *remote labs* is more blurry, because there are laboratories which have characteristics of both.

Traditionally, these types of lab have certain aspects in common which result in a different set of advantages and trade-offs [6], [7]. Traditional virtual laboratories require no physical space, are highly scalable and can adapt reality to fit the teaching needs, such as by simplifying it or by displaying unobservable phenomena. Traditional remote laboratories rely only on real equipment so they provide real data and include authentic delays and unanticipated events such as measurement inaccuracies, through which students can learn about the complexities of science [8].

Currently, efforts are being dedicated to the research of advanced forms of lab that not only mimic a hands-on experience but provide their own new features and advantages, which may not exist in a real hands-on lab. One of these models are *hybrid* labs. The definition of *hybrid* lab is not clearly established in the literature, but, in the context of this work, a *hybrid* lab is simply a lab which mixes virtual and real components [9]. Some authors do not use the word *hybrid*, but simply refer to these labs as either *virtual* or *remote*. Fig. 1 shows where this lab model fits within the traditional characterization of labs. Thus, hybrid laboratories try to leverage some of the advantages of virtual and remote ones, mainly by being able to provide realism, cost-effectiveness and additional features such as gamification or virtual environments.

In this line, some works have applied augmented reality (AR) to a robotics remote lab [10], [11]. Researchers from UNED have applied AR to a hybrid lab for the control of a thermal process [12], [13] and to a hybrid lab for the control of a three-tanks system [14]. Researchers from the University of Deusto have integrated remote laboratories into the SecondLife[1] virtual world [15] and developed a hybrid FPGA lab which overlays a virtual watertank model over a real FPGA controller, allowing bidirectional interaction with the board. Other works from the University of Ulster have integrated an electronics remote lab in a gamified virtual world [16], [17] and in a serious game [18]. Authors from the Polytechnic University of Madrid have integrated electronics remote laboratories in a virtual environment [19]. Other researchers from TU Graz and MIT have integrated a simulation and a remote lab to build a collaborative Wonderland-based virtual world [20].

In Section 2 this paper proposes a set of requirements and criteria for a software architecture for the development

• L. Rodriguez-Gil, P. Orduña, and D. López-de-Ipiña are with the Faculty of Engineering, University of Deusto, Avda. Universidades, 24, Bilbao 48007, Spain, and with the DeustoTech-Deusto Foundation, Avda. Universidades, 24, Bilbao 48007, Spain.
E-mail: {luis.rodriguezgil, pablo.orduna, dipina}@deusto.es.
• J. García-Zubia is with the Faculty of Engineering, University of Deusto, Avda. Universidades, 24, Bilbao 48007, Spain. E-mail: zubia@deusto.es.

Manuscript received 23 Mar. 2016; revised 13 June 2016; accepted 11 July 2016. Date of publication 18 July 2016; date of current version 20 Sept. 2017.
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TLT.2016.2591953

1. http://www.secondlife.com

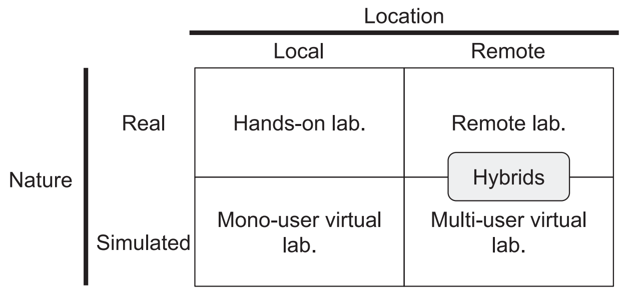

Fig. 1. Hybrid laboratories within a characterization of labs (adapted from [1] and [9]).

of advanced hybrid online laboratories, according to the industry and the remote lab community trends. The most significant of these requirements are *universality* and intensive graphics. The architectures that previously published works implicitly or explicitly describe are analyzed. Section 3 proposes an architecture that builds upon that specification and knowledge. This is achieved by relying on the new HTML5 [21] and WebGL [22] standards (and on the Unity3D [23] technology), which has only recently become technically feasible due to advances in browser-based graphical technologies. Afterwards, in Section 4, the suitability of the proposed architecture is analyzed according to the previously set criteria, it is compared against other approaches, and its strengths and weaknesses are examined. The architecture is illustrated, in Section 5, with an implementation of a concrete hybrid lab. Its benefits are evaluated in Section 6 through educational experiments.

2 Motivation

2.1 Virtual, Remote and Hybrid Laboratories

Research works have concluded that hands-on, virtual and remote laboratories can all be educationally effective, though there is no consensus on which one is the *most* effective; and that is likely to depend on the specific circumstances, educational goals and other factors [4], [5].

Table 1 provides a comparative summary of some of the characteristics of each type of lab. These apply to the *average lab* of its kind but not necessarily to all of them. Realism indicates how close to reality the lab is, and how realistic experimentation with it is felt. Remote laboratories tend to be very realistic because by their nature they rely on real equipment and provide real data, but even then, some works report that only properly designed ones are indeed perceived as realistic [6]. *Can alter reality* indicates whether these laboratories can purposefully simplify reality or even modify it, in order to make certain concepts easier to understand and to hide complexities that are out of a specific educational scope. *Recurring costs* indicates how expensive the lab tends to be in time. The initial investment is not taken into account here because it depends too much on the concrete lab. Hands-on and remote laboratories tend to have the highest recurring costs because they require significant maintenance—including the repair or replacement of hardware components—and supervision. Virtual laboratories have the lowest because, for the most part, once they are developed they require little to no infrastructure. *Augmented features* refers to the fact that some laboratories admit additional software features to enhance or extend the learning experience. Examples of such features would be overlaying a non-visible physical reality on the display so that it can be understood more easily—for instance, an electromagnetic field [20]—or being integrated into a serious game. *Depends on rich media* indicates whether the lab needs to use media such as non-basic computer graphics, videos, or sound. Hands-on laboratories quite predictably don't have such a dependency. Remote laboratories often require only a webcam and basic components to interact with the equipment, so they don't need it either. The average virtual and hybrid lab, however, will need to depict or extend the experimentation reality through graphics, and sometimes additional media.

TABLE 1
Comparative Summary of Characteristics of Traditional Hands-on, Virtual and Remote Labs, and of Hybrid Laboratories

	Hands-on	Virtual	Remote	Hybrid
Realism	Very high	Low	High	Medium
Can alter reality	No	Yes	No	Yes
Recurring costs	High	Low	High	Medium
Augmented features	No	Yes	No	Yes
Depends on rich media	No	Yes	No	Yes

2.2 Requirements

This section lists the different goals and requirements that the proposed architecture should meet and the criteria and rationale that has been used for establishing them:

1) **Universality**: Accessible to as many as possible.
2) **Security**: Minimize the security risk that its users are exposed to.
3) **Power**: Support relatively advanced technical features such as rich media (graphics, sound and video) or low-latency bidirectional communications.

The importance of universality and security for an online lab was supported by a group of experts and discussed in [24].

2.2.1 Universality

Research suggests that this is the characteristic that experts in the Remote Laboratory community value most, and has long been the industry trend. Some criteria can be used to evaluate how universal an architecture is:

- Support in desktop browsers
- Support in mobile browsers
- Reliance on HTTP and HTTPS ports only
- Non-dependency on plugins.

2.2.2 Security

Remote laboratories are usually hosted by institutions such as universities. Their IT teams are often hesitant to offer intrusive technologies to students because they do not want to expose them to security risks, for which the university itself could be liable [24]. *Non-intrusive* technologies are thus preferred. Browser plugins increase the system's attack surface and have been a source of vulnerabilities in the past, so reliance on them also tends to lower security.

Additionally, it is more convenient if the existing infrastructure does not need to be modified to support the online laboratories. For this, they should ideally rely on HTTP or HTTPS, so that no special ports need to be opened or non-standard protocols allowed.

There tends to be some positive correlation between *universality* and *security*, which is why some features discussed here are present in both sections.

Thus, in summary, these criteria can be used to evaluate how security-oriented an architecture is:

- HTTP or HTTPS reliance
- Non-intrusive
- Non-dependency on plugins.

2.2.3 Power

Most remote laboratories, in terms of power and of rich media, need only a webcam stream (e.g., [25]). Hybrid laboratories, however, have similar requirements at this respect to virtual laboratories, and require advanced graphics or other media such as audio. Many rely on 2D or even 3D graphics to provide or enhance the experience. It is also common to require relatively low-latency bidirectional communication with the server.

The power—understood in relation to the number of technical features that it can offer—of a hybrid lab architecture can be evaluated through the following:

- Hardware acceleration
- Audio & video.

The network protocol that it is based on will also influence the capabilities of the lab significantly, in terms of power.

2.3 Previous Experiences

The University of Deusto has had several previous experiences regarding hybrid laboratories. The original WebLab-FPGA-Watertank [26], [27], [28] was a WebGL-based hybrid lab that allowed students to program a physical FPGA board to control a virtual watertank. That proof-of-concept laboratory, which was based in WebGL, is a precursor to the implementation example that is described in Section 5. The experience suggested that hybrid laboratories do indeed have potential and that it is technically possible to control a virtual water tank with a real physical FPGA. At the same time, however, the difficulties encountered in the creation, implementation and deployment of the lab highlighted the potential benefits of a careful architectural design and analysis and the appropriateness of the criteria and requirements that have been described in this work. Also, it raised the question of whether users would find using such a lab satisfactory.

2.4 Difficulties and Goals

Creating hybrid laboratories, especially multiplatform ones, is currently a significant challenge. Though remote laboratories and virtual laboratories on their own are well-established technologies, and much literature exists on the topic, hybrid laboratories are less thoroughly explored, at least in part because until recently graphic technologies in the browser were limited, and often relied on non-standard plugins [29]. There are very few well-documented models, architectures or guidelines for the creation of new hybrid laboratories; and the few hybrid laboratories that exist are often proofs-of-concept that are not maintained or made widely available to the public.

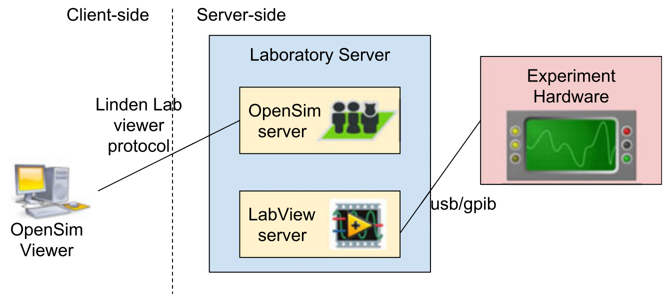

Fig. 2. Architecture of the eLab3D system.

The goal of the architecture that is proposed in this work is thus to establish a model—which leverages the modern technologies and standards that have appeared—upon which hybrid laboratories can more easily and effectively be created. The architecture thus aims to be useful and reusable, and for that purpose an attempt will be made:

1) To rely on open standards and on free technologies.
2) To not be bound to specific Remote Laboratory Management Systems (RLMS)—even though the example remote lab will have to depend on one—.

2.5 Related Works and Architectures

Hybrid labs are relatively new and the literature dedicated to their architectures and models is scarce. There are, however, significant examples of hybrid labs—either in production or as proof-of-concepts—. In this section some of those labs and their architectures, extracted from their published works, from their websites or from other sources, are described.

2.5.1 eLab3D

The *eLab3D* system is an educative virtual world developed by researchers from the Polytechnic University of Madrid. It provides access to both virtual and hybrid labs [19], [30]. In this virtual environment, students can access a virtual campus, which offers labs of different kinds [31]. As of now, most of these labs are simulations (virtual labs) but there are also some electronics hybrid labs available. Students use seemingly virtual electronics equipment through the 3D virtual environment, and can then obtain *real* measurements (from real hardware). The system is designed with collaboration in mind, and students are able to see each other.

The *eLab3D* makes use of an OpenSim-based technology, which is a Java-based technology to help build virtual worlds. Heavily inspired on the SecondLife architecture and protocol, its feature-set is similar. There are several viewers available, though eLab3D recommends a specific one. That one, and most others, are Java-based desktop applications.

The students install Java and the OpenSim viewer, which is configured to connect to the eLab3D servers, they run it, authenticate against the eLab3D servers, and access the virtual campus, from which they can *move* or teleport to the lab they want to use.

Fig. 2 summarizes the *eLab3D* architecture. There is a central *Laboratory Server* which runs different modules. The students use a desktop computer to connect directly,

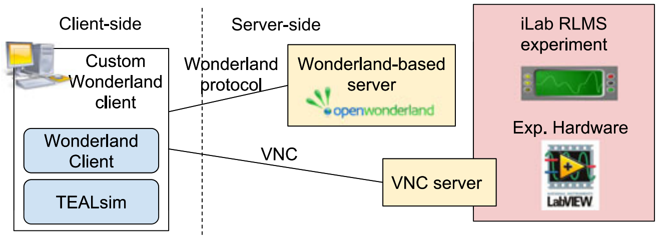

Fig. 3. Architecture of the iLab-TEALsim system.

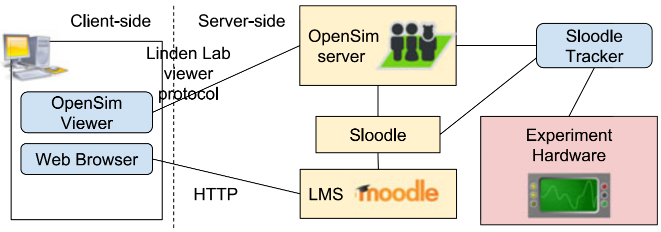

Fig. 4. Architecture of the virtual world version of circuit warz.

through an OpenSim Viewer client, to a standard (not modified) OpenSim server. A LabView server is connected to the hardware (electronic boards) through USB or GPIB. A custom HTTP-based Web Service component communicates the OpenSim server and the LabView server. Table 5 summarizes the technical characteristics of the *eLab3D* architecture.

2.5.2 iLab-TEALsim

The iLab Project [32] is a RLMS developed at MIT. TEALsim [20] is an open-source simulation toolkit to illustrate physical concepts. Using Wonderland [33]—an open-source toolkit for creating collaborative 3D virtual worlds—researchers have created a collaborative virtual environment for hybrid physics experiments [20], [34].

Fig. 3 summarizes the architecture. The virtual environment, multi-user and collaboration features are provided by a Wonderland server. Clients connect to the server through a custom java-based Wonderland Client. The client includes TEALsim-based simulations, and lets students interact with LabVIEW–which provides access to the real hardware—through a VNC client that is included in Wonderland. Table 5 summarizes the technical characteristics of the project's architecture.

2.5.3 Circuit Warz (Virtual World Versions)

The Ambient Intelligence and Virtual Worlds Research Team from the University of Ulster has published several related works in this area. In 2009 they describe the *Engineering Education Island* [35], which is an educational virtual world based on Second Life [36], [37]. Students can use the standard Second Life client to view a virtual increased-scale CPU, learn how it works and perform exercises. Their behaviour and results can be tracked through Sloodle and a purposedly-developed extension for Sloodle (*Sloodle Tracker*). Sloodle is an Open Source technology that integrates Second Life with Moodle [38], [39]. The Sloodle Tracker adds additional capabilities, such as tracking user position. The use of Second Life guarantees that several students can be present at the

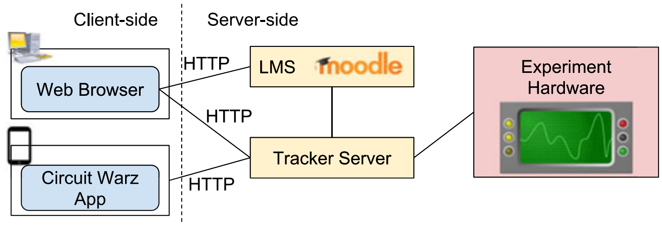

Fig. 5. Architecture of the Circuit Warz 2 system.

same time and interact with each other, though no further specific collaboration or gamification capabilities are provided at this stage.

Later works extend this design and name it *Circuit Warz*. The scheme that is described in [40] adds gamification capabilities and integrates real electronics hardware into the virtual world. Students compete in teams to achieve the highest score in resolving exercises, some of which include resolving circuits that are backed by real hardware. Fig. 4 provides a high-level view of this latter scheme. The server is no longer Second Life itself; but OpenSim, an Open Source project which aims to provide mostly protocol-compatible capabilities, which is open and thus easier to build upon [41], [42]. Moodle, Sloodle and the specific-purpose Sloodle Tracker are still used to register and display user performance, but it is also used to integrate real hardware circuits into the virtual world. Thus, some of the experiments that may be offered to the user are hybrid: they interact with a circuit in the virtual world (for instance, by choosing the value of a resistance) and the system calculates the result through a real-hardware circuit. Table 5 summarizes the technical characteristics of the virtual-world version of the Circuit Warz architecture.

2.5.4 Circuit Warz (Serious Game Versions)

Later alternative versions of Circuit Warz [18], [43] fundamentally changed the approach, architecture and technologies. The OpenSim-based architecture was abandoned in favour of using a custom-made Unity3D-based engine. This implies that there is no longer a Java dependency. Unity3D can easily be exported to different platforms, including desktop, mobile, or in recent versions, even the Web. However, it also implies that multi-user and other virtual-world features are no longer provided by default. This version of Circuit Warz also changes the goal and user experience. It is no longer a multi-user gamified virtual world, but instead a single-user serious game with a backstory in which the player needs to repair a space station by resolving circuits, which may be connected to the remote lab.

Fig. 5 provides a high-level view of this new approach. The lab is no longer a multi-user virtual world—it is now a serious game with a single-user virtual environment—so it no longer relies on the OpenSim server or protocol. Instead, it uses a tracker server to register user actions into Moodle and to obtain results from the hardware circuits. A most significant advantage of this approach—and one of the main reasons of the architecture change—is that Unity3D can generate browser apps, mobile apps and (in later versions) even WebGL, so it can be accessed online from a

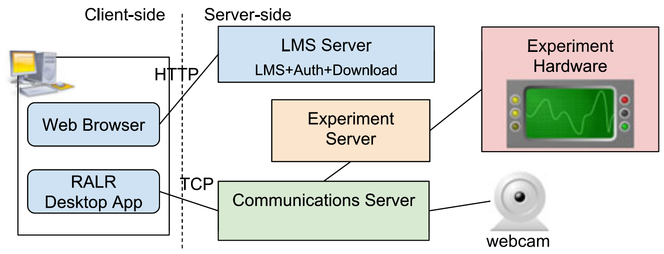

Fig. 6. Architecture of the ARL system.

web page without requiring Java-based viewers.[2] The serious game can be integrated into Moodle (by being displayed within a Moodle course in the browser) or can be played on its own as a mobile app. Table 5 summarizes the technical characteristics of the serious game version of the Circuit Warz architecture.

2.5.5 Augmented Remote Laboratory

A. Mejas, from the University of Huelva describes in his PhD thesis the *augmented laboratory* [10], [11]. The model he describes focuses on the use of augmented reality to *augment* electronics remote labs. The original design provides access to a programmable FPGA board and to a programmable robot. Through the virtual parts of the lab, modules can be added which can interact with that hardware bidirectionally. For example, users can program a robot which interacts with virtual obstacles and a virtual maze. Also, a FPGA-controllable virtual watertank is described, which is similar in purpose to the one that will be described in further sections of this paper.

The *augmented laboratory*, whose architecture is summarized in Fig. 6, relies on Python-based augmented reality libraries and on a desktop applications. The student's workflow includes visiting the LMS server and downloading the required desktop software, authenticating through the Moodle-based LMS, and accessing the lab itself through the desktop software. Table 5 summarizes the technical characteristics of the original ARL architecture. It is noteworthy that newer, modified versions of this architecture exist, which rely on Java Applets and later, on JavaScript.

3 Proposed Architecture

The previous sections show that there are many different valid approaches to design hybrid labs. Most of them have different strengths, weaknesses and capabilities. A comparison can be found in Section 4. The architecture that we propose in this paper takes into account these previous works and ensures that the specific requirements that were listed in Section 2.2 are met.

Fig. 7 shows an overview of the proposed architecture. The front-end of the architecture is web-based. The client is browser-based and it communicates with the server only through a single HTTP channel. In the server-side there is a Remote Laboratory Management System which hides the complexity of the actual physical deployment and which can provide access to different labs. The experiment server provides the logic of the experiment itself and controls the interaction with the hardware. This architecture and the design and technology choices are explained in more detail in the following sections. Table 5 summarizes the technical characteristics of the architecture.

2. The referenced implementation uses the *Unity3D web plugin*, which implies that users require that plugin to run the content. Currently, Unity3D can deploy to WebGL almost just as easily, so that should no longer be a limitation.

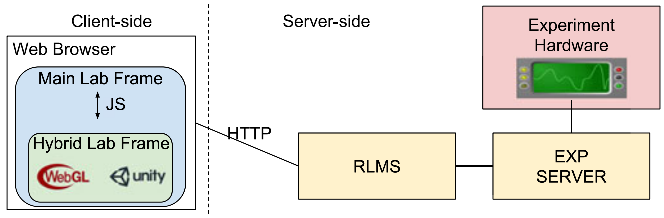

Fig. 7. Proposed architecture.

3.1 Client-Side

Two of the main goals that were established for this architecture were *universality* and *conformance* with the industry trends. Therefore, the architecture, at least from a client-side perspective, needs to be fully web-based. Nowadays this is expected for new applications unless there are restrictions that make it impossible, it is in line with the industry trends and offers many advantages in terms of development, deployment, maintenance, user experience, and security.

Porting online labs and other applications to the Web is not new. The issues this involves, applied to remote labs, were described in [24]. That work also summarizes the strengths and weaknesses of the different technologies that in 2009 were available to meet the different criteria. This paper updates some of them to reflect the technological changes that have since occurred, and partially bases its client-side architecture on the one that results from its analysis.

Also, more recent works in remote lab architectures, such as the one that describes the LaboREM [44] architecture, analyze the advantages and disadvantages of different front-end technologies and propose the use of web-based ones (HTML5) as the most appropriate technical choice.

3.1.1 Classification of Technologies

The client of an online lab can be developed in a wide range of technologies. Two of the large groups they could be classified into are:

1) Desktop clients
2) Web-based clients.

Desktop clients run *natively* in the user's OS, and are thus very powerful: they can generally do anything that the computer itself can do. Unfortunately, they also tend to be less portable and are often bound to a platform (be it, for example, the OS itself in the case of C or C++ applications, or the Java Virtual Machine in the case of Java applications). They are also more intrusive and thus less secure, which especially in the case of an online lab is a problem [24]. Online labs are hosted by Universities, who prefer to avoid exposing their students to such risks, and who may have some responsibility if their students' computers are breached because of them.

Web-based clients run within a browser. By themselves they can run in any platform that has a compliant browser,

TABLE 2
Analysis of the Different Possible Communication Technologies

	Low latency	Requires plugins	Firewall traversal
Raw sockets (UDP)	◆◆◆◆◆	Yes	◆
Raw sockets (TCP)	◆◆◆◆	Yes	◆◆
Basic HTTP	◆	No	◆◆◆◆◆
AJAX (HTTP)	◆◆	No	◆◆◆◆◆
Web Sockets (HTTP)	◆◆◆	No	◆◆◆

they demand no installation and they are subject to the browser's security frame. However, traditionally certain features have been provided by browser plug-ins, which were less restrictive but which have many of the disadvantages of desktop software.

3.1.2 Communication Technologies

An online lab that supports interaction requires bidirectional communication—it needs to be able to send the actions of the user as well as receive the response or the changing state of the experiment—. Table 2 summarizes the different methods that could be used. The *latency* and *firewall traversal abilities* have been analyzed comparatively (more stars is better). Latency refers to the amount of time that sending, completing and replying to a request will typically take. Firewall traversal ability refers to the fact that some protocols are blocked by firewalls more often than others.

The table shows that the lower the latency of a technology, the higher the chance of firewall issues. Particularly, protocols that are not based on HTTP will be blocked by most firewalls, especially if they rely on ports different from 80 (HTTP) and 443 (HTTPS). Basic HTTP and AJAX will have the least issues. Web Sockets are now also part of the HTTP standard and their usage is fast-growing, so eventually they should have good support, but, as of now, many firewalls and proxies do not support them and they often use blocked ports. Considering these characteristics, the client-side communication technology that the proposed architecture relies on will be *AJAX*.

3.1.3 Client Technologies

Online labs are what used to be called Rich Internet Applications (RIAs). They depend on rich media such as graphics or even audio and video. Traditionally these capabilities were not supported by HTML. With new standards such as HTML5 and WebGL, this is no longer the case. The use or not of plug-ins and the support or not of certain features greatly affect universality and security. The choice of client-side technology is thus particularly important for the architecture.

TABLE 3
Analysis of Supported Features in Browser Technologies

	HTML4	Canvas & SVG	WebGL	Flash	Java
2D acceleration	Limited	✓	✓	✓	✓
3D acceleration	✘	Limited	✓	✓	✓
Video	✘	✓	✓	✓	✓
Audio	✘	✓	✓	✓	✓
Plugin independency	✓	✓	✓	✘	✘
Mobile	✓	✓	✓	✘	✘

TABLE 4
Analysis of Browser Support for Different Technologies

	HTML4	Canvas	WebGL	Flash	Java
MS IE	✓	✓	✓(recent)	Plugin	✓
MS Edge	✓	✓	✓	Included	✘
Chrome	✓	✓	✓	Included	Dropped
Firefox	✓	✓	✓	Plugin	Dropping
Safari	✓	✓	✓	Included	✓
Mobile (misc)	✓	✓	Partial	✘	✘

Table 3 summarizes the capabilities of each technology. HTML4 refers to basic HTML, without HTML5-related features. *Canvas* and *SVG* refer, respectively, to the *canvas* and the *svg* elements, which are part of the HTML5 standard, and provide different ways to draw graphics in a web page. These graphics are not accelerated, so its 3D capabilities are limited. WebGL is a standard [22] by the Khronos group, which is related to HTML5 but not strictly part of it. It provides an OpenGL-based API through JavaScript, which gives access to fully-accelerated graphics in the browser. Flash and Java are technologies that have historically been used to provide advanced media features in web pages. Traditionally they have relied on external browser plugins. This has led to significant security issues. The May 2015 McAfee Labs Threats Report states that "Adobe Flash has long been an attractive attack surface for cybercriminals" [45], and that even today the number of Adobe Flash malware samples is growing. Because of this and other issues, the main browsers are dropping or limiting support for these technologies and for addons with native access in general. As of now, newest Chrome does not support them and Firefox is dropping support. Some browsers such as Chrome include their own Flash plugin. Support in different browsers is summarized in Table 4.

Other browser-based technologies that can provide these capabilities exist. *Microsoft Silverlight* is a technology by Microsoft which has capabilities that are similar to Flash. It has been much less popular, is less widely supported, and is no longer under development. *CSS3's* new features could be used to render 2D and 3D graphics, though they are not intended for this use. *Unity Web* is a plugin to run Unity 3D projects on a browser. Newest Unity versions can export to WebGL, which is expected to eventually replace it completely. *CSS 3D* additions to CSS are a powerful part of the HTML5 standard *LabView Remote Panels* can run LabView from a browser. It is a domain-specific technology and requires a plugin with native access in all browsers, in a way similar to Java Applets. It has no mobile support.

3.1.4 Choosing the Client-Side Technologies

As established in previous sections, one of the main goals of the proposed architecture is *universality*. Relying on browser plugins would be contrary to that. This discards most technologies, including *Adobe Flash*, *Java Applets*, *LabVIEW Remote Panels*, *Microsoft Silverlight* or *Unity Web Player*. None of these technologies are consistently supported in mobile phones either, which is also a requirement. After these considerations, the client-side technologies that could be used are *basic*

TABLE 5
Comparison of the Architecture Characteristics

Approach	Application type Remote lab type Interaction type	elab3D Virtual world RLMS Batch	iLab-TEALsim Virtual world RLMS Interactive	CW1 Virtual world Domain-specific Batch	CW3 Serious game Domain-specific Batch	ARL Augmented RL Domain-specific Interactive	Proposed Augmented RL RLMS Interactive
Universality	Desktop browsers	✘	✘	✘	✓	✘	✓
	Mobile browsers	✘	✘	✘	✓	✘	✓
	HTTP/HTTPS	✘	✘	✘	✓	✓	✓
	Built-in tech.	✘	✘	✘	✓	✘	✓
Security	HTTP/HTTPS	✘	✘	✘	✓	✓	✓
	Non-intrusive	✘	✘	✘	✓	✘	✓
	Built-in tech.	✘	✘	✘	✓	✘	✓
Power	Hardware acceleration	✓	✓	✓	✓	✓	✓
	Audio & video	✓	✓	✓	✓	✓	✓
	Network protocol	Opensim	Wonderland	Opensim	HTTP	Custom	HTTP

HTML, *HTML5 canvas*, or *WebGL*. It also discards raw sockets (UDP or TCP) as communication technologies, leaving *basic HTTP*, *AJAX* and *Web Sockets* as possible options.

Of the three communication technologies, *basic HTML* would be the most universal, because it is certainly supported by every browser and server and it can be deployed without any particular firewall consideration. However, nowadays AJAX is just as prevalent and much more flexible. It is also a proven technology, and the most popular communications technology in modern web applications. AJAX is also the choice of some recent remote lab architectures [46]. Thus *AJAX* or the more modern *web sockets* are technologies to consider. *Web sockets* are now part of the HTML standard and are already supported by most important browsers, including mobile ones, and by most HTTP servers. They can provide a bidirectional stream communication channel and a lower latency than request-based AJAX, so they would theoretically be perfect for an interactive remote lab application. Unfortunately, as of now many firewalls and institutional proxies do not yet support them. Because online labs are often deployed behind such systems, *web sockets* are also discarded in favour of *AJAX*, though this choice could likely be re-evaluated in the future, once *web socket* support increases.

Regarding RIA technologies, the most universal would, again, be HTML4. Unfortunately, as Table 3 shows, it is also very limited feature-wise, and not really an option for a virtual or hybrid lab that relies on interactive graphics. The realistic choice would thus be among *Canvas* and *WebGL*. *Canvas* is very widely supported. It is part of the core HTML5 standard, and is supported in every major browser, including mobile. *SVG* is similar, but generally slower for fast-changing animation, and declarative in nature. *WebGL* is more powerful, but its support is not as wide. Because this architecture aims to support labs with advanced graphics, WebGL will be the proposed graphics technologies. Its 3D acceleration capabilities grant it significantly more power, and there are many appropriate development tools and frameworks available. Even though it is slightly less supported than standard Canvas, modern desktop and mobile browsers support it already, and support is increasing steadily.

In summary, the proposed architecture will rely on *WebGL* to provide advanced multimedia features and on AJAX for client-server communication. As a result, online labs that use such an architecture:

- Will be supported in modern desktop and mobile browsers
- Will be able to be deployed behind most firewalls and institutional proxies
- Will be able to use modern accelerated 3D graphics and have real-time communication with a relatively low latency.

3.1.5 Development Technologies

One of the advantages of historically popular plugin-based systems such as *Adobe Flash*, *Java Applets* or *LabVIEW Remote Panels* is that they have a powerful associated set of development tools. The choice of WebGL is convenient at this respect, because being a popular and powerful standard, it has many tools available. One of such tools is *Unity 3D*. *Unity 3D* is a set of tools that can be used to develop 3D interactive applications and deploy them easily into many different platforms, one of which is WebGL.

3.2 Server-Side

The client-server communication takes place through HTTP. The server-side will thus need to have a web server to handle these requests. Beyond that, internally, there are no particular restrictions. Because the proposed architecture aims to be extensible and to support different labs and hardware, it will rely on a Remote Laboratory Management System, which can act as a front-end and that can easily provide capabilities such as federation and integration with other lab frameworks. The Remote Laboratory Management System has a module for the particular lab, which also handles any lab-specific interaction which may be required.

Additionally, although the proposed architecture does not require it, implementations could also rely on the *smart device* paradigm [47]. The *smart device* paradigm aims to decouple the client and the remote lab by establishing a common specification that is shared among different remote labs. This could be used to allow the laboratories to easily use the same virtual model with different types of remote hardware, or even to ‘plug and play’ remote hardware. It would also help integration into Massive Open Online Laboratories (MOOLs) [48].

4 Comparison with Other Architectures

As described in Section 2.5 several different architectures exist and have been used for the creation of different types of hybrid online labs. Table 5 compares the characteristics of these architectures and of the proposed one.

A key aspect that the table shows is that the chosen technologies have a very significant effect on *universality* and *security*. The architectures that are based on Opensim (eLab3D, CW1) or on specific desktop technologies (ARL) tend to be less universal because they tend to rely on native desktop applications which do not run on a browser, are intrusive—they require native access to the Operative System—, do not run on mobile devices and rely on non-standard protocols and ports which are often blocked by institutional firewalls. On the contrary, those that are designed for the newer web standards, including the proposed architecture, can provide advanced features without compromising universality or security.

The main advantage of the proposed architecture over some of the listed ones is thus that it provides significantly greater *universality* and *security*, which were indeed the main goals set in Section 2.2.

It can be observed that in these terms the proposed architecture is similar to the CW3 architecture. CW3 was designed with similar goals and also relies on modern Web standards to provide those features without compromising *universality* or *security*. The approach, however, is significantly different. While CW3 is a single-player serious game that integrates domain-specific remote hardware (circuits to be solved) and whose interaction with the hardware is *batch* in nature—the interaction is discrete; a request is sent to the hardware, which returns the result once calculated—the proposed architecture is designed for augmented remote labs which are interactive and most likely hosted within a RLMS environment. The virtual environment is added upon a traditional remote lab experiment (which is thus *augmented*) and a continuous interaction takes place between the virtual environment and the real hardware: users reserve the hardware for a time, and through that time the virtual environment affects the physical model, and the physical model affects the virtual environment.

Which approach is more appropriate to teach a specific subject (an augmented remote lab, a serious game, or a virtual world) is an interesting question but it is most likely dependent on the subject itself, on the preferences of the teacher and of the students, on the particular context and on other factors. It is thus beyond the scope of this paper.

5 Implementation Example: Watertank FPGA Laboratory

5.1 Limitations of Some Traditional Remote Labs

Many remote labs that exist provide access to a particular hardware development board, such as a FPGA device [49], [50], a PLD [51], a PLC [52] or a microcontroller [53]. These labs are useful because these boards require specific training, and are themselves relatively expensive and hard to setup. The standard FPGA remote labs at the University of Deusto [54] provides access to a Xilinx FPGA Development Board. Students design the VHDL logic and can then program it into the board, interact with it through virtual switches and other inputs that are mapped to real, physical ones, and control a set of LEDs that act as outputs and that can be seen remotely through a webcam.

Although this scheme is by itself very useful, and has been successfully used by the University of Deusto for years, it has some limitations. The outputs are very simplistic. Although the hypothetical exercises that students need to resolve often include devices such as heaters, storage tanks, or industrial devices, to know whether their logic works they need to imagine its results through the LEDs, which are the only actual outputs available. As a result, exercises tend to not be particularly engaging. Additionally, it is often very hard to get a good idea of how a particular logic would perform under real conditions by just seeing the outputs.

The most straightforward solution would be to connect the FPGA device to industrial hardware. Students would be able to program the logic and see how it behaves when applied to a real model. This approach is taken by some remote labs, but it also has several drawbacks. Real industrial hardware and realistic scale models tend to be expensive to purchase. If a straightforward approach is taken, each industrial model will need to be linked to its controller. As a result, to give support to several models, several controller boards would be required as well. Those can normally be used by a single user at the same time. They also require a significant amount of physical space and maintenance efforts. As a result, the costs are often too high, especially because some industrial hardware can also be a security risk.

5.2 The FPGA-Watertank Laboratory

The FPGA-Watertank labs aims to use the hybrid labs architecture described in this paper to provide a hybrid lab that partially resolves some of the aforementioned issues. The lab provides access to a real FPGA board which can control virtual industrial models rather than real, physical ones. This has several advantages. The user can see how their logic behaves under potentially realistic conditions, but at the same time the costs are driven down. Developing and maintaining a virtual industrial model tends to be much cheaper than purchasing and maintaining real industrial equipment. Also, a single controller board could potentially be used with an arbitrary number of different virtual models (e.g., engines, semaphores, robots). As a result, with several users and models, instances of the controller board could be used interchangeably and a much lower number of boards would be needed.

5.3 Components and Scheme

Fig. 8 shows how the main components of the lab relate to each other. The controller is a Xilinx FPGA board, which is the same controller board that is used for the standard FPGA remote lab at the University of Deusto. This board still has the LEDs as outputs, and these LEDs can still be watched by the students through a webcam. However, these outputs are also mapped to a virtual industrial model, which in this case is an industrial water tank. The virtual water tank has two actuators—the water pumps, mapped to the LEDs—. When

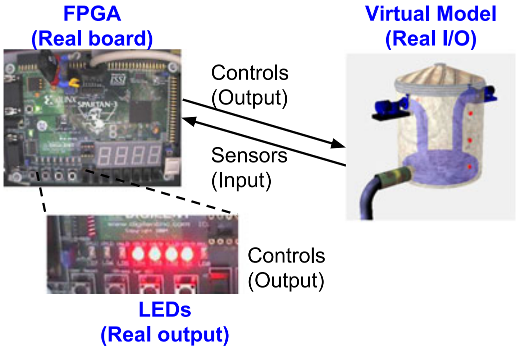

Fig. 8. Relationship between components.

the users' logic turns on a physical LED on the board, the virtual water pump will simultaneously be turned on. The virtual water tank also has several sensors: three water level ones, and two overheating ones. These sensors are mapped into the physical board as input signals. Finally, the water tank model also has a water output whose rate randomly increases and decreases, to simulate a varying water demand. Through this scheme, the real, physical hardware board can be used to control the virtual model.

Fig. 9 shows a screenshot of the Watertank-FPGA lab running in the Chrome browser. The lab is integrated within the WebLab-Deusto [54] RLMS. In the upper left a webcam stream displays the real, physical board that is running the logic that the student has provided. In the upper right the virtual environment, created with Unity3D, is rendered in WebGL, without requiring any plugin or addon. In the lower side, virtual controls can be used to interact with the board.

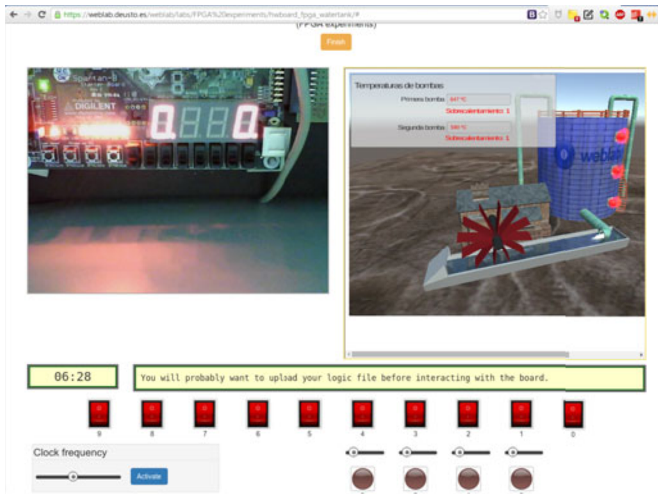

Fig. 9. Watertank-FPGA laboratory running in Chrome.

5.4 Software Design

The lab software is developed according to the architecture that was described in this paper. The client has two parts. First, a web based framework—based on Angularjs—that is integrated within the WebLab-Deusto RLMS and through which students can see the physical board, interact with it and send their logic. Second, a WebGL view that integrates within that aforementioned framework, which displays the industrial model that the board is controlling. This scheme is designed to provide a consistent user experience across user devices—PCs, mobiles and tablets—and controlled virtual devices—though at the time only the watertank is provided—.

Because the virtual model relies only on standard technologies such as WebGL, it is currently compatible with all modern browsers, including mobile ones. Because raw WebGL development is costly and time-consuming—WebGL is essentially an OpenGL for the Web—, the Unity3D toolset has been used. Unity3D provides an integrated graphical editor that greatly accelerates the development of such applications, and which relies on C#, a complex editor, and other technologies. As an additional advantage, applications can easily be built to different platforms, so apart from WebGL, it would be straightforward to build it as a native mobile application.

The lab server is also integrated with WebLab-Deusto, a RLMS. In WebLab-Deusto, labs such as this one have a separate *experiment server*, which could be implemented in any language but which is most often implemented in Python. Though the graphics are client-side, the virtual model simulation itself runs server-side here. This module is also the one with interacts with the hardware—the physical FPGA controller board—.

5.5 Technical Evaluation

A technical analysis has been performed to verify that the lab that implements the architecture conforms to the requirements that were previously established and that it supports important characteristics.

1) *The lab runs on modern browsers without requiring plugins:* The lab relies only on HTML5 and WebGL. It has been successfully tested under recent versions of Google Chrome, Mozilla Firefox, Internet Explorer, Microsoft Edge, Safari, Mobile Firefox (*webgl needed to be explicitly enabled*) and Mobile Chrome.
2) *The lab can be deployed behind institutional proxies and firewalls:* The lab has been tested in the University of Deusto and has been deployed behind its proxy and firewall. It works through standard HTTP traffic in port 80, or through HTTPS in port 443.
3) *Deploying the virtual models is straightforward:* Client-side the virtual models can be developed in Unity3D and built to WebGL.
4) *The system relies on open-source or free technologies:* All the technologies, including the tools, that have been used are open-source or free. Specifically, the significant technologies are: GNU/Linux—open source, for the server-side deployment—, Python—open-source, for the server-side development—, Unity3D—closed-source but free—, HTML5 and WebGL—open standards—, WebLab-Deusto—open-source, a RLMS—.

6 Prospects and Satisfaction Survey

6.1 Procedure

This study, conducted in December 2015, aims to evaluate whether the model has indeed a significant potential for education. Although online labs are sometimes evaluated by measuring their educational effectiveness—for instance,

TABLE 6
First and Second Questionnaires

First questionnaire	
Q1.1	*How much interest would you have in using each of the following technologies to promote your learning?* 4 point scale (1 to 4) for pen and paper, simulators, remote labs and augmented remote labs
Q1.2	*Do you think that using a remote lab (normal or augmented) can help your learning?* 4 point scale (1 to 4) from *helps very little* to *helps very much*
Second questionnaire	
Q2.1	*How much value do you think that using the lab has added to your learning?* 4 point scale (1 to 4) from *very little* to *very much*
Q2.2	*How could it have added more value?* Free comments
Q2.3	*How much value does the virtual environment add to the remote lab?* 4 point scale (1 to 4) from *very little* to *very much*
Q2.4	*Would you have preferred to have used a different system?* Multiple choice (Satisfied with the ARL, pen and paper, simulator, standard RL)
Q2.5	*How much interest would you have in using each of the following technologies to promote your learning?* 4 point scale (1 to 4) for pen and paper, simulators, remote labs and augmented remote labs
Q2.6	*How did the lab work? Check the statements you agree with* Check 1 or more: *There were technical issues*; *Technically worked with no issues*; *I barely learned anything*; *I am satisfied with what I have learned*

by comparing the knowledge gain against other methods such as hands-on labs—this would not be particularly appropriate for this purpose because it aims to evaluate the model itself and not the particular example implementation. Instead, a survey-based study has been conducted. The goal is to ensure that the students find such a model useful and attractive, and that they are reasonably satisfied after using a prototype of a lab based on that model. The surveys have relied on a 4-point Likert-style scale with no neutral.

First, students that had never used or been exposed to a remote lab were briefly described different types of online labs: remote labs, virtual labs, augmented remote labs, simulators. A first survey measured the initial interest that they had in using those technologies to learn. The students were not informed of the steps that would follow or of whether they would later use any of these technologies. Once the survey was filled, the students were asked to solve an electronics problem using the FPGA-Watertank lab. Finally, they took a second survey to measure their satisfaction and whether they believe to have learned effectively. The content of both surveys is included in Table 6, translated to English. The original questionnaires were in Spanish.

All the students in class returned the questionnaire, but not all questions were always filled. Empty or incomplete responses to individual questions were discarded.

6.2 Participants

The participants of the study were 58 first-year students of the Double Degree in Business Management and Industrial Engineering. They were all enrolled in a digital electronics course, and the test was conducted within that subject's context. They completed the tests in-class. None of them had used a remote lab before.

TABLE 7
Results of the Initial Survey's Q1.1: Interest on the Technologies (n=56)

Technology	Mean	S.D.
Pen and paper	2.500	0.981
Simulators	3.107	0.795
Remote Labs	3.250	0.543
Augmented Remote labs	3.357	0.789

6.3 Initial Test

The purpose of the first test was to assess the initial interest of the students in each type of tool, and particularly to determine how the interest and learning expectation with the lab model that is proposed in this work compares against other approaches.

That survey had only two questions. The first to measure their initial interest, and the second to measure their expectations regarding remote labs in general. Table 7 shows the result of the first question, which asks students to grade the interest they have in using different learning technologies in a four-grades scale. Fig. 10 shows the result of the second question, which asks the students whether they believe that a normal or augmented remote lab can help their learning, and which also uses a four-grades scale.

6.4 Second Test

A second survey was conducted after the students had used the FPGA-Watertank lab. 58 students returned the questionnaire. Questions that were not filled were discarded. Its main goals were the following:

- To measure their satisfaction with the lab.
- To check whether their opinion on the potential of the different technologies had changed after using the ARL.
- To evaluate whether they believe the lab model is useful.

Table 8 shows the interest in different technologies (Q2.5), which is exactly the same question as the Q1.1 in the first survey (see Table 7). Table 9 shows the results of Q2.1

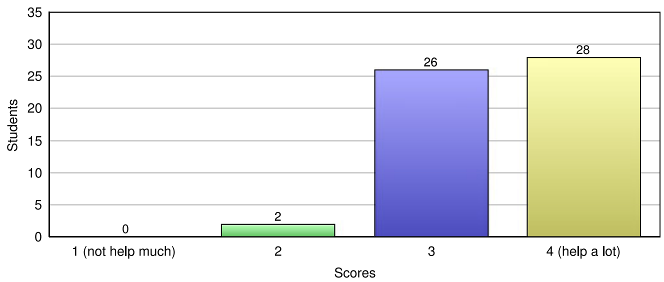

Fig. 10. Results of Q1.2: Perceived learning potential of RLs / ARLs (n=56; mean: 3.464).

TABLE 8
Results of the Posterior Survey Q2.5—Interest on the Technologies (n=53; Scale: 1-4)

Technology	Mean	S.D.
Pen and paper	2.604	0.809
Simulators	3.189	0.646
Remote Labs	3.189	0.585
Augmented Remote labs	3.359	0.676

TABLE 9
Results of the Second Survey: Q2.1 and Q2.3—Perceived Value of Learning and of the Augmented Features (n=57; Scale: 1-4)

Question	Mean	S.D.
Perceived learning value (Q2.1)	3.614	0.522
Perceived augmented features value (Q2.3)	3.158	0.615

and Q2.3, which asks students to grade in a scale of 1 to 4 how they value the contribution of the lab to their learning, and how much they value the addition of a virtual environment to the remote lab.

Q2.4 asked the students whether they would have preferred to use a different technology instead of an Augmented Remote Lab. 43 students out of 56 who answered the question (76.8 percent) would have preferred no other.

Q2.6 asked the students to check the statements that they agreed with. 40 students out of 55 who answered the question (72.7 percent) agreed with the statement: "*I am satisfied with what I have learned*".

Additionally, the questionnaire included a space (Q2.2) for students to include free comments and suggestions to improve the system. Some of the provided ones were the following:

"I agree with the system"

"I think that the system is fine as it is"

"I think it's fine as it is, and it is possibly what has helped me the most to study the subject"

Several students also expressed that they would have liked more practical sessions.

6.5 Discussion

The surveys suggest that the students are definitely interested in learning through tools such as simulators, RLs, and ARLs. Of those, the ARL ranks the highest. This suggests that the architecture that is described in this work to facilitate the creation of such labs not only meets the technical requirements that are set, but can lead to labs that are found useful and possibly more engaging by students. It is remarkable that the opinion of the students did not vary significantly after using the Augmented Remote Laboratory: the interest that they had in using each technology remained mostly the same.

The other questions of the second survey also show that the students, in general, find the lab satisfactory. The students believe that it added value to their learning (Q2.1). Determining whether this perception is true, and to what extent, would require further educational experiments. Also, they find that augmenting a RL with a Virtual Environment is valuable (Q2.3). Questions that measure their general satisfaction (Q2.4, Q2.5, Q2.6) are also positive.

7 Conclusion and Future Work

This paper has described a new model for the creation of Augmented Remote labs which leverages recent advances in web technologies to support the advanced RIA features that hybrid online labs require (3D graphics, communications) while maximizing universality. By relying on HTML5 and WebGL, which are available on all modern browsers, it is possible to support the required technical features without using non-standard plugins (such as Java or Adobe Flash) that are not necessarily deployed everywhere and that, in fact, are not supported in many mobile systems. The concrete implementation that has been described suggests that the architecture meets the established requirements. Also, the tests conducted with students suggest that the described hybrid lab model is interesting and engaging to the users, and that it has educational potential.

In the future, several lines of work remain open. Hybrid labs are relatively new, and their possibilities are still being explored. The possibility of adding features such as multiuser collaboration, gamification, or new input mechanisms to the architecture will be explored. Additionally, once the presence of web sockets is wider, their suitability as a communications technology for interactive labs should be reevaluated. Also, the interest on *smart devices* and MOOLs is growing, so exploring how those paradigms can be incorporated into laboratories that use the proposed model remains an interesting line of work.

An additional benefit of hybrid labs based on this model or on similar ones, which may be explored in the future, is that they have a very complete knowledge of their own state. For instance, in the implementation example, the lab keeps full track of the water level and of the board's output state. Thus, it is relatively easy to keep a full record of user activities, which could be used to augment the learning experience even further through learning analytics, automatic assessment, or even intelligent tutors based on that real-time data.

Acknowledgments

This work has received financial support from the Department of Education, Language policy, and Culture of the Basque Government through a Predoctoral Scholarship granted to Luis Rodriguez-Gil.

References

[1] S. D. Bencomo, "Control learning: Present and future," *Annu. Rev. Control*, vol. 28, no. 1, pp. 115–136, 2004.
[2] M. Auer, A. Pester, D. Ursutiu, and C. Samoila, "Distributed virtual and remote labs in engineering," in *Proc. IEEE Int. Conf. Ind. Technol.*, vol. 2, 2003, pp. 1208–1213.
[3] C. S. Tzafestas, N. Palaiologou, and M. Alifragis, "Virtual and remote robotic laboratory: Comparative experimental evaluation," *IEEE Trans. Educ.*, vol. 49, no. 3, pp. 360–369, Aug. 2006.
[4] T. De Jong, M. C. Linn, and Z. C. Zacharia, "Physical and virtual laboratories in science and engineering education," *Science*, vol. 340, no. 6130, pp. 305–308, 2013.
[5] J. Ma and J. V. Nickerson, "Hands-on, simulated, and remote laboratories: A comparative literature review," *ACM Comput. Surveys*, vol. 38, no. 3, 2006, Art. no. 7.

[6] Z. Nedic, J. Machotka, and A. Nafalski, "Remote laboratories versus virtual and real laboratories," in *Proc. IEEE 33rd Annu. Frontiers Educ.*, vol. 1, 2003, pp. T3E-1–T3E-6.

[7] J. V. Nickerson, J. E. Corter, S. K. Esche, and C. Chassapis, "A model for evaluating the effectiveness of remote engineering laboratories and simulations in education," *Comput. Educ.*, vol. 49, no. 3, pp. 708–725, 2007.

[8] E. E. Toth, B. L. Morrow, and L. R. Ludvico, "Designing blended inquiry learning in a laboratory context: A study of incorporating hands-on and virtual laboratories," *Innovative Higher Educ.*, vol. 33, no. 5, pp. 333–344, 2009.

[9] L. Gomes and S. Bogosyan, "Current trends in remote laboratories," *IEEE Trans. Ind. Electron.*, vol. 56, no. 12, pp. 4744–4756, Dec. 2009.

[10] J. M. Andujar, A. Mejías, and M. A. Marquez, "Augmented reality for the improvement of remote laboratories: An augmented remote laboratory," *IEEE Trans. Educ.*, vol. 54, no. 3, pp. 492–500, Aug. 2011.

[11] A. M. Borrero, "Aportaciones a los laboratorios remotos en los estudios de ingenieriá. interaccioń de elementos virtuales y reales mediante realidad aumentada: El laboratorio remoto aumentado," Ph.D. dissertation, Departamento de Tecnologías de la Información de la, Universidad de Huelva, Huelva, Spain, Dec. 2011.

[12] H. Vargas, J. Sánchez-Moreno, S. Dormido, C. Salzmann, D. Gillet, and F. Esquembre, "Web-enabled remote scientific environments," *Comput. Sci. Eng.*, vol. 11, no. 3, pp. 36–46, 2009.

[13] H. Vargas, J. Sanchez Moreno, C. A. Jara, F. A. Candelas, F. Torres, and S. Dormido, "A network of automatic control web-based laboratories," *IEEE Trans. Learn. Technol.*, vol. 4, no. 3, pp. 197–208, Jul.–Sep. 2011.

[14] N. Duro, et al., "An integrated virtual and remote control lab: The three-tank system as a case study," *Comput. Sci. Eng.*, vol. 10, pp. 50–59, 2008.

[15] J. García-Zubia, et al., "Developing a second-life-based remote lab over the WebLab-Deusto architecture," in *Proc. Remote Eng. Virtual Instrum. Conf.*, 2010, pp. 171–176.

[16] M. Callaghan, K. McCusker, J. L. Losada, J. Harkin, and S. Wilson, "Using game-based learning in virtual worlds to teach electronic and electrical engineering," *IEEE Trans. Ind. Informat.*, vol. 9, no. 1, pp. 575–584, Feb. 2013.

[17] M. Callaghan, K. McCusker, J. L. Losada, J. Harkin, and S. Wilson, "Circuit Warz, the games; collaborative and competitive game-based learning in virtual worlds," in *Proc. IEEE 9th Int. Conf. Remote Eng. Virtual Instrum.*, 2012, pp. 1–4.

[18] M. Callaghan, N. McShane, and A. G. Eguiluz, "Using game analytics to measure student engagement/retention for engineering education," in *Proc. IEEE 11th Int. Conf. Remote Eng. Virtual Instrum.*, 2014, pp. 297–302.

[19] A. Carpeno, S. Lopez, and J. Arriaga, "Using remote laboratory eLab3D for a broader practical skills training in electronics," in *Proc. IEEE 11th Int. Conf. Remote Eng. Virtual Instrum.*, 2014, pp. 98–99.

[20] B. Scheucher, P. H. Bailey, C. Gütl, and J. V. Harward, "Collaborative virtual 3D environment for internet-accessible physics experiments," *Int. J. Online Eng.*, vol. 5, no. S1, pp. 65–71, 2009.

[21] "Html5 specification," *W3, Tech. Rep.*, Jan. 2016. [Online]. Available: https://www.w3.org/TR/html5

[22] "Webgl specification," Khronos WebGL Working Group, *Tech. Rep.*, Oct. 2014. [Online]. Available: https://www.khronos.org/registry/webgl/specs/1.0/

[23] *Unity3d website*. [Online]. Available: https://unity3d.com, Accessed on: Jan. 19, 2016.

[24] J. García-Zubia, P. Orduña, D. López-de Ipiña, and G. R. Alves, "Addressing software impact in the design of remote laboratories," *IEEE Trans. Ind. Electron.*, vol. 56, no. 12, pp. 4757–4767, Dec. 2009.

[25] M. T. Restivo, J. Mendes, A. M. Lopes, C. M. Silva, and F. Chouzal, "A remote laboratory in engineering measurement," *IEEE Trans. Ind. Electron.*, vol. 56, no. 12, pp. 4836–4843, Dec. 2009.

[26] J. García-Zubía, I. Angulo, L. Rodriguez-Gil, P. Orduña, O. Dziabenko, and M. Güenaga, "Boole-WebLab-FPGA: Creating an integrated digital electronics learning workflow through a hybrid laboratory and an educational electronics design tool," *Int. J. Online Eng.*, vol. 9, no. S8, pp. 19–22, 2013.

[27] L. Rodriguez-Gil, P. Orduña, J. García-Zubia, I. Angulo, and D. López-de Ipiña, "Graphic technologies for virtual, remote and hybrid laboratories: WebLab-FPGA hybrid lab," in *Proc. IEEE 11th Int. Conf. Remote Eng. Virtual Instrum.*, 2014, pp. 163–166.

[28] L. Rodriguez-Gil, J. Garca-Zubia, P. Ordua, I. Angulo, and D. Lpez-de-Ipia,"Hybrid laboratory for rapid prototyping in digital electronics," in *Online Experimentation: Emerging Technologies and IoT*, M. Restivo, A. Cardoso, and A. M. Lopes, Eds. Barcelona, Spain: Int. Freq. Sensor Assoc. Publishing, 2015, ch. 9, pp. 179–194.

[29] J. Behr, P. Eschler, Y. Jung, and M. Zöllner, "X3DOM: A DOM-based HTML5/X3D integration model," in *Proc. 14th Int. Conf. 3D Web Technol.*, 2009, pp. 127–135.

[30] S. Lopez, A. Carpeno, and J. Arriaga, "Laboratorio remoto eLab3D: Un mundo virtual inmersivo para el aprendizaje de la electrónica," in *Proc. IEEE 11th Int. Conf. Remote Eng. Virtual Instrum.*, 2014, pp. 100–105.

[31] S. Lopez, A. Carpeno, and J. Arriaga, "Remote laboratory eLab3D: A complementary resource in engineering education," *IEEE Revista Iberoamericana de Tecnologias del Aprendizaje*, vol. 10, no. 3, pp. 160–167, Aug. 2015.

[32] V. J. Harward, et al., "The iLab shared architecture: A web services infrastructure to build communities of internet accessible laboratories," *Proc. IEEE*, vol. 96, no. 6, pp. 931–950, Jun. 2008.

[33] J. Kaplan and N. Yankelovich, "Open wonderland: An extensible virtual world architecture," *IEEE Internet Comput.*, vol. 15, no. 5, pp. 38–45, Sep./Oct. 2011.

[34] F. R. dos Santos, C. Guetl, P. H. Bailey, and V. J. Harward, "Dynamic virtual environment for multiple physics experiments in higher education," in *Proc. IEEE Global Edu. Eng. Conf.*, 2010, pp. 731–736.

[35] M. Callaghan, K. McCusker, J. Lopez Losada, J. Harkin, and S. Wilson, "Engineering education island: Teaching engineering in virtual worlds," *Innovation Teaching Learn. Inf. Comput. Sci.*, vol. 8, no. 3, pp. 2–18, 2009.

[36] Second life website. [Online]. Available: http://secondlife.com, Accessed on: Dec. 18, 2015.

[37] M. N. K. Boulos, L. Hetherington, and S. Wheeler, "Second life: An overview of the potential of 3-D virtual worlds in medical and health education," *Health Inf. Libraries J.*, vol. 24, no. 4, pp. 233–245, 2007.

[38] Sloodle website. [Online]. Available: http://sloodle.org, Accessed on: Dec. 18, 2015.

[39] J. W. Kemp, D. Livingstone, and P. R. Bloomfield, "SLOODLE: Connecting VLE tools with emergent teaching practice in second life," *British J. Edu. Technol.*, vol. 40, no. 3, pp. 551–555, 2009.

[40] K. McCusker, M. Callaghan, J. Harkin, and S. Wilson, "Intelligent assessment and learner personalisation in virtual 3D immersive environments," in *Proc. Eur. Conf. Games Based Learn.*, 2012, Art. no. 591.

[41] *Opensim website*. [Online]. Available: http://opensimulator.org, Accessed on: Dec. 18, 2015.

[42] S. L. Delp, et al., "OpenSim: Open-source software to create and analyze dynamic simulations of movement," *IEEE Trans. Biomed. Eng.*, vol. 54, no. 11, pp. 1940–1950, Nov. 2007.

[43] M. Callaghan and M. Cullen, "Game based learning for teaching electrical and electronic engineering," in *Advances in Computer Science and Its Applications*. Berlin, Germany: Springer, 2014, pp. 655–660.

[44] F. Luthon and B. Larroque, "LaboREM—A remote laboratory for game-like training in electronics," *IEEE Trans. Learn. Technol.*, vol. 8, no. 3, pp. 311–321, Jul.–Sep. 2015.

[45] "McAfee Labs Threats Report," Intel Secur., Santa Clara, CA, USA, *Tech. Rep.*, May 2015.

[46] M. Kaluz, J. Garcia-Zubia, M. Fikar, and L. Cirka, "A flexible and configurable architecture for automatic control remote laboratories," *IEEE Trans. Learn. Technol.*, vol. 8, no. 3, pp. 299–310, Jul.–Sep. 2015.

[47] C. Salzmann and D. Gillet, "Smart device paradigm standardization for online labs," in *Proc. IEEE Global Eng. Educ. Conf.*, 2013, pp. 1217–1221.

[48] C. Salzmann, D. Gillet, and Y. Piguet, ''MOOLs for MOOCs: A first edX scalable implementation,'' in *Proc. IEEE 13th Int. Conf. Remote Eng. Virtual Instrum.*, 2016, pp. 246–251.

[49] W. M. El-Medany, "FPGA remote laboratory for hardware e-learning courses," in *Proc. IEEE Region 8th Int. Conf. Comput. Technol. Elect. Electron. Eng.*, 2008, pp. 106–109.

[50] J. Lobo, "Interactive demonstration of a remote reconfigurable logic laboratory for basic digital design," in *Proc. 1st Experiment@ Int. Conf.—Remote Virtual Labs*, vol. 11, 2011, pp. 262–263.

[51] J. Garcia-Zubia, et al., "Application and user perceptions of using the WebLab-Deusto-PLD in technical education," in *Proc. IEEE Frontiers Educ. Conf.*, 2011, pp. GOLC-1-1–GOLC1-6.

[52] E. Besada-Portas, J. A. Lopez-Orozco, L. De La Torre, and J. M. de la Cruz, "Remote control laboratory using EJS applets and TwinCat programmable logic controllers," *IEEE Trans. Educ.*, vol. 56, no. 2, pp. 156–164, May 2013.

[53] M. Gilibert, et al., "80C537 microcontroller remote lab for e-learning teaching," *Int. J. Online Eng.*, vol. 2, no. 4, pp. 1–3, 2006.

[54] P. Orduña, J. Irurzun, L. Rodriguez-Gil, J. G. Zubía, F. Gazzola, and D. López-de Ipiña, "Adding new features to new and existing remote experiments through their integration in WebLab-Deusto," *Int. J. Online Eng.*, vol. 7, no. S2, pp. 33–39, 2011.

Luis Rodríguez-Gil received the double degree in computer engineering and industrial organization engineering, in 2013, and the MSc degree in information security, in 2014. He is currently working toward the PhD degree in the DeustoTech Internet Group. Since 2009, he has been involved in the WebLab-Deusto Research Group, collaborating in the development of the WebLab-Deusto RLMS. He has published several peer-reviewed publications and contributed to some Open Source projects.

Javier García-Zubia (M'08-SM'11) received the PhD degree in computer sciences from the University of Deusto. He is a full professor in the faculty of engineering, University of Deusto, Spain. His research interests include remote laboratory design, implementation, and evaluation. He is the leader of the WebLab-Deusto research group. He is a senior member of the IEEE.

Pablo Orduña (M'05) received the degree in computer engineering and the PhD degree from the University of Deusto, in 2007 and 2013, respectively. He is a full time researcher and project manager in the MORElab Research Group, DeustoTech Internet. During the PhD degree, he was a visiting researcher twice for 6 weeks each, in the MIT CECI, in 2011, and UNED DIEEC, in 2012. Since 2004, he has also been involved in the WebLabDeusto Research Group, leading the design and development of WebLab-Deusto. He is a member of the IEEE.

Diego López-de-Ipiña received the PhD degree from the University of Cambridge, in 2002. He is an associate professor and P.R. of MORElab Group and director of DeustoTech Internet unit, and of the PhD program within the Faculty of Engineering, University of Deusto. He is responsible for several modules in the BSc and MSc in computer engineering degrees, he is interested in pervasive computing, IoT, semantic service middleware, open linked data, and social data mining. He is taking and has taken part in several big consortium-based research european (IES CITIES, MUGGES, SONOPA, CBDP, GO-LAB, LifeWear) and Spanish projects, and has more than 70 publications in relevant international conferences, and journals, including more than 25 JCR-indexed articles.