

An Extensible Architecture for the Integration of Remote and Virtual Laboratories in Public Learning Tools

Pablo Orduña, *Member, IEEE*, Danilo G. Zutin, *Member, IEEE*, Sten Govaerts, Irene Lequerica, Philip H. Bailey, Elio Sancristobal, *Member, IEEE*, Christophe Salzmann, Luis Rodriguez-Gil, Kimberly DeLong, Denis Gillet, *Senior Member, IEEE*, Manuel Castro, *Fellow, IEEE*, Diego López-de-Ipiña, Javier Garcia-Zubia, *Senior Member, IEEE*

Abstract—Remote laboratories are software and hardware tools that allow students to remotely access real equipment located in universities. The integration of remote laboratories in Learning Tools (Learning Management Systems, Content Management Systems or Personal Learning Environments) has been achieved to integrate remote laboratories as part of the learning curricula. A cross-institutional initiative called gateway4labs has been created to perform this integration extensible to multiple remote laboratories in multiple learning tools. This contribution focuses on describing this initiative and, in particular, how opening it to public systems (where users do not need to be registered) produces new technical and organizational challenges due to the public availability of labs. Additionally, this contribution shows integrations of systems that were not previously addressed in this initiative, such as PhET or ViSH, as well as a new approach for integrating supported laboratories in external specifications such as the Smart Device one through OpenSocial.

Index Terms— Learning Technologies Interoperability, Virtual and Remote Laboratories, Remote Laboratory Federation, Remote Laboratory Management Systems

I. INTRODUCTION

AN Educational Remote Laboratory is a software and hardware solution that enables students to access real equipment located in their institution, as if they were in a

P. Orduña, L. Rodriguez-Gil, D. López-de-Ipiña and J. García-Zubia are with DeustoTech, University of Deusto, Bilbao, Spain (e-mail: {pablo.orduna,luis.rodriguezgil,dipina,zubia}@deusto.es)

D. G. Zutin is with the Carinthian University of Applied Sciences, Villach, Austria (e-mail: D.GarbiZutin@fh-kaernten.at)

S. Govaerts, C. Salzmann, D. Gillet are with École Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland (e-mail: {sten.govaerts,christophe.salzmann,denis.gillet}@epfl.ch)

I. Lequerica, E. Sancristobal, M. Castro are with Spanish University for Distance Education (UNED), Madrid, Spain (e-mail: {ilequerica,elio,mcastro}@ieec.uned.es)

P.H.Bailey, K. DeLong are with the Massachusetts Institute of Technology (MIT), Cambridge, MA, USA (e-mail: {pbailey,kirky}@mit.edu)

DOI (Digital Object Identifier) Pendiente

hands-on-lab session. There are many kinds of remote laboratories, even in fields such as Physics, Electronics, Robotics or Chemistry.

Once these laboratories are available through the Internet, it becomes possible to share them with other universities. Given that both traditional and remote laboratories spend a long time unused (nights, weekends, weeks with no practical lessons...), if University A shares two laboratories with University B and vice versa, then both have 4 labs with similar maintenance costs. For this reason, remote laboratory federations have been explored and used in production for years.

Integrations of remote laboratories in Learning Tools –in the rest of the document, LT–, such as Content Management Systems (CMSs) or Learning Management Systems (LMSs) have been addressed in the literature. The focus is to integrate remote laboratories (which are educational resources after all) into LTs (which is where the students and teachers meet and where students find other educational resources). While this integration has always been a relevant topic in the area of remote laboratories, most solutions in the literature have been ad-hoc: a remote laboratory developed in University A is integrated in the LMS used in University A and will not work with other LTs.

The usage of remote laboratory federations for the integration of remote laboratories in LTs through a centralized Open Source component called gateway4labs was proposed and evaluated in [1]. In that contribution, that component made it possible to support different LMSs through the use of IMS LTI, as well as other systems (such as LMSs not supporting LTI, or CMSs) with plug-ins. This component also supported a set of plug-ins for integrating new remote laboratory management systems, so it was extensible in both sides. It was evaluated by testing it in Moodle LMS (through IMS LTI), the .LRN LMS (without IMS LTI) and the Joomla CMS.

Additionally, the Go-Lab project¹ aims to provide remote and virtual labs [2] for the large-scale use in education [3], [4]. In the scope of this project, existing laboratories will be

¹ <http://www.go-lab-project.eu>

provided to schools free of charge, saving teachers the bother of registration processes, and through a technical solution based on OpenSocial.

The focus of this contribution, which is an extended version of [5], is that gateway4labs is a valid tool for supporting different LTs, including Personal Learning Environments based on OpenSocial in those remote laboratories that have been integrated for their support in other LTs. This provides organizational and technical challenges. Originally, gateway4labs was designed for supporting remote laboratories under the assumption that these remote laboratories were closed, requiring certain authentication mechanisms. So, they had to be registered in gateway4labs one by one, and it was important to provide them feedback on who was using the system. However, the context of Go-Lab –where all laboratories must be publicly available– is an important challenge for this design:

- It requires that certain systems managing multiple laboratories export only a subset of these laboratories.
- It requires that open systems with hundreds of public laboratories such as PhET or ViSH are exported automatically, with no administrator selecting a subset.
- It requires that systems, which are traditionally closed, can still be used under this approach transparently, wrapping the authentication and authorization procedure.
- It requires that those remote laboratory providers who are not entirely convinced of sharing the laboratory can still feel comfortable by being able to use the OpenSocial protocol and still control that it is only used in certain spaces. While this is discouraged by the authors, it could help laboratory owners to join the initiative and in a second phase agree on opening the laboratory.

The paper is structured as follows: Section II introduces the concepts of remote laboratory, remote laboratory management system and remote laboratory federation. Then, Section III shows what integrations of remote laboratories in LTs are available in the literature. Section IV explains the integration on OpenSocial and the challenges that arise when dealing with public or private laboratories. Section V shows the integration of particular laboratories in this environment. Finally, Section VII describes the conclusions and the future work.

II. REMOTE LABORATORIES FEDERATIONS

This section introduces the concepts of remote laboratories, Remote Laboratory Management Systems (RLMS) and remote laboratory federations.

A. Remote Laboratories

A remote laboratory is a software and hardware tool that allows students to remotely access real equipment located in the university. Users access this equipment as if they were in a traditional hands-on-lab session, but through the Internet. To show a clear example, Figure 1 shows a mobile low-cost robot laboratory described in [6]. Students learn to program a Microchip PIC microcontroller, and they write the code at home, compile it with the proper tools, and then submit the

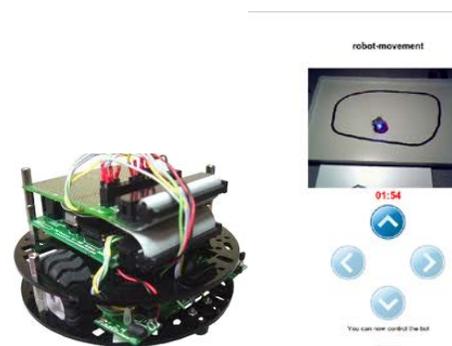


Fig. 1. Robot laboratory [6]. At the left, the mobile robot itself. At the right, the user interface once the program has been submitted.

binary file to a real robot through the Internet. Then, students can see how the robot performs with their program through the Internet (e.g., if it follows the black line according to the submitted program, etc.) in a real environment.

In this line, there are many examples and classifications in the literature [7], [8]. Indeed, remote laboratories were born nearly two decades ago [9], [10], [11], and since then they have been adopted in multiple fields: Chemistry [12], [13], Physics [14], [15], Electronics [16], [17], Robotics [18], [19] and even in nuclear reactors [20].

B. Remote Laboratory Management Systems

Every remote laboratory manages at least a subset of the following features: authentication, authorization, scheduling users to ensure exclusive accesses –typically through a queue or calendar-based booking–, user tracking and administration tools. These features are common to most remote laboratories, and are actually independent of the particular remote laboratory settings. For example, an authentication and queuing system is valid both for an electronics laboratory and for a chemistry laboratory.

For this reason, Remote Laboratory Management Systems (RLMSs) arose. These systems (e.g., MIT iLabs², WebLab-Deusto³ or Labshare Sahara⁴) provide development toolkits for developing new remote laboratories, as well as management tools and common services (authentication, authorization, scheduling mechanisms). The main idea is that by adding a new feature to one of them (e.g., supporting LDAP, or LMSs), all the laboratories which are developed on top of them will support this feature automatically.

C. Federating Remote Laboratories

One of the features that RLMSs started supporting was federating their remote laboratories. For example, if two universities (University A and University B) install a particular RLMS, they support federation protocols so University A shares a laboratory with students of University B without knowing these students. The key here is that the provider university does not need to register particular students, but

² <http://ilab.mit.edu>

³ <http://weblab.deusto.es>

rather groups or simply universities. It is the consumer system that defines that a set of local users can access a particular laboratory of the provider system.

Therefore, the relationship between two federated entities is the following:

- The consumer system manages the authentication and authorization of its students.
- The provider system manages the scheduling and the access to the laboratories, storing what the users did.
- The consumer system will later ask for results to the provider system.
- At every moment, the provider system does not need to know anything related to the particular students.

These federation protocols have been used for fostering interoperability between RLMS [21]. These interoperable bridges between different systems can be enhanced if properties such as transitivity or federated load balance are provided [22].

III. EXISTING INTEGRATION OF REMOTE LABORATORIES IN LEARNING TOOLS

The relevance of this field is that, as detailed in [23], [24], there are several services duplicated between remote laboratories and learning management systems. The administration and user experience would increase if they were merged. Both systems usually support user authentication, authorization, group management, administrative tools, user tracking, and even scheduling. Some integration approaches suggest to delegate all these services to the LMS, but some of these services will still be at least shared, such as scheduling (especially when federation systems arise) or user tracking (since some interactions with the remote laboratory might occur outside the scope of the web browser).

There are two types of solutions in the literature for using remote laboratories in LTs: ad-hoc, or based on federation protocols.

A. Ad-hoc Solutions

In order to integrate remote laboratories and LMSs, [7] discusses the usage of SCORM and [25] implements an architecture around it. This technology is designed to be supported by different LMSs and indeed multiple LMSs have implemented different versions. However, since it is a client-side technology and therefore it cannot contain any server code, it does not support a secure way to exchange credentials, ensure reservations or return results to the LMS.

Another approach is to develop an ad-hoc plug-in to include a particular remote laboratory on a LMS, applying the required changes in the remote laboratory [26], [27]. This approach is common in the literature, and sometimes it is implemented by just copying or exchanging the users among both systems. Within the field of integrating remote laboratories on electronic tools appears the integration of remote laboratories

on CMSs.

In [28], the remote laboratory relies on Joomla to perform all the administrative tasks. This approach is interesting since it does not duplicate all the tasks mentioned in [23]. However, it is an example of an ad-hoc integration that neither supports the integration of other remote laboratories nor the integration on other CMSs.

B. Federated Solutions

In [29], it was proposed another approach to integrate remote laboratories in CMSs and LMSs (C/LMS), relying on the existing federation protocols already existing in certain RLMSs. Federation protocols have the same targets as the integration protocols of remote laboratories in C/LMSs, where the consumer system is the C/LMS. Indeed, if in the description of federations presented in Section II-C the term consumer system was replaced by C/LMS, the following description would be presented, which applies very well to the definition of integration of remote laboratories in C/LMSs:

- The C/LMS manages the authentication and authorization of its students.
- The provider system manages the scheduling and the access to the laboratories, storing what the users did.
- The C/LMS will later ask for results to the provider system.
- At every moment, the provider system does not need to know anything related to the particular students.

C. Gateway4labs

On top of this concept, [1] showed an implementation of an Open Source middleware, called gateway4labs⁵, which provides a centralized component for this type of integrations. This component supports multiple laboratories through a plug-in system, where the plug-in wraps the federation protocol or the connection to the remote laboratory. Back then, it also had two systems for being integrated in different CMSs and LMSs, through two main systems: IMS LTI (so it works in Moodle since 2.3, as well as other major LMSs) and a custom RESTful interface (and a Joomla plug-in implementing it was presented).

IV. OPENING REMOTE LABORATORIES

As previously stated, within the Go-Lab project, the target is to provide laboratories at zero cost for school students. So as to make the process as simple as possible for teachers, no registration process will be required [30], and all the labs from different institutions will be available at zero cost. Teachers will not access the labs directly, but through the Go-Lab portal⁶ [31] that provides a global registry of supported labs. They will create environments that later students will use in class, using Inquiry Learning Spaces [30] through a social platform developed at the EPFL called Graasp⁷ [32]. This

⁴ <http://labshare-sahara.sf.net>

⁵ <https://github.com/gateway4labs/>

⁶ <http://www.golabz.eu>

⁷ <http://graasp.eu>

platform uses an industry standard called OpenSocial⁸ for registering and developing applications.

So as to integrate existing laboratories without reimplementing them in OpenSocial, it was considered that gateway4labs could act as a glue system by supporting OpenSocial as a new protocol on its middle component. This way, if a remote laboratory owner attempts to integrate its laboratory in Go-Lab, it can be done by adding a new plug-in to gateway4labs, and since gateway4labs already wraps it in OpenSocial, the laboratory is available for Go-Lab. However, more importantly, a key feature of this approach is that if remote laboratory owners willing to support their laboratories in other system (such as in Joomla or any LMS supporting IMS LTI such as Moodle), they might still be willing to develop the same plug-in, and if they later want to make the laboratory available in Go-Lab, no technical work is required.

However, the fact that it had to provide public access to the laboratories (and therefore, the gateway4labs administrator must select which laboratories are public and which ones are not) added some organizational and technical challenges. This section describes them.

A. Architecture

The architecture of gateway4labs is described in Figure 2. As described in [1], gateway4labs has two parts: the left part is called Client side, which represents the final system used by the user (e.g., a LMS, a CMS, or, as it is novel in this contribution, the social platform compatible with OpenSocial). The right side is called Lab side, which represents the laboratory management system (whether it is a RLMS itself – such as WebLab-Deusto or the iLab Shared Architecture– or a laboratory itself –such as the UNR FCEIA one–). In the middle, there is a central component called LabManager, which provides the plug-in system and supports the protocols used by the Learning Tools (e.g., the custom RESTful interface, or IMS LTI or OpenSocial in this case).

It is important to remark that this process (user contacting the Learning Tool which contacts the LabManager, which contacts the RLMS which contacts the laboratory) is not taken in one step and does not affect latency or network efficiency. The whole process is only executed for the handshake when performing the reservation, but once the reservation is provided, the user is contacting the final laboratory through an HTML iframe (a web component that opens a web page inside a part of other web page). For example, if the user uses an aquarium laboratory in WebLab-Deusto, the user will contact the LabManager, which will contact WebLab-Deusto providing the LabManager credentials. WebLab-Deusto will provide a URL which includes a secure token and the student will be redirected to that URL in the iframe. The student cannot perform new reservations (a new call to the LabManager would be required), and the connection to the laboratory is direct during that reservation.

⁸ <http://opensocial.org>

B. OpenSocial integration

OpenSocial is a wide and complete standard that includes authorization, data storage or messaging mechanisms. However, since the target laboratories in gateway4labs are developed by third parties which might not have embraced the OpenSocial standard, many of these mechanisms cannot be used.

In this sense, the target was to develop an OpenSocial application that could wrap any laboratory automatically and still maintain the requirements of gateway4labs in terms of maintaining a federation and security. This way, for any registered laboratory in the LabManager, an OpenSocial app is automatically generated. This app uses the authorization mechanism to retrieve who the user is, provided he/she is logged in. The LabManager obtains this information in a secure way through a security token generated by the OpenSocial platform, and this information is provided to the final RLMS. Typically, this might be ignored by the particular laboratory. However, in certain cases it is possible to develop Learning Analytics customized for the remote laboratory domain, as done in [33]. While the Go-Lab portal [31] already provides Learning Analytics tools [34], and most lab owners will rely on them to know what users were doing, it is still very useful for remote laboratory owners to create its Learning Analytics fully customized for the particular laboratory and independently of external systems.

Additionally, the integration obtains the current space where the student is (e.g., a robotics-dedicated space added and customized by the teacher). Typically these spaces are public, but the target OpenSocial platform (Graasp) enables users to close or hide spaces so only certain users can access them. Obtaining the current space and the parent spaces is useful to provide the LabManager with more data in case a particular remote lab owner wants to customize the access to a particular school, as explained in the following subsection.

C. Tradeoff between publishing laboratories or keeping them private

In the context of Go-Lab, remote laboratories are publicly available, with no registration. This makes it easier to get teachers and students involved, since no registration process is required.

However, some initiatives consider that certain remote laboratories might require some associated direct or indirect costs for sustainability purposes. [35] provides a cost model for remote laboratories, and Labicom [36] actually charges small amounts of money for its use. Freemium models have also been proposed (where the access is free but certain features are provided for registered users), and in certain circumstances the laboratories can be free but still require detailed information about its use so as to find other external funding mechanisms.

For this reason, gateway4labs also implemented an OpenSocial integration that does not require a laboratory to be completely public. In this sense, teachers can optionally register their school, they can request access to laboratories,

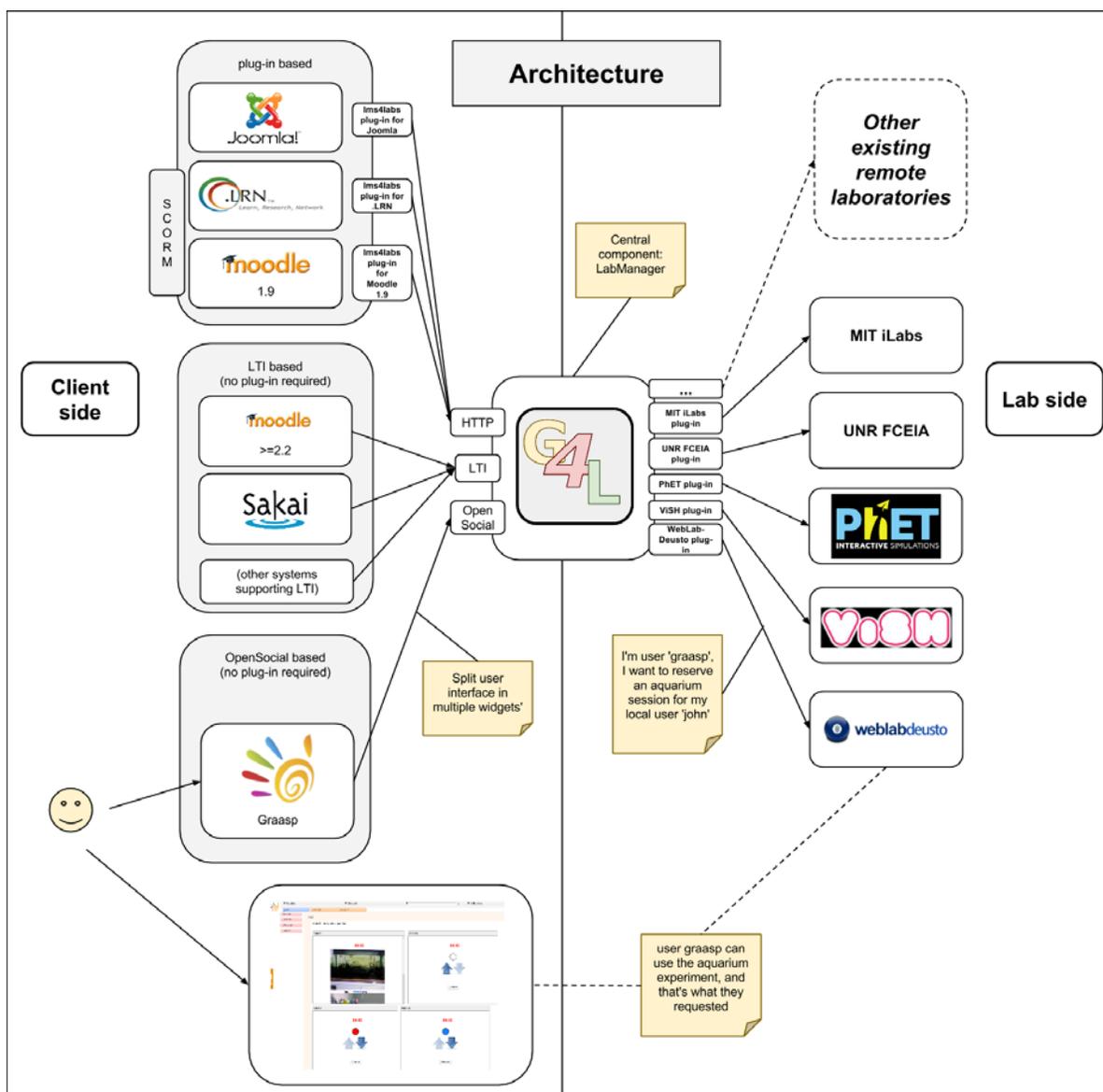


Fig. 2. Gateway4labs architecture

and if granted, they can select in which closed spaces they want to use these laboratories. This mechanism, which is far more complex for the teacher, maintains this degree of security and supports remote laboratory owners who want to provide access to certain features to particular schools. A tradeoff between complexity for the user and sustainability must be decided by the remote laboratory owner, while gateway4labs provides mechanisms for either decision. While this feature is outside the scope of the Go-Lab project, it is implemented to foster more remote lab owners to integrate their labs in gateway4labs and therefore potentially in Go-Lab in the future.

D. Public and private laboratory systems

In the Lab side of Figure 2, two types of systems are presented:

- Remote laboratories requiring authentication, such as WebLab-Deusto, UNR FCEIA or iLabs.

- Public systems, such as PhET⁹ or ViSH¹⁰, where no authentication is required. PhET provides simulations (not remote laboratories), and ViSH provides a set of educational resources among which simulations and remote laboratories are present.

For the first type, the plug-in supporting the laboratory will act as a bridge for bypassing the authentication mechanism used by the remote laboratory. Since these systems manage more than one laboratory, gateway4labs retrieves the list of provided laboratories with certain credentials, and within this list, the gateway4labs administrator will select which laboratories will be public and which ones will not. This way all the laboratories are private by default. Certain operations, such as searching among those laboratories in the list, are not public.

However, this approach does not fit the second type. Both

⁹ <https://phet.colorado.edu>

¹⁰ <http://vishub.org>

Selected	Identifier	Name	Widgets
<input type="checkbox"/>	http://phet.colorado.edu/en/simulation/acid-base-solutions	Acid-Base Solutions	list
<input type="checkbox"/>	http://phet.colorado.edu/en/simulation/alpha-decay	Alpha Decay	list
<input type="checkbox"/>	http://phet.colorado.edu/en/simulation/area-builder	Area Builder	list
<input type="checkbox"/>	http://phet.colorado.edu/en/simulation/arithmetic	Arithmetic	list
<input type="checkbox"/>	http://phet.colorado.edu/en/simulation/atomic-interactions	Atomic Interactions	list

Fig. 3. Public systems: PhET scenario

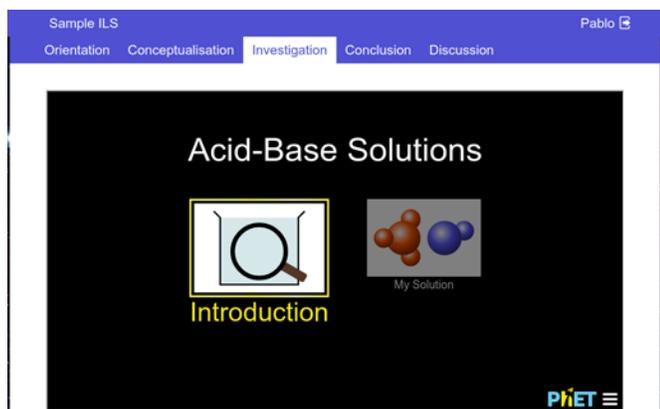


Fig. 4. PhET laboratory running inside a Go-Lab environment

PhET and ViSH[37], [38] have hundreds of resources which are already public. For this reason, gateway4labs¹¹ supports that a plug-in is marked as "public", and therefore all the resources provided by it are automatically public and operations such as searching for these resources are public too. No contact with the administrator is required if PhET or ViSH support new resources: these resources will automatically be available through gateway4labs and therefore it will be possible to use them in Go-Lab. Figure 3 shows the public listing of PhET labs through gateway4labs, and Figure 4 shows how a PhET laboratory is loaded in the Go-Lab context. Figure 5 shows a ViSH excursion added to Go-Lab through gateway4labs.

V. INTEGRATING REMOTE LABORATORIES IN GATEWAY4LABS

This section describes the laboratories integrated as part of this contribution in gateway4labs.

A. WebLab-Deusto integration

A plug-in for WebLab-Deusto¹² has been developed. This means that any remote laboratory developed in WebLab-Deusto can automatically be used from Graasp (including not only the laboratories developed in the University of Deusto¹³ but also in other universities using WebLab-Deusto, such as

¹¹ <http://gateway.golabz.eu/public/>

¹² https://github.com/gateway4labs/rfms_weblabdeusto

¹³ <https://weblab.deusto.es/weblab/>

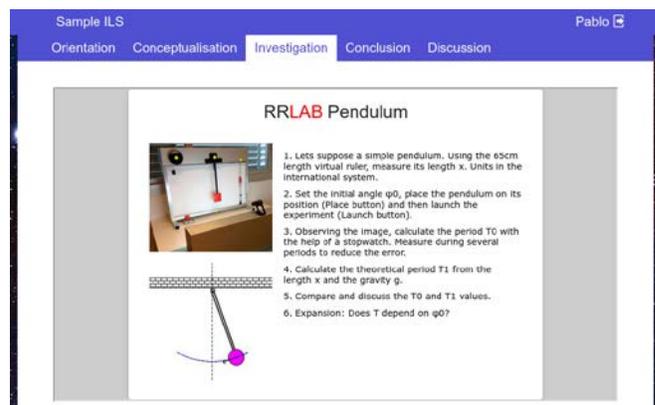


Fig. 5. ViSH excursion displayed in Go-Lab

STU¹⁴, PUC Sao Paulo¹⁵ or BSU¹⁶ among others). However, it had to be modified to support the new features for OpenSocial, in particular, splitting the user interface in multiple widgets. In Figure 6 an aquarium with the user interface splitted is shown.

The total code required for this integration was 925 lines of Python code, out of which 693 were an existing library for accessing WebLab-Deusto from Python. The remaining 232 lines of code are not relevant compared with the over 7000 lines of code provided by the LabManager.

B. iLab Shared Architecture integration

The iLab Shared Architecture [39] was integrated in gateway4labs¹⁷. This architecture is splitted in two different architectures: the batch architecture and the interactive architecture. The former is fully supported in gateway4labs. The latter, which is typically used with a calendar-based booking mechanism, is only supported if this mechanism is removed. Figure 7 shows a laboratory running in the Carinthian University of Applied Sciences (CUAS) in Austria inside Graasp through gateway4labs.

Any other laboratory managed by iLab can be integrated. For example, in the Go-Lab project, the radioactivity laboratory from the University of Queensland [40] is used, as seen on Figure 8. In this case, Go-Lab uses the gateway4labs component to contact the iLab Service Broker located in Austria, which uses the standard iLab federation system to consume the laboratory deployed in the University of Queensland. This way, gateway4labs acts as an actor of iLab, which internally uses its own federation mechanism to reach laboratories from other institutions. In any case, none of these systems needed to register the particular Go-Lab user, since this is transparent through gateway4labs.

The total code required for supporting iLab was 211 lines of Python code. In addition, few changes were required in the iLab architecture itself to support the creation of users automatically for supporting this federation mechanism.

¹⁴ <http://weblab.chtf.stuba.sk>

¹⁵ <http://weblabduino.pucsp.br/weblab/>

¹⁶ <http://weblab.bsu.ge>

¹⁷ https://github.com/gateway4labs/rfms_ilabs

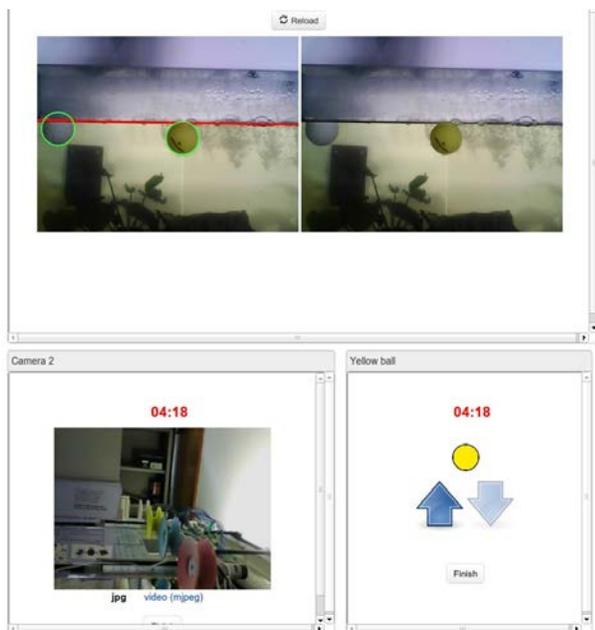


Fig. 6. Aquarium laboratory at Deusto running in Graasp

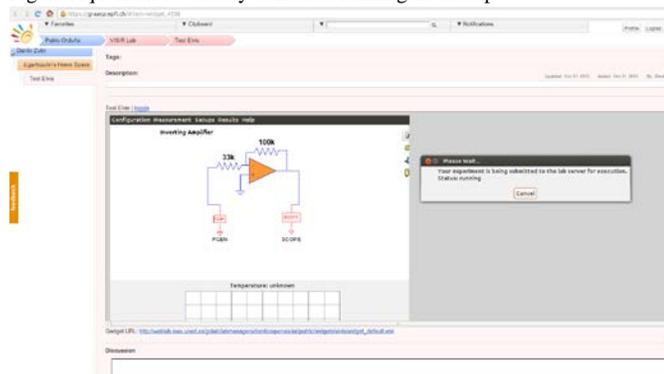


Fig. 7. Elvis laboratory at CUAS running in Graasp

C. Other laboratories

The gateway4labs RLMS API is public¹⁸ and can be adapted for other remote laboratories. So as to proof this concept, in [1] the UNR FCEIA laboratory [41] was integrated. This laboratory was selected since it does not use any RLMS, so it represents an example of standalone laboratory. The integration¹⁹ required a total of 141 lines of Python code. The PhET integration²⁰ explained in the previous section required 142 lines of Python code, and the ViSH integration²¹, 118 lines of Python code.

With this small amount of code, all these laboratories are automatically supported by Go-Lab through Graasp, as well as by the LMSs supported by gateway4labs.

VI. FULL INTEGRATION OF LABORATORIES IN THE SMART GATEWAY

In the context of Go-Lab, new laboratories are using the

¹⁸ <http://gateway4labs.readthedocs.org>

¹⁹ https://github.com/gateway4labs/rlms_unr

²⁰ https://github.com/gateway4labs/rlms_phet

²¹ https://github.com/gateway4labs/rlms_vish

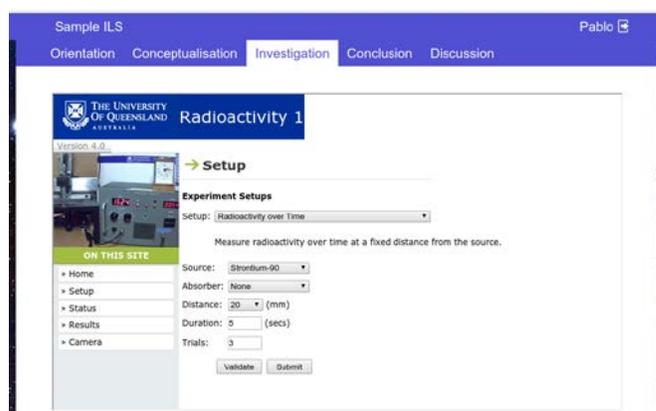


Fig. 8. Radioactivity laboratory at UQ running in Go-Lab

Smart Device paradigm [42]. By adopting this paradigm, remote laboratory developers would not develop remote laboratories as a monolithic piece, where you have a client (user interface) which is connected to a server software, connected to a certain equipment. Instead, they would use a modular approach, where devices focus on providing a set of services using a set of specifications [43], and external, third party clients can match and support those specifications. This way, the reuse of components is encouraged. For example, two different cameras could support the same specification by providing the stream through a particular protocol and different clients could support it. Additionally, a specification defining a triggered action could be matched by different client types (e.g., different types of buttons). Using Smart Devices, teachers can literally play with all the components, customizing the laboratory experience, showing only what they want to show and with the components they want to show to the students.

However, existing laboratories are not implemented using this paradigm. So two levels of integration of remote laboratories were defined in Go-Lab:

- Lightweight integration (gateway4labs), where users could use external laboratories with their original user interfaces, while not fully embracing the Smart Device paradigm.
- Full integration, where laboratories are integrated as if they were Smart Devices, using the same protocols.

The lightweight integration is developing a gateway4labs plug-in, and it achieves that users use the laboratories as designed by the laboratory owners. However, no interoperability with components from other laboratories can be achieved. And teachers are very limited in customizing the user interface that will be presented to the students. On the other hand, the full integration requires a considerable amount of work, since it requires the developer to export all the functionalities through certain services, and usually rewriting the user interface. Therefore, the remote laboratory developer must select one of the two approaches.

So as to demonstrate how a remote laboratory could support the full integration, the concept of protocol translator was defined. A protocol translator is a software component that

acts as a proxy between the remote laboratory and the protocol specified by the Smart Device paradigm. So essentially, gateway4labs still manages the authentication, authorization and scheduling, but once a reservation has been created, it relies on a protocol translator to offer interoperability to the rest of the Go-Lab ecosystem. In the future, other parts of Go-Lab will use the protocol translator to interact with the laboratory in a standardized way. For example, other Go-Lab applications and services could directly contact the laboratory, obtain sensor information and store it somewhere else or make operations on top of this information without the remote laboratory developer being involved.

The main problem of the protocol translator is that it requires a relevant amount of time, since it cannot be reused among remote laboratories easily. For example, once you develop a plug-in for WebLab-Deusto for gateway4labs, all remote laboratories of WebLab-Deusto will work in Graasp automatically. However, there cannot be a protocol translator for WebLab-Deusto, but a protocol translator for each remote laboratory. Therefore, if there are 15 remote laboratories developed using WebLab-Deusto, 15 implementations would be required.

At the time of this writing, two protocol translators are available. Since other Go-Lab layers are missing, they are still not functional, but they show how remote laboratory developers aiming to have a full integration in the Go-Lab ecosystem can implement it.

A. Archimedes laboratory protocol translator

A protocol translator for a WebLab-Deusto remote laboratory has been developed²². Given an existing reservation, it stores the reservation in a Redis server, and a background task is continuously obtaining new values for that reservation and storing them back to Redis. In a different process, a WebSocket server is listening and if a client is connected, it will start receiving a flow of the information stored in Redis when stored, through a pubsub event. This way, regardless the number of clients connected to the protocol translator, a single request every few seconds will be performed to the real laboratory. However, at the time of this writing, the protocol translator only acts as a sensor data provider, since it does not support clients to push information to the laboratory. This protocol translator is implemented in 420 lines of Python code, and it does not cover the new client that should be implemented (since certain parts could be reused from other laboratory) or the reservation process (since it relies on gateway4labs).

B. Radioactivity lab protocol translator

A protocol translator for a radioactivity laboratory has been developed²³. This protocol translator has been developed in PHP, and it supports bidirectional communication using WebSockets with an MIT iLabs batched laboratory. The protocol translator hides the complexities of the web services

used by the iLab Shared Architecture, and whenever the client requests the protocol translator to start pushing information through a WebSocket, it will start a loop where it submits new tasks to the batched laboratory and, whenever there is a response, it is converted to the JSON format required by the Smart Device specification. The client perspective does not see any difference from a regular Smart Device, since it is obtaining real information from the remote laboratory in a loop pushed by the protocol translator. This protocol translator is implemented in 1060 lines of PHP code and, as in the previous case, it does not cover the new client if necessary or the reservation process since it relies on gateway4labs for that task.

VII. CONCLUSIONS AND FUTURE WORK

This contribution is based on the assumptions that it is possible to integrate remote laboratories in Learning Tools using federation protocols [29], and that a middle component can help decoupling Learning Tools from particular remote laboratories [1].

On top of these assumptions and on the existing Open Source implementation (gateway4labs), this contribution shows that this implementation can be extended to the generic integration of remote laboratories in Personal Learning Environments such as Graasp, supporting a wide variety of scenarios. Given that it is a central component, remote laboratory owners willing to be supported in any of the Learning Tools (e.g., Graasp, or a particular LMS such as Moodle, or a particular CMS such as Joomla) will automatically be supported by the rest. This way, if a remote laboratory owner wants to make his or her laboratory available in Moodle using gateway4labs and develops a plug-in for gateway4labs aiming that, its inclusion in Graasp is an organizational issue (whether the institution wants to share the laboratory in Graasp or not), rather than a technical issue (no new code is needed).

Furthermore, this central component has been modified to support public laboratories to foster wider user communities. There is an organizational tradeoff between making the laboratory completely public with no registration required (and easy to use) and requiring a registration (for whatever purpose: charging costs, auditing, fostering Learning Analytics, or providing more features to a particular educational institution). Gateway4labs solves this by providing by default an easy way to establish that a laboratory is public but at the same time providing the chance to lab owners to establish that a laboratory can only be accessed from a particular space of Graasp (which, if it is closed, only certain people will be able to enter, and this is decided by the teacher).

So as to demonstrate it, these changes have been applied, as well as the support of the iLab Shared Architecture (in its batch form and in its interactive form without scheduling) has been added. This way, two RLMS (WebLab-Deusto and iLab Shared Architecture) are supported, as well as a remote laboratory (UNR FCEIA) and two public systems (PhET and

²² https://github.com/gateway4labs/protocol_translator_weblabdeusto

²³ https://github.com/gateway4labs/protocol_translator_radioactivityLab

ViSH). Examples in Graasp have been presented, and their addresses and source code are public.

Regarding future work, ongoing efforts are focused on including more remote laboratories, both using the lightweight approach and supporting the full Smart Device specification.

ACKNOWLEDGMENTS

This contribution is partially funded by the European Union in the context of the Go-Lab project (Grant Agreement no. 317601) under the Information and Communication Technologies (ICT) theme of the 7th Framework Programme for R&D (FP7). This document does not represent the opinion of the European Union, and the European Union is not responsible for any use that might be made of its content.

REFERENCES

- [1] P. Orduña, S. Botero Uribe, N. Hock Isaza, E. Sancristobal, M. Emaldi, A. Pesquera Martin, K. DeLong, P. Bailey, D. López-de Ipiña, M. Castro, and J. García-Zubia, "Generic integration of remote laboratories in learning and content management systems through federation protocols," in 2013 IEEE Frontiers in Education Conference, Oklahoma City, OK, USA, Oct. 2013, pp. 1372–1378.
- [2] 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.
- [3] D. Gillet, T. de Jong, S. Sotirou, and C. Salzmman, "Personalised learning spaces and federated online labs for stem education at school," in Global Engineering Education Conference (EDUCON), 2013 IEEE. IEEE, 2013, pp. 769–773.
- [4] T. de Jong, S. Sotiriou, and D. Gillet, "Innovations in stem education: the go-lab federation of online labs," *Smart Learning Environments*, vol. 1, no. 1, pp. 1–16, 2014.
- [5] P. Orduña, A. Caminero, I. Lequerica, D. Garbi-Zutin, P. Bailey, E. Sancristobal, L. Rodriguez-Gil, A. Robles-Gomez, M. Latorre, K. DeLong, L. Tobarra, S. Ros, M. Castro, D. L. de Ipiña, and J. Garcia-Zubia, "Generic integration of remote laboratories in public learning tools: organizational and technical challenges," 2014 Frontiers in Education Conference Proceedings, 2014.
- [6] O. Dziabenko, J. García-Zubia, and I. Angulo, "Time to play with a microcontroller managed mobile bot," in Global Engineering Education Conference (EDUCON), 2012 IEEE. IEEE, 2012, pp. 1–5.
- [7] L. Gomes and S. Bogosyan, "Current trends in remote laboratories," *Industrial Electronics, IEEE Transactions on*, vol. 56, no. 12, pp. 4744–4756, 2009.
- [8] C. Gravier, J. Fayolle, B. Bayard, M. Ates, and J. Lardon, "State of the art about remote laboratories paradigms-foundations of ongoing mutations," *iJOE*, vol. 4, no. 1, 2008.
- [9] B. Carisa, A. Burain, S. Molly H, and C. Lawrence, "Running control engineering experiments over the internet," 1995.
- [10] B. Aktan, C. Bohus, L. Crowl, and M. Shor, "Distance learning applied to control engineering laboratories," *Education, IEEE Transactions on*, vol. 39, no. 3, pp. 320–326, 1996.
- [11] J. Henry, "Running laboratory experiments via the world wide web," in ASEE Annual Conference, 1996.
- [12] A. Coble, A. Smallbone, A. Bhave, R. Watson, A. Braumann, and M. Kraft, "Delivering authentic experiences for engineering students and professionals through e-labs," in Education Engineering (EDUCON), 2010 IEEE. IEEE, 2010, pp. 1085–1090.
- [13] R. Cedazo, F. Sanchez, J. Sebastian, A. Martínez, A. Pinazo, B. Barros, and T. Read, "Ciclope chemical: a remote laboratory to control a spectrograph," *Advances in Control Education—ACE*, vol. 6, 2006.
- [14] J. Del Alamo, L. Brooks, C. McLean, J. Hardison, G. Mishuris, V. Chang, and L. Hui, "The mit microelectronics weblab: A web-enabled remote laboratory for microelectronic device characterization," in World Congress on Networked Learning in a Global Environment, Berlin (Germany), 2002.
- [15] D. Gillet, H. Latchman, C. Salzmman, and O. Crisalle, "Hands-on laboratory experiments in flexible and distance learning," *Journal of Engineering Education*, vol. 90, no. 2, pp. 187–191, 2001.
- [16] I. Gustavsson et al. "The visir project—an open source software initiative for distributed online laboratories," in Proceedings of the REV 2007 Conference, Porto, Portugal, 2007.
- [17] Z. Nedic, J. Machotka, and A. Nafalski, "Remote laboratory netlab for effective interaction with real equipment over the internet," in Human System Interactions, 2008 Conference on. IEEE, 2008, pp. 846–851.
- [18] R. Safaric et al. "Control and robotics remote laboratory for engineering education," *International Journal of Online Engineering (iJOE)*, vol. 1, no. 1, 2005.
- [19] F. Torres, F. Candelas, S. Puente, J. Pomares, P. Gil, and F. Ortiz, "Experiences with virtual environment and remote laboratory for teaching and learning robotics at the university of alicante," *International Journal of Engineering Education*, vol. 22, no. 4, pp. 766–776, 2006.
- [20] J. Hardison, K. DeLong, P. Bailey, and V. Harward, "Deploying interactive remote labs using the ilab shared architecture," in Frontiers in Education Conference, 2008. FIE 2008. 38th Annual. IEEE, 2008, pp. S2A–1.
- [21] P. Orduña, P. Bailey, K. DeLong, D. López-de Ipiña, and J. García-Zubia, "Towards federated interoperable bridges for sharing educational remote laboratories," *Computers in Human Behavior*, vol. 30, pp. 389–395, Jan. 2014.
- [22] P. Orduña, "Transitive and scalable federation model for remote laboratories," Ph.D. dissertation, Universidad de Deusto, Bilbao, Spain, May 2013. Available: <http://paginaspersonales.deusto.es/porduna/phd/>
- [23] E. Sancristobal, M. Castro, J. Harward et al., "Integration view of web labs and learning management systems," in Education Engineering (EDUCON), 2010 IEEE. IEEE, 2010, pp. 1409–1417.
- [24] E. Sancristobal Ruiz, A. Pesquera Martin, P. Orduna, S. Martin, R. Gil, E. Ruiz Larrocha, M. Albert, G. Diaz, R. Meier, and M. Castro, "Virtual and remote industrial laboratory: Integration in learning management systems," *Industrial Electronics Magazine, IEEE*, vol. 8, no. 4, pp. 45–58, 2014.
- [25] T. Richter, D. Boehringer, and S. Jeschke, "Lila: A european project on networked experiments," *Automation, Communication and Cybernetics in Science and Engineering 2009/2010*, pp. 307–317, 2011.
- [26] J. Ferreira, A. Cardoso et al., "A moodle extension to book online labs," *International Journal of Online Engineering*, vol. 1, no. 2, 2005.
- [27] F. Lerro, S. Marchisio, S. Martini, H. Massaccesi, E. Perretta, A. Gimenez, N. Aimetti, and J. I. Oshiro, "Performing real experiments from a remote learning management system," *Tecnologias del Aprendizaje, IEEE Revista Iberoamericana de*, vol. 9, no. 1, pp. 23–27, 2014.
- [28] M. Abdulwahed and Z. Nagy, "Developing the trilab, a triple access mode (hands-on, virtual, remote) laboratory, of a process control rig using labview and joomla," *Computer Applications in Engineering Education*, 2010.
- [29] P. Orduña, E. Sancristobal, M. Emaldi, M. Castro, D. L. de Ipiña, and J. Garcia-Zubia, "Modelling remote laboratories integrations in e-learning tools through remote laboratories federation protocols," 2012 Frontiers in Education Conference Proceedings, vol. 0, pp. 1–6, 2012.
- [30] N. Li, A. C. Holzer, S. Govaerts, and D. Gillet, "Enforcing privacy for teenagers in online inquiry learning spaces," in Understanding Teen UX workshop at CHI Conference on Human Factors in Computing Systems, no. EPFL-CONF-197892, 2014.
- [31] S. Govaerts, Y. Cao, A. Vozniuk, A. Holzer, D. G. Zutin, E. S. C. Ruiz, L. Bollen, S. Manske, N. Faltin, C. Salzmman et al., "Towards an online lab portal for inquiry-based stem learning at school," in Advances in Web-Based Learning—ICWL 2013. Springer, 2013, pp. 244–253.
- [32] E. Bogdanov, D. Gillet, and C. Salzmman, "Widget-based approach for remote control labs," in 9th IFAC Symposium on Advances in Control Education, no. EPFL-CONF-178719, 2012.
- [33] P. Orduña, A. Almeida, D. López-De-Ipiña, and J. García-Zubia, "Learning analytics on federated remote laboratories: tips and techniques," Apr. 2014.
- [34] T. Hecking, S. Manske, L. Bollen, S. Govaerts, A. Vozniuk, and H. U. Hoppe, "A flexible and extendable learning analytics infrastructure," in Advances in Web-Based Learning—ICWL 2014. Springer, 2014, pp. 123–132.

- [35] D. Lowe, M. de la Villefromoy, K. Jona, and L. Yeoh, "Remote laboratories: Uncovering the true costs," in Remote Engineering and Virtual Instrumentation (REV), 2012 9th International Conference on. IEEE, 2012, pp. 1–6.
- [36] I. Titov, "Labicom. net-the on-line laboratories platform," in Global Engineering Education Conference (EDUCON), 2013 IEEE. IEEE, 2013, pp. 1137–1140.
- [37] A. Gordillo, E. Barra, D. Gallego, and J. Quemada, "An online e-learning authoring tool to create interactive multi-device learning objects using e-infrastructure resources," in Frontiers in Education Conference, 2013 IEEE. IEEE, 2013, pp. 1914–1920.
- [38] A. Gordillo, E. Barra Arias, and J. Quemada Vives, "Enhancing k-12 science education through a multi-device web tool to facilitate content integration and e-infrastructure access," 2013.
- [39] V. J. Harward et al., "The ilab shared architecture: A web services infrastructure to build communities of internet accessible laboratories," Proceedings of the IEEE, vol. 96, no. 6, pp. 931–950, 2008.
- [40] M. Schulz, F. Chen, and L. Payne, "Real-time animation of equipment in a remote laboratory," in Remote Engineering and Virtual Instrumentation (REV), 2014 11th International Conference on. IEEE, 2014, pp. 172–176.
- [41] F. Lerro, S. Marchisio, M. Plano, M. Protano, and O. Von Pamel, "A remote lab like an educational resource in the teaching of the physics of electronic devices," in International Conf. ICL, 2008.
- [42] C. Salzmann and D. Gillet, "Remote labs and social media: Agile aggregation and exploitation in higher engineering education," in Global Engineering Education Conference (EDUCON), 2011 IEEE. IEEE, 2011, pp. 307–311.
- [43] C. Salzmann, S. Govaerts, W. Halimi, and D. Gillet, "The smart device specification for remote labs," in 6th International Conference on Remote Engineering and Virtual Instrumentation, IEEE REV2015.



Pablo Orduña (M'05) is a full time researcher and project manager at the MORElab Research Group at DeustoTech Internet. He finished Computer Engineering in 2007 and his PhD in 2013 in the University of Deusto. During his PhD 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 WebLab-Deusto Research Group, leading the design and development of WebLab-Deusto.



Danilo G. Zutin is currently Research Assistant at the department of Engineering & IT at the Carinthia University of Applied Sciences, Austria and a PhD candidate at Technical University of Ilmenau, Germany. He holds a degree in electrical engineering at the State University of Sao Paulo (UNESP), Brazil, and obtained his Master degree in Systems Design (specialization in Remote Systems) at the Carinthia University of Applied Sciences in Villach, Austria. His research interests are in the field of remote engineering, online labs, remote control of devices

and software development for online labs. Danilo Garbi Zutin is currently a senior Researcher and team member of the Center of Competence in Online Laboratories and Open Learning (CCOL) at the Carinthia University of Applied Sciences (CUAS), Villach, Austria, where he has been engaged in projects for the development of online laboratories.



Sten Govaerts is a postdoctoral researcher at the REACT research group of EPFL, Switzerland. His research interests include user experience design, information visualization, and findability applied to both the technology-enhanced learning domain and contextualized music. Currently, he is working on the Go-Lab project (<http://www.go-lab-project.eu/>)



Irene Lequerica Zorrozuza received her Computer Science degree from the Polytechnical University of Madrid, Spain. Currently, she works as a researcher in a European remote laboratories project (GoLab) at the National University for Distance Education (UNED). Previously, she collaborated in other European projects related to the use of TICs in society (C@R) and the reuse of public data (HABITATS).



Philip H. Bailey is a Senior Project Manager with the Center for Educational Computing Initiatives (CECI), Massachusetts Institute of Technology, Cambridge. He joined CECI in 1992 and has worked on numerous CECI projects. Previously, he designed user interfaces for Bell Atlantic Software Systems, multimedia authoring tools for Times-Mirror, and manufacturing software for Honeywell/Bull.



Elio Sancristobal has a doctoral engineering degree from the ETSII (Industrial Engineering School) of the Spanish University for Distance Education (UNED), 2010. He also holds a Computer Science Engineering degree by the Salamanca Pontifical University (UPS), Madrid, 2002 and a Technical Engineering degree in Computer Networks (UPS), Madrid 1998. He worked for the University Distance Education Institute (IUED) from UNED. He is currently working as associated professor in the Electrical, Electronic and Control department from UNED.



Christophe Salzmann is a Senior Research Associate at the École Polytechnique Fédérale de Lausanne (EPFL). He received his MS degree in Computer and Information Sciences from the University of Florida in 1999 and his PhD degree from the EPFL in 2005. His research interests include Massive Open Online Labs (MOOLs) in collaboration with MOOCs, intelligent & collaborative systems, Web-based Interaction Systems, distributed web 2.0 technologies, real-time

interaction over the Internet. Christophe Salzmann and his colleagues developed and maintain one of the largest remote labs in the world where 32 devices are available 24/7, used by more than 200 students/year.



Luis Rodriguez-Gil is a Research Intern at the MORElab Research Group at DeustoTech-INTERNET. He finished his studies of a double degree in Computer Engineering and Industrial Organization Engineering in 2013, and is currently studying a MSc in Information Security. Since 2009, he has been involved in the WebLab-Deusto Research Group, collaborating in the development of the WebLab-Deusto remote laboratory management system. Throughout these years, he has published several peer-reviewed publications. He has also contributed to some Open Source projects.

Kimberly DeLong received the B.Sc. degree in computer science from Smith College, Northampton, MA, and the master's degree in computer systems engineering from Northeastern University, Boston, MA. She has been with the Center for Educational Computing Initiatives, Massachusetts Institute of Technology, Cambridge, since 1995. During this time, she has worked on many projects involving a broad range of technologies. She has been involved with the iLab project since 2002.





Denis Gillet leads an interdisciplinary research Group at the Swiss Federal Institute of Technology in Lausanne (EPFL). His research interests include Technologies Enhanced Learning (TEL), Human-Computer Interaction (HCI), Engineering Education, as well as Dynamic Coordination of Distributed Systems and Devices. His current research focus is on the design and deployment of social media platforms for Science Technology Engineering and Mathematics (STEM) Education and Knowledge

Management. Denis Gillet is Associate Editor of the International Journal of Technology Enhanced Learning. He has been an Executive of previous European research projects on Technology Enhanced Learning and is currently the Technical Coordinator of the Go-Lab Integrated European initiative investigating online labs federation and exploitation for inquiry-based learning at school.



Manuel Castro (M'87–SM'94–F'08) received the Industrial Engineering degree and the Ph.D. degree in Engineering from the ETSII/Madrid Polytechnic University, Spain. He received the Extraordinary Doctoral Award in the UPM and the Viesgo 1988 Award to the Doctoral Thesis improving the Scientific Research about the Industrial Process Electricity Application, as well as the 1997 and 1999 years UNED's Social Council Award for the Best Didactic Materials in Experimental Sciences and the

2001 Award for the Innovative Excellence in Teaching, Learning & Technology from the Center for the Advancement of Teaching and Learning.

He works as researcher, coordinator and director in different projects, ranging from systems applications of simulation techniques, solar system and advanced microprocessor system simulation to telematics and distance learning applications and systems, as well as computer-aided electrical engineering (CAEE), acting now as and senior technical director.

He is now with the UNED (Spanish University for Distance Education) as Professor of Electronics Technology inside the Electrical and Computer Engineering Department. He was previously UNED's New Technologies Vice-Rector, UNED's Information Services Center Director, Research and Doctorate Vice-director, Academic Affairs Vice-director of the Engineering School at UNED as well as Director of the Department. He worked for 5 years in Digital Equipment Corporation as senior system engineer. He has published different technical, research and teaching books and articles for journals and conferences as well as multimedia materials and radio and TV programs.

He belongs to the organizing committee of IEEE EDUCON, IEEE FIE (International and Europe Chair, 2000-2006), ISES, TAAE and SAAEI conferences as well as program and planning committees' member and reviewer and chairman of several ones. He is co-chair of the conference EDUCON 2010, TAAE 2010 and ICECE 2005. He was co-chair of the conference FIE 2014 (Frontiers in Education Conference organized in Madrid, Spain, by the IEEE and the ASEE). He is co-editor of IEEE-RITA and of the Electronic Journal of Spanish Chapter of the IEEE Education Society.

He is Fellow member of IEEE (for contributions to distance learning in electrical and computer engineering education) and member of the Administration Committee (AdCOM) (2005–2012) of the IEEE Education Society and Past President Jr (2015-2016) of the IEEE Education Society; Founder and Past-Chairman (2004-2006) of the Spanish Chapter of the IEEE Education Society, and Chair of the IEEE Spain Section (2010-2011). IEEE Region 8 Educational Activities Subcommittee Chair (2015-2016). He was awarded with the 2012 TAAE (Tecnologías Aplicadas a la Enseñanza de la Electrónica) to the Professional Life, the IEEE EDUCON 2011 Meritorious Service Award (jointly with Edmundo Tovar), of the EDUCON 2011 conference; 2010 Distinguished Member Award of the IEEE Education Society; 2009 Edwin C. Jones, Jr. Meritorious Service Award of the IEEE Education Society; with the 2006 Distinguished Chapter Leadership Award and for the collective work inside the Spanish Chapter of the IEEE Education Society with the 2011 Best Chapter Award (by the IEEE Region 8) and with the 2007 Chapter Achievement Award (by the IEEE Education Society). He is member of the Board of the Spanish International Solar Energy Society (ISES).



Diego López-de-Ipiña is an associate professor and principal researcher of "MORElab - Envisioning Future Internet" group and director of DeustoTech-INTERNET unit, associated to DeustoTech - Deusto Institute of Technology, University of Deusto in Bilbao, SPAIN. He is the director of the PhD program within the Faculty of Engineering of the University of Deusto. He received his PhD from the University of Cambridge, U.K in 2002 with a

dissertation entitled "Visual Sensing and Middleware Support for Sentient Computing".

He is responsible for the teaching modules "Software Process & Quality", "Advanced Software Development" and "Internet Protocols, Technologies and Services" offered in the BSc and MSc in Computer Engineering degrees offered by the Faculty of Engineering of the University of Deusto.

His main research interests are pervasive computing, internet of things, semantic service middleware, open linked data, social data mining and mobile-mediated and tangible human-environment interaction. He is currently focusing his work on the role of citizens as active data contributors to the knowledge of a city modeled as Linked Data. He is taking and has taken part in several big consortium-based research european (IES CITIES, MUGGES, SONOPA, CBDP, GO-LAB, LifeWear) and Spanish (THOFU, mIO!, ADAPTA, SABESS, PIRAmIDE, ACROSS) projects involving the adoption of mobile computing, semantic web, social data mining, linked open data, social robotics and Web 2.0 and beyond to novel Aml-related application areas such as urban computing, sustainable computing or AAL. He has more than 70 publications in relevant international conferences and journals on Ubiquitous Computing, Semantic Web, Middleware, Smart Cities and Aml, including more than 25 JCR-indexed journal articles.



Javier Garcia Zubia (M'08–SM'11) holds a PhD in Computer Sciences by the University of Deusto. He is a full professor in the Faculty of Engineering of the University of Deusto, Spain. His research interest is focused on remote laboratories: design, implementation and evaluation. He is the leader of the WebLab-Deusto research group.