Adding New Features to New and Existing Remote Experiments Through Their Integration in WebLab-Deusto

http://dx.doi.org/10.3991/ijoe.v7iS2.1774

P. Orduña[1], J. Irurzun[1], L. Rodriguez-Gil[1], J. García-Zubia[1], F. Gazzola[2], D. López-de-Ipiña[1]
[1] University of Deusto, Bilbao, Spain
[2] Federal Institute of Santa Catarina, IFSC, Florianópolis,Brazil

***Abstract*—During the last decade, efforts have been made in the development and publishing of remote experiments for educational purposes. In order to reduce the duplicity of work and to improve the common requirements that are shared by different remote laboratories, remote experiment management platforms have been developed, such as MIT iLabs, LabShare Sahara or WebLab-Deusto. In this paper, we describe how the development of experiments is handled in WebLab-Deusto, supporting both managed (developed used the APIs provided by WebLab-Deusto) and unmanaged experiments (using Virtual Machines or LabVIEW), and comparing both approaches. It also shows the results of integrating remote experiments under this system, with the use case of VISIR, the electronics remote laboratory developed in BTH.**

***Index Terms*—remote-labs, architecture, distributed**

I. INTRODUCTION

The use of Remote Laboratories is particularly useful nowadays. Since their creation, Remote Labs enable students to remotely access experiments which are physically located in the university, so students can be at home or in traditional libraries while using experiments. The quality of the experiment can be improved by the integration of the Remote Laboratory in other digital learning tools, or enhancing the remote collaboration. Furthermore, the costs of the experiments can be reduced given the degree of sharing among students that can be achieved when they are queued much easier than in a traditional laboratory. In summary, a traditional laboratory is tied to a set of geographical and temporal requirements that a Remote Laboratory can avoid, adding many benefits to this avoidance.

However, Remote Laboratories have pursued further benefits. Complex experiments such as VISIR[1] have been built [1], being so successful that VISIR has been deployed in 6 European universities at the time of this writing, along with other deployments scheduled in Asia, and other universities consuming it remotely. New approaches have focused on making it easy to deploy experiments [2] and even making them mobile [3].

In terms of supported devices, the range has been improved [4], supporting m-learning with Remote Labs. In this line, different approaches have been taken for developing Remote Laboratories and making them available without requiring plug-ins in the web browser or dealing with firewalls. They were majorly analysed in [5], and different groups have worked on web standards solutions (based on AJAX) since 2005 [6] to nowadays [7]. The range of remote laboratories in general is so wide [8] that tools to index are required and efforts to create and maintain these indexes have been placed, such as Lab2go [9]. The amount of research lines within Remote Laboratories has been extended to include quality of learning [10], even existing workshops focused on the educational outcomes of remote laboratories[2]. These outcomes have been particularly addressed by the survey developed as part of the LabShare project [11]. There are also efforts to consider the issue of group formation within the context of remote laboratories [12].

This increasing interest on Remote Laboratories has generated big efforts to share resources among different universities. The idea is that a provider university can share experiments that it is using with its students, so students of a consumer university can use them.

The MIT iLabs team[3] is pioneer in this field, having shared batch experiments among different universities since 2004 [14], and interactive experiments since 2008 [13]. The iLabs are the most popular remote laboratories platform, being used in different countries among the five continents.

In the same line, the LabShare project[4] in Australia is a publicly funded project focused on building a network of remote laboratories. As part of this project, the Sahara[5] platform has been developed, and it has recently created the LabShare Institute[6] as a not-for-profit organization that will be an independent service broker promoting, maintaining and even hosting remote laboratories.

In Europe, the LiLa project[7] (Library of Labs) was created focused on providing the LiLa portal which manages both virtual and remote experiments.

Efforts towards the integration of these efforts through interoperability bridges have been placed, as detailed in [15] for iLabs with Sahara. In fact, the Global Online Laboratory Consortium (GOLC[8]) was recently created to promote the development and sharing of remote laborato-

[1] http://openlabs.bth.se/

[2] GOLC 2011: Improving Laboratory Learning Outcomes

[3] http://ilab.mit.edu/iLabServiceBroker/

[4] http://www.labshare.edu.au/project

[5] http://labshare-sahara.sourceforge.net/

[6] http://www.labshare.edu.au/

[7] http://www.lila-project.org/

[8] http://www.online-lab.org/

ries, which includes members of most of the initiatives described, as well as members of WebLab-Deusto, described in this paper.

One of the core research lines in the field of Remote Laboratories is the rig management frameworks or remote laboratory platforms, which are focused on the management of the common requirements of the remote laboratories.

Every Remote Laboratory has a set of requirements that can be reused among other laboratories. For instance, most Remote Laboratories require some kind of scheduling: some Remote Labs will require a calendar so students book a certain moment to use the experiment, while other Remote Labs will handle this through a queue of users. Authentication (asserting that the user is who claims to be) and authorization (asserting the permissions of the user, such as what experiments can run) are other clear examples of requirements common to different laboratories, since it does not matter if the experiment is an electronics experiment or an experiment of physics to handle authentication and authorization.

A standalone Remote Laboratory can build all these features from scratch for that specific domain, without using a rig management framework. VISIR is an example of this: it is a domain specific -electronics- experiment, but it handles authentication, authorization, scheduling, user tracking, security, communications, etc. The problem of this approach is that it requires the experiment developers to keep and maintain all these layers of software.

As opposed to this approach, an experiment could be implemented using the tools provided by a rig management framework. With this approach, as the rig management framework publishes new versions, the experiment automatically supports new features developed by the particular framework. For instance, if developers of the rig management framework develop a new authentication scheme such as OpenID, the experiment will be accessible through OpenID. Without using a rig management framework, the experiment developers would need to implement this support for their particular experiment.

This paper explores the technical novelties of the WebLab-Deusto rig management framework in its current version (4.0M1), available for any experiment developed within the WebLab-Deusto architecture [16].

II. DEVELOPMENT OF EXPERIMENTS WITH WEBLAB-DEUSTO

A. *Introduction*

WebLab-Deusto[9] is an Open Source[10] remote laboratory platform [17] or rig management framework, developed in the University of Deusto. It has been used with students as part of their courses since February 2005 through different versions of the system. It manages rigs on top of which experiments will be run.

The development of experiments can currently be developed using two different approaches:

- Managed experiments
- Unmanaged experiments

B. *Managed experiments*

A Remote Laboratory is usually composed by two components:

- Client: it runs in a web browser or in the student's desktop. All what the user will see is what the client shows. Therefore, the client manages all the interaction with the student, translating the actions of the user into messages that will be sent to the server. Examples of clients would be Java Applets, Adobe Flash applications or the HTML code that will be shown in the web browser. It is sometimes referred as "rig client".
- Server: it runs attached to the hardware that will be used by the student, located in the university. This hardware might be an electronics board, a robotic arm, or any device that a lesson needs. The server code will receive requests from the client and will control the interaction with the hardware, avoiding harmful requests if possible. It can be developed in any typical technology, such as Java, .NET, Python, C++, etc. It is sometimes referred as "rig server" in the literature.

WebLab-Deusto embraces this clear client-server architecture in what it calls "managed experiments". It provides libraries both in the client side and the server side to handle communications and it also provides software components that will be in the middle between client and server. This way, it ensures that an unauthorized client can not perform requests to the rig server, since the software components placed in the middle will check that the client is who claims to be. The WebLab-Deusto servers, which are part of the software components that are in the middle, will keep a track of all the commands sent and received from the experiment. Furthermore, since the actual communications between client and server are performed through these software components, WebLab-Deusto provides a Secure Sockets Layer to avoid the experiment code to handle encryption issues. Given that all the communications, and therefore several security and tracking aspects, are handled by the platform itself, this type of experiment is called "managed", since they are fully managed by WebLab-Deusto.

As an example of managed experiment, in Figure 1 a FPGA experiment running in WebLab-Deusto can be appreciated. Once students have logged in the system, selected which experiment they want to run (FPGA in this case), waited in a queue (if there were other students using the available rigs), and sent a file that will be programmed in the FPGA, they will see this panel to control the device. Several buttons and switches are available, and students will see the result of their actions in the device when the webcam shows the results in the displays.

In order to develop this managed experiment, a pair of client and server software components must be developed.

To develop the client component, WebLab-Deusto provides libraries for JavaScript, Adobe Flash and Java Applets. With these libraries, the client only needs to perform some calls to a common API to send and receive commands and files to the server. In the case of the FPGA experiment, it will show the buttons, switches and the webcam in the web page. It will also define that when a switch is pressed, a command will be sent through the provided API, containing a string such as *"TURN SWITCH 1 ON"*.

[9] http://www.weblab.deusto.es/
[10] http://code.google.com/p/weblabdeusto/

To develop the server component, another library will be used. WebLab-Deusto provides libraries for C, C++, Java, .NET, Python or LabVIEW. This component will have a set of methods that will be called each time a command or file is sent. The server is expected to handle those commands and generate a proper response, which will be processed again in the client. In the case of the FPGA, the server will check *"TURN SWITCH 1 ON"* and will translate it to a digital signal sent to the FPGA as an input, and this FPGA will react doing something in the 7-segment display that students will see through the webcam.

The provided library in the client side will manage to send this command to the WebLab-Deusto servers, which will propagate it to the experiment server once they have tracked it and validated that the user has permission to use it. All this process is transparent for the experiment developer. While this process may seem that it adds a big latency, it has been measured and it is kept low, as shown in Figure 2, although it depends on the configuration used.

Therefore, the use of the managed experiments approach facilitates the development of secure experiments that can use the power of web applications, optionally without requiring any plug-in and thus being able to be used even from mobile devices. Furthermore, since WebLab-Deusto manages all the communications through standard HTTP/HTTPs ports, no issue will arise regarding HTTP proxies or institutional firewalls.

C. *Unmanaged experiments*

While the managed experiments approach facilitates the development of experiments, experiment developers still need to develop code. Therefore, those engineers who are experts in the domain of the experiments but not in web development will find it difficult to develop the experiments.

In order to tackle this problem, WebLab-Deusto started supporting "unmanaged experiments". Within WebLab-Deusto, unmanaged experiments are those whose communications are not handled directly by the platform. While the authentication, authorization, scheduling, load balancing and basic user tracking is still handled by the platform, the client will contact the server by its own, without relying on the WebLab-Deusto libraries.

As an example, Figure 3 shows a Virtual Machine (VM) running a LabVIEW experiment. When a user requests an experiment, WebLab-Deusto will load a Virtual Machine and will grant the user access to this Virtual Machine. This access will be performed through VNC, SSH or Remote Desktop, using a specific client or an applet within the web application.

In order to do this, WebLab-Deusto starts a VM using VirtualBox[11], starting from a particular snapshot to reduce the startup time to few seconds. Then, WebLab-Deusto generates a random password and will configure it in the VM. From this moment, the VM starts allowing connections with that password. WebLab-Deusto will send this password to the client, so it can start a connection to the Virtual Machine. At the moment of this writing, we support Secure Shell (SSH) and VNC in Linux VMs and VNC and Remote Desktop in Microsoft VMs.

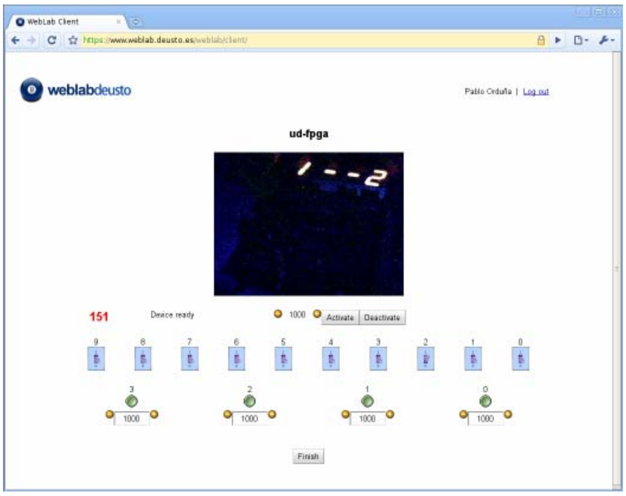


Figure 1: FPGA experiment running in WebLab-Deusto

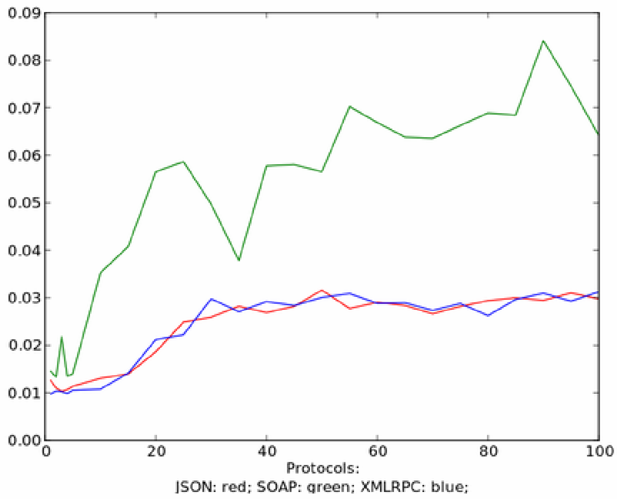
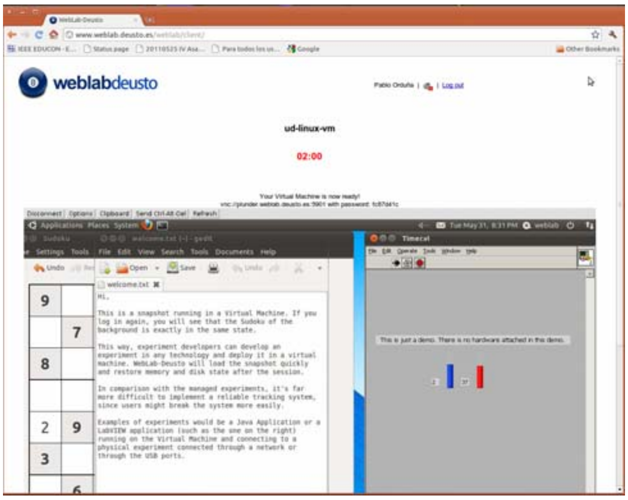


Figure 2: Latency added by the platform additionally to the network latency when sending a file, measured in seconds (Y axis) and in number of concurrent students (X axis)



Your Virtual Machine is now ready!
vnc://plunder.weblab.deusto.es:5901 with password: fc87d41c

```
Hi,

This is a snapshot running in a Virtual Machine. If you
log in again, you will see that the Sudoku of the
background is exactly in the same state.

This way, experiment developers can develop an
experiment in any technology and deploy it in a virtual
machine. WebLab-Deusto will load the snapshot quickly
and restore memory and disk state after the session.

In comparison with the managed experiments, it's far
more difficult to implement a reliable tracking system,
since users might break the system more easily.

Examples of experiments would be a Java Application or a
LabVIEW application (such as the one on the right)
running on the Virtual Machine and connecting to a
physical experiment connected through a network or
through the USB ports.
```

This is just a demo. There is no hardware attached in this demo.

Figure 3: Virtual Machine managed by WebLab-Deusto, showing a dummy LabVIEW experiment

[11] http://www.virtualbox.org/

The importance of these Virtual Machines is that now it is possible to develop experiments in any desktop technology in WebLab-Deusto. If an experiment is developed using the LabVIEW user interface, this experiment can be run on the Virtual Machine. As long as the Virtual Machine manages the hardware connections correctly, the experiment will run.

Using the same approach, other experiments based on LabVIEW Remote Panel are being adapted as shown in Figure 5 and it is currently experimental. A protocol has been developed between WebLab-Deusto and LabVIEW, so it loads the .vi that user has permissions to execute. Again, scheduling, authentication, authorization, and user tracking are handled by WebLab-Deusto automatically, so the experiment developer only needs to know LabVIEW.

However, the communications, as shown in Figure 4, are not handled through WebLab-Deusto. While the randomly generated password in VMs or secret in LabVIEW is sent through WebLab-Deusto securely, all the VNC/SSH/Remote Desktop/Remote Panel connection is managed directly. Therefore, network problems such as firewalls blocking the ports (5900 for VNC, 22 for SSH, 3389 for Remote Desktop) will arise. It will also not be possible to run these experiments from institutions using a common HTTP proxy, since these connections will probably be forbidden.

The security of these protocols can also become a problem. While SSH and Remote Desktop protocols are considered secure by design, VNC is not secure by default.

Finally, the fact that these communications are performed outside the WebLab-Deusto platform makes it difficult to deploy a reliable user tracking system, in comparison with the one developed in the managed experiments, where every command and response is stored and can be accessed by teachers, administrators and researchers.

While this solution is worse in terms of security, maintenance, or user tracking since it requires several ports to be open and a direct connection with the laboratory server that could be sniffed, the development of experiment becomes far easier since the developer can use any technology. Therefore, it is interesting to keep both approaches.

From the point of view of WebLab-Deusto developers, the use of unmanaged experiments should be avoided if possible, given the results in terms of relying on plug-ins and ports that can generate conflicts. However, wherever experiment developers find it difficult or not productive to deal with web technologies, unmanaged experiments provide an easy solution for experiment development.

D. *Features*

WebLab-Deusto provides several features to the experiment developers. Since these features are not domain dependent, they can be reused in all types of Remote Laboratories deployed on top of WebLab-Deusto.

First of all, WebLab-Deusto manages user authentication. It provides different authentication schemes to do so. The administrator of the system can select which scheme is more appropriate for each user or group of users. The following are the most commonly used schemes in WebLab-Deusto, which counts with other more complex authentication schemes:

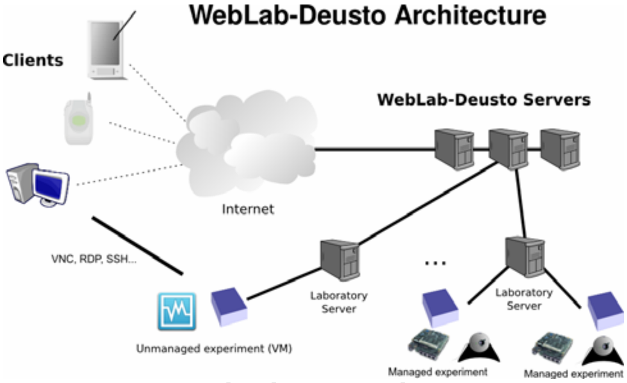


Figure 4: Managed and unmanaged experiments in WebLab-Deusto platform

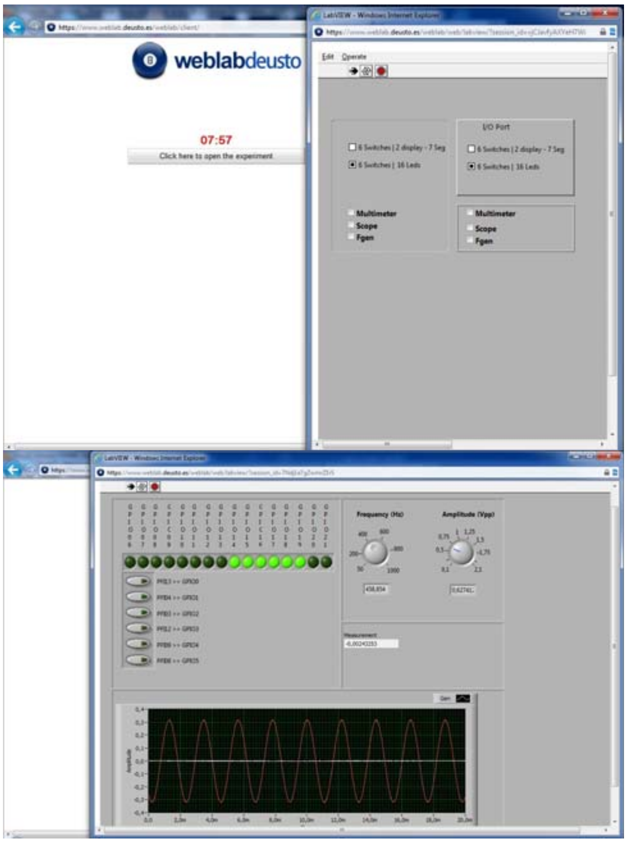


Figure 5: LabVIEW Remote Panel running in WebLab-Deusto

- Database: the most simple is storing a hash of the password in a database. Every time a user logs in, the system will check if the hash of the provided password is the same as the stored hash to grant access to the student.
- LDAP: given that a Remote Laboratory is a service provided by the university, it should be integrated as such and perceived as such by students. Asking students to have a particular password for the Remote Laboratory breaks this perception. Universities usually have an internally accessible directory service. The most popular protocol to access this directory service information is called LDAP (Lightweight Directory Access Protocol). WebLab-Deusto supports LDAP to validate the credentials provided by the student against a given LDAP server. Different LDAP servers can be configured, and each student will be configured to be validated against one or no

LDAP server. With this integration, when a local student changes his university credentials, the next time he accesses WebLab-Deusto he will be able to use the new credentials.

- OpenID: in order to share experiments with external universities, the Single Sign-On (SSO) OpenID protocol is supported. The provider university can create locally a set of users and establish that they will be authenticated through the OpenID server of the consumer university. Universities supporting this simple scheme (such as all the Spanish universities integrated in the RedIRIS SIR –RedIRIS Identity Service[12]–) will be able to become consumer universities. Furthermore, the integration of this system in certain Learning Management Systems (LMS) becomes very simple even with a small SCORM object given that WebLab-Deusto will rely on the same Single Sign-On server as the Learning Management System itself.

Secondly, WebLab-Deusto manages the experiment authorization. This is, it will check which users have permissions on which experiments, based on policies applied to the particular user or to a groups of users.

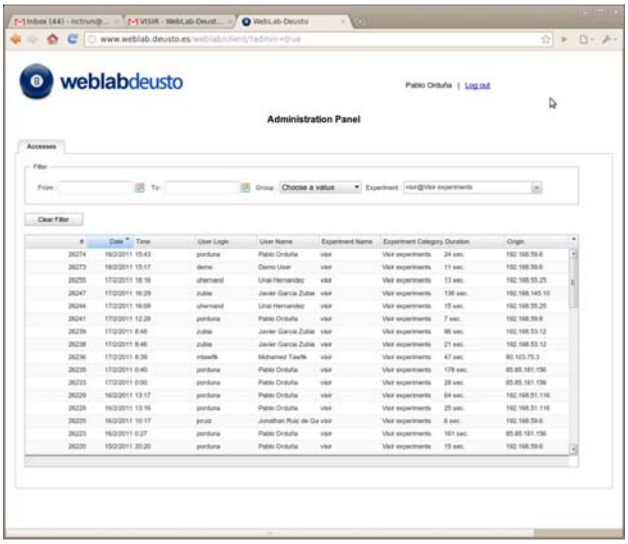
As any rig management framework, WebLab-Deusto clearly separates the experiment management from the experiment development, which tends to be domain-specific.

In terms of management, WebLab-Deusto is extensively used with students. Therefore, there was the need to create administration tools, both in Command-Line Interface (CLI) and web based (as shown in Figure 6). This administration tools are common for all the experiments. Therefore, once the administrator is familiar with the tools, managing more experiments is trivial since it is handled through the same system. Among the typical tasks that the administration tools will handle are the management of students and groups of students, permissions of these students, e-mail notifications to the students of a particular group, and tracking the uses of a given student. Regarding this user tracking, all the messages sent to the experiment are stored in a database, it is possible to keep a fine grained track of what the student did.

Another common feature required by all the Remote Laboratories is the scheduling management. In Remote Labs, usually a single user or a single group of users working together must be able to access the experiment at the same time. If one student is working on a lesson in a physical device, the rest of students should wait until this student has finished before using the experiment. Therefore, there must be a locking mechanism, where each user locks a certain rig while he is using the experiment or while the rig is preparing the experiment or the rig is cleaning the resources. In order to handle this problem, WebLab-Deusto provides an extensible scheduling system, at the moment based on priority queues.

At the moment of this writing, a rig in WebLab-Deusto can be used by different types of experiments. For instance, the same electronics device could be used for different lessons for different courses. At the same time, a given lesson can be built on top of different types of rigs. Computer engineering students in their first courses could learn how a simple educational microprocessor works. Since these simple educational microprocessors do not exist, one could build them in a programmable logic device such as a CPLD or a FPGA. However, there are other courses where CPLD and FPGA programming is taught. So it would be interesting that students of the first year use *either* a CPLD or a FPGA with an already provided scheme of the simple microprocessor, while, at the same time, higher degree students can perform more complex lessons on the same CPLDs, and students learning how to program a FPGA can use the available FPGAs. WebLab-Deusto supports these types of complex scheduling scheme, supporting even that the students of higher degrees have a higher priority since the other students can use a CPLD or a FPGA. It also supports correct load balancing of experiments, so different copies of the same rig can be provided to reduce the size of the queues.

In terms of user interface, WebLab-Deusto user interface has been adapted to mobile devices. While the support of mobile devices of each particular experiment depends on the technology used by the experiment developer, the WebLab-Deusto provides tools to make AJAX based applications accessible through mobile devices. This can be appreciated in Figure 7, where the same experiment runs with a very similar user interface in the desktop web browser (left) and in the mobile web browser (right), and it is detailed more in detail in [4].



Administration Panel

Accesses

Filter

From: To: Group: Choose a value Experiment: visir@Visir experiments

Clear Filter

#	Date	Time	User Login	User Name	Experiment Name	Experiment Category	Duration	Origin
26274	18/2/2011 15:43		porduna	Pablo Orduña	visir	Visir experiments	24 sec.	192.168.59.6
26273	18/2/2011 15:17		demo	Demo User	visir	Visir experiments	11 sec.	192.168.59.6
26255	17/2/2011 18:16		uhernand	Unai Hernandez	visir	Visir experiments	13 sec.	192.168.55.25
26247	17/2/2011 16:29		zubia	Javier Garcia Zubia	visir	Visir experiments	136 sec.	192.168.145.10
26244	17/2/2011 16:09		uhernand	Unai Hernandez	visir	Visir experiments	15 sec.	192.168.55.25
26241	17/2/2011 12:28		porduna	Pablo Orduña	visir	Visir experiments	7 sec.	192.168.59.6
26239	17/2/2011 8:48		zubia	Javier Garcia Zubia	visir	Visir experiments	86 sec.	192.168.53.12
26238	17/2/2011 8:46		zubia	Javier Garcia Zubia	visir	Visir experiments	21 sec.	192.168.53.12
26236	17/2/2011 8:39		mtawfik	Mohamed Tawfik	visir	Visir experiments	47 sec.	80.103.75.3
26235	17/2/2011 0:40		porduna	Pablo Orduña	visir	Visir experiments	178 sec.	85.85.181.156
26233	17/2/2011 0:00		porduna	Pablo Orduña	visir	Visir experiments	28 sec.	85.85.181.156
26229	16/2/2011 13:17		porduna	Pablo Orduña	visir	Visir experiments	64 sec.	192.168.31.116
26228	16/2/2011 13:16		porduna	Pablo Orduña	visir	Visir experiments	25 sec.	192.168.31.116
26225	16/2/2011 10:17		jeruiz	Jonathan Ruiz de Ga	visir	Visir experiments	6 sec.	192.168.59.6
26223	16/2/2011 0:27		porduna	Pablo Orduña	visir	Visir experiments	161 sec.	85.85.181.156
26220	15/2/2011 20:20		porduna	Pablo Orduña	visir	Visir experiments	15 sec.	192.168.59.6

Figure 6: Administration panel of WebLab-Deusto showing the use of the experiments by students

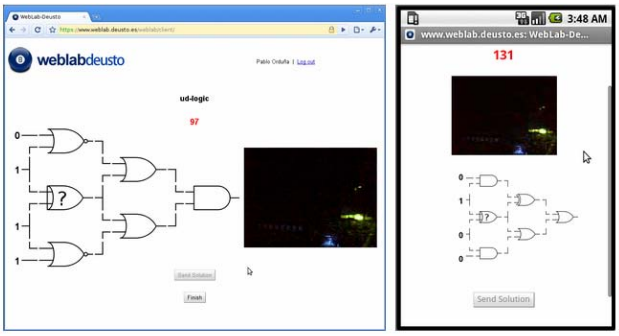


Figure 7 A concrete experiment running in a Desktop and in an Android mobile phone

[12] http://www.rediris.es/sir/index.html.en

Finally, WebLab-Deusto has recently started being integrated in Facebook. Given that Facebook supports OAuth 2.0, which is a protocol to provide authorization on resources placed in remote sites, WebLab-Deusto started integrating it as another authentication scheme. When a Facebook user logs in the Facebook WebLab-Deusto application[13], two options are shown: creating a new account and using the existing credentials of WebLab-Deusto. Once this step is performed, the Facebook account is linked with the WebLab-Deusto account. From this moment, users will see in the Facebook application those experiments they had permissions to use in their WebLab-Deusto account, but within the Facebook website, being able to use the provided tools (chat, etc.), as seen in Figure 8.

However, the most important point among all these features is that as more services and features are integrated in WebLab-Deusto, they are automatically provided to all the experiments in a seamless way. For instance, as the integration in Learning Management Systems with WebLab-Deusto improves [18], the experiments will be automatically integrated.

III. EVALUATION OF AN INTEGRATION EXAMPE: VISIR PROJECT

The VISIR project is a successful electronics Remote Laboratory developed in the BTH, which has been deployed in several european universities. The user interface is composed by a Flash application and different servers as detailed in [1]. As an evaluation of the design of WebLab-Deusto, authors try to integrate the experiment in this section.

In a regular VISIR session, the Flash client will send a set of messages and process the responses through a plain socket or an HTTP request. With help of the VISIR team, another "protocol" was introduced, which sent these messages through the WebLab-Deusto Flash libraries. Once in the server side, the VISIR measurement server receives the same requests that it receives in a regular session, so no code modification was required there. Therefore, with few glue code in the Flash communications layer the system and a rig server that proxied the requests to VISIR's server, the integration was running as a managed experiment within WebLab-Deusto.

This way, students of WebLab-Deusto see VISIR as yet another experiment in the list of experiments they have permissions to use. Since the authentication and authorization is handled through WebLab-Deusto, their credentials are validated with the LDAP directory of the university or through OpenID in case they are foreign students. Also, teaching staff that already use WebLab-Deusto will use the same administration tools as the rest of experiments of WebLab-Deusto to handle the permissions, and group management. Additionally, WebLab-Deusto tracks all the uses of VISIR, as well as all the commands sent and the responses generated by VISIR, and teachers have access to this information.

Furthermore, as seen in Figure 9, VISIR reuses the features provided by WebLab-Deusto, such as the Facebook integration described in the previous section.

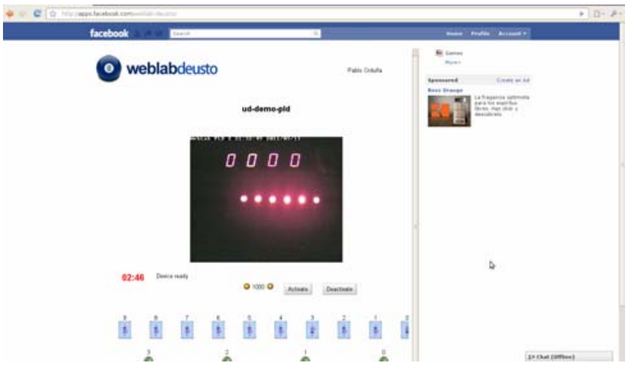


Figure 8 CPLD experiment running in Facebook through WebLab-Deusto

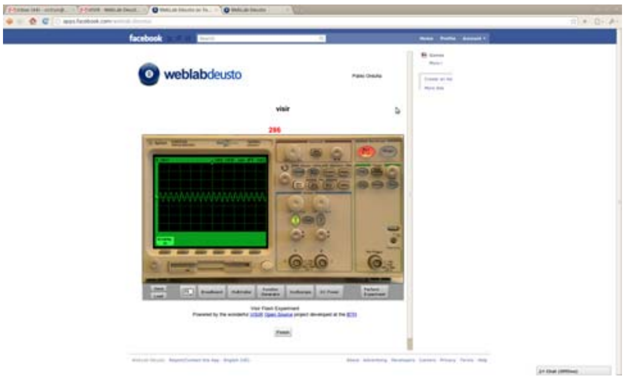


Figure 9 VISIR experiment running in Facebook through WebLab-Deusto

IV. CONCLUSIONS AND FUTURE WORK

In this paper, the basic concepts behind a rig management framework or remote laboratory platform were presented. The particular case of WebLab-Deusto, attending to the supported development approaches (called "managed experiments" and "unmanaged experiments") and features provided (complex authentication schemes, scheduling schemes, mobile devices support and facebook integration) were detailed.

Finally, it uses the integration of the VISIR project as a showcase that can be used as an evaluation of the proposed design.

Among the future work, the WebLab-Deusto team is currently working in three areas described in this paper:

- More complex scheduling schemes, focusing on the federation of experiments
- Interoperability with other Remote Labs such as iLabs or Sahara.

REFERENCES

[1] I. Gustavsson. et al. "The VISIR project--an open source software initiative for distributed online laboratories." Proceedings of the REV 2007 Conference, Porto, Portugal.

[2] J. García-Zubia et al. "Easily Integrable platform for the deployment of a Remote Laboratory for microcontrollers". IEEE EDUCON 2010 (ISBN: 978-1-4244-6570-5). 2010.

[3] Y. Lyalina; O. Makarov; R. Langmann. "Smart Labs for remote experimentation in communication technologies". Proceedings of the REV 2011 Conference, Brasov, Romania.

[4] P. Orduña et al. "Enabling mobile access to Remote Laboratories". IEEE EDUCON 2011 (ISBN: 978-1-61284-641-5). Amman, Jordan, April 2011.

[13] http://apps.facebook.com/weblab-deusto/

[5] J. García Zubia, P. Orduña, D. López-de-Ipiña, G. Alves. Addressing Software Impact in the Design of Remote Labs. IEEE Transactions on Industrial Electronics. ISSN: 0278-0046. DOI: 10.1109/TIE.2009.2026368. Volume 56, Issue 12, Dec. 2009 Page(s):4757 - 4767.

[6] J. García Zubía, D. López-de-Ipiña, P. Orduña. Towards a Canonical Software Architecture for Multi-Device WebLabs. IECON 2005, 31st Annual Conference of the IEEE Industrial Electronics Society, ISBN: 0-7803-9253, pp. 2146-2151, November 2005.

[7] J. Zornig, S. Chen, O. Al Kylaney. “A REST API Architecture for Remote Labs”. Proceedings of the REV 2011 Conference, Brasov, Romania.

[8] C. Gravier, J. Fayolle, B. Bayard, M. Ates and J. Lardon, State of the Art About Remote Laboratories Paradigms - Foundations of Ongoing Mutations. International Journal of Online Engineering, Vol 4, No 1 (2008).

[9] D. Garbi-Zutin, M. Auer, C. Maier, M. Niederstatter. Lab2go – a repository to locate educational online laboratories. Proceedings of the IEEE Education Engineering Conference (EDUCON) 2010. April 2010. Madrid, Spain.

[10] C. Gravier, J. Fayolle. Quality of learning: using a semantic web approach to enhance learner control during collaborative remote laboratories. International Journal of Innovation and Learning, vol. 6. Pp. 606-624. 2009. http://dx.doi.org/10.1504/IJIL.2009.026647

[11] T. Kostulski, S. Murray. “The National Engineering Laboratory Survey – A review of the delivery of practical laboratory education in Australian undergraduate engineering programs”. December 2010. Part of the outcomes of the LabShare Project: http://www.labshare.edu.au/media/img/labshare_report_panel_website.pdf

[12] A. Mujkanovic, D. Lowe. Policy-Based Remote Laboratory Multi-User Access Management. Proceedings of the REV (Remote Engineering and Virtual Instrumentation) 2010. Stockholm, Sweden.

[13] J. Harward, et al. “iLabs: A scalable architecture for sharing online laboratories”. International Conference of Engineering Education. 2004. Gainesville, Florida. October 16-21, 2004.

[14] 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, June 2008. http://dx.doi.org/10.1109/JPROC.2008.921607

[15] H. Yeung et al., “Interoperability of Remote Laboratories Systems,” International Journal of Online Engineering (iJOE). Vol 6. Special issue REV2010.

[16] P. Orduña et al. “Designing Experiment Agnostic Remote Laboratories”. Remote Engineering and Virtual Instrumentation. Bridgeport, CT, USA. June 2009.

[17] http://code.google.com/p/weblabdeusto/

[18] E. Sancristobal et al. “Service-Labs: Reusing Services and Laboratories from Open Learning Management Systems” ICL2009. Interactive Computer aided Learning. Villach, Austria: September 23-25, 2009.

AUTHORS

Pablo Orduña is with DeustoTech – Deusto Institute of Technology, University of Deusto, Avda Universidades 24, 48007, Bilbao, Spain (pablo.orduna@deusto.es).

Jaime Irurzun is with DeustoTech – Deusto Institute of Technology, University of Deusto, Avda. Universidades 24, 48007, Bilbao, Spain (jirurzun@deusto.es).

Luis Rodriguez is with the Faculty of Engineering, University of Deusto, Avda. Universidades 24, 48007, Bilbao, Spain (luis.rodriguez@ opendeusto.es).

Javier García-Zubia, is with the Faculty of Engineering, University of Deusto, Avda. Universidades 24, 48007, Bilbao, Spain. (zubia@deusto.es).

Fabricio Gazzola, is with the Faculty of Engineering, University of Deusto, Avda. Universidades 24, 48007, Bilbao, Spain. fabricio.gazzola@gmail.com).

D. López-de-Ipiña., is with the Faculty of Engineering, University of Deusto, Avda. Universidades 24, 48007, Bilbao, Spain. dipina@deusto.es).

This article is an extended version of a paper presented at the International Conference on Remote Engineering & Virtual Instrumentation (REV2011), held at Transylvania University, Brasov, Romania, June 28 - July 1, 2011. Received 31 July 2011. Published as resubmitted by the authors 28 September 2011.