Addressing low cost remote laboratories through federation protocols: fish tank remote laboratory

Ignacio Angulo, Javier García-Zubia
Faculty of Engineering,
University of Deusto
Bilbao, Spain
Ignacio.angulo@deusto.es, zubia@deusto.es

Pablo Orduña, Olga Dziabenko
DeustoTech – Deusto Institute of Technology
University of Deusto
Bilbao, Spain
Pablo.orduna@deusto.es, olga.dziabenko@deusto.es

Abstract— **A remote laboratory is a software and hardware tool which enables students to use real equipment -located in an educational institution- through the Internet. This way, students can experiment as if they were using the laboratories with their own hands. There are usually two approaches when designing remote laboratories: relying on small, inexpensive devices that can be deployed anywhere or relying software rich software infrastructures that support high load of users, providing panel administration, access to other institutional servers (e.g. directories such as Lightweight Directory Access Protocol "LDAP"), etc. With distributed remote laboratory architectures, it is possible to have the laboratory server on the former approach, but the management usually relies on the latter. In certain entities, such as secondary schools or farm schools, they may not be willing to buy and maintain a dedicated server for remote laboratories, and therefore the former approach is more adequate. However, a tradeoff is being made between management capabilities and how easy is to deploy the system. This contribution shows how federation could help in solving this tradeoff, and it uses a real fish tank remote laboratory as a case study.**

Keywords-component; remote-labs; federation; low-cost-systems

I. INTRODUCTION

The deployment of a remote laboratory usually requires the use of powerful back end systems that control access to experiments in an orderly manner. The heavy investment required and the necessary technical knowledge makes it difficult publishing experiments via the Internet by institutions and laboratories that own interesting experiments but lack of experience in software development. The focus of this contribution is if one laboratory can be deployed in low cost devices and still supports federation, then a regular remote laboratory management system, which handles hundreds of users, can provide management layers automatically. This way, the low cost laboratory can be deployed wherever in the internet, with its own policies, and different consumers (different secondary schools, universities, etc.) can consume it through a regular server which handles management layers (e.g. authentication backends not available in the laboratory side). This permits future deployments in scenarios which typically will not install a remote laboratory due to the costs and complexity of the system, such as secondary schools, farm schools, etc. In addition sharing laboratories between institutions favors collaboration and provides innovative didactical materials to educators, as result of the endeavor of each institution. So as to clarify the federations, section II describes more in detail this concept, section III assesses options for low cost servers for the development of this feature and section 0 describes the case study.

II. FEDERATIONS OF REMOTE LABORATORIES

A unique characteristic of remote laboratories when compared to traditional laboratories is that the distance of the student to the real equipment is not an issue, so remote laboratories can be shared with other institutions. One institution can share a laboratory to other institution. We refer to institutions rather than universities since they do not need to be universities: research centers may be interested in sharing local resources as part of an agreement, and secondary schools would reasonably be consumers.

This sharing can be managed in a direct, simple way: the provider entity (the one where the equipment is located) creates accounts of users of the consumer entity (the one interested in using the provider university's equipment for their students). Students of the consumer entity directly access in the provider entity and the provider entity does all the work: it authenticates the user, authorizes him to use the laboratory and provides the laboratory.

There are multiple problems with this solution. First, the provider entity must create and manage the user accounts of all the interested consumer universities. In a complex scenario, where a wide variety of consumers are involved -such as foreign universities and even secondary schools that simply do not speak the provider entity's language-, this approach does not scale. Second, the management of this approach is cumbersome: consumer universities would need to notify providers every change, and local databases or protocols such as LDAP would not be available. Third, the consumer entity cannot carry a proper accounting of the uses performed: it must trust the provider entity.

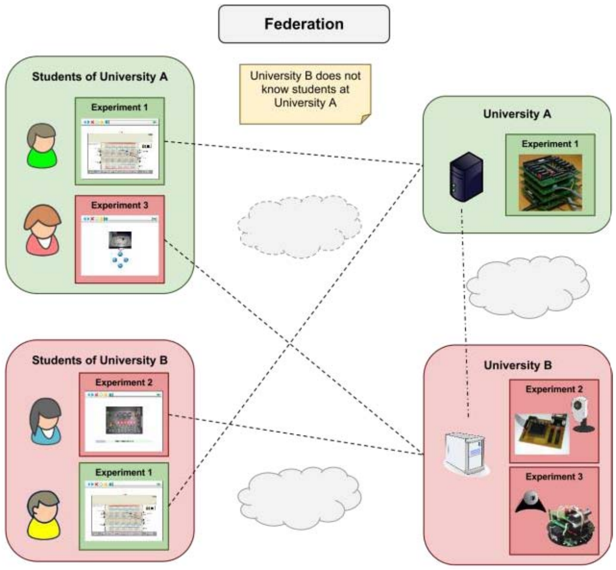

Figure 1 Example of remote laboratory federation: students of universities A and B accessing remote laboratories of each other

If both institutions come to an agreement where users of the consumer entity can access up to 10,000 times, there will be no way for the consumer entity to audit this if the provider entity at some point says "you have already reached the limit".

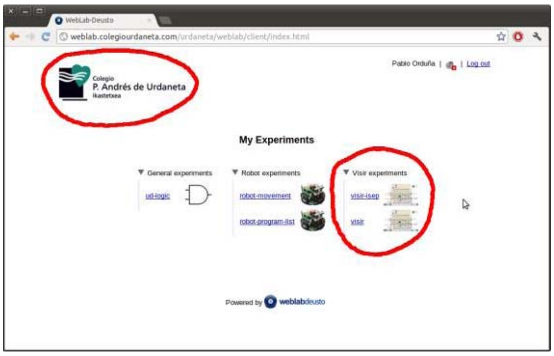

Figure 2 Secondary school Colegio Urdaneta accessing laboratories deployed in the University of Deusto through the WebLab-Deusto federation protocol

In order to handle these and other problems, a two-side model is required (see Figure 1), where both universities have the same software framework that manages this sharing. The consumer entity can authenticate and authorize local students, and once authorized, the local framework will contact the provider entity and request a slot. This way, the provider entity does not need to manage students and courses of the consumer entity, and the consumer entity can track all the requests performed to the provider entity, being able to track students and audit the overall use. In this sense, MIT iLabs have been effectively sharing remote laboratories around the world for years [1]. Different universities can use the MIT iLabs framework to develop, maintain and share their remote laboratories with other universities. In the federation model defined by the iLabs Shared Architecture (ISA), two types of remote laboratories can be shared: batch laboratories (using queues) and interactive laboratories (using a calendar-based booking system). As shown on Figure 2, WebLab-Deusto has also adopted a federation model described in [2], which provides new features such as federated load balance and transitivity which are outside the scope of this contribution.

III. SUPPORTING LOW COST SERVERS

Significant advances made in recent years in 32-bit microprocessors have permitted achieving a performance of embedded systems close to desktop systems [3]. This increased performance allows applications historically deployed in desktop computers or servers can now be implemented in embedded platforms. Furthermore the open source movement has spread to the hardware allowing the proliferation of low cost embedded platforms like RaspBerry Pi, Beagle Board, OlinuXino, Igep V2, etc. Alongside the advancement of culture DIY (Do It Yourself) in the world of electronics allows amateurs to control precise sensors and actuators without requiring extensive knowledge. This culture emphasizing open sharing, learning, and creativity has democratized the design of programmable electronic systems and development of simple embedded systems [5].

From a software standpoint, the functional requirements of WebLab-Deusto platform for deploying a remote laboratory are implementing an Ajax-enabled Web server, support for python 2.5 onward and access to a management system database. Thus, WebLab-Deusto supports resource constrained platforms, since the memory consumption can be limited and it supports a wide range of backend systems (such as lighthttp or sqlite).

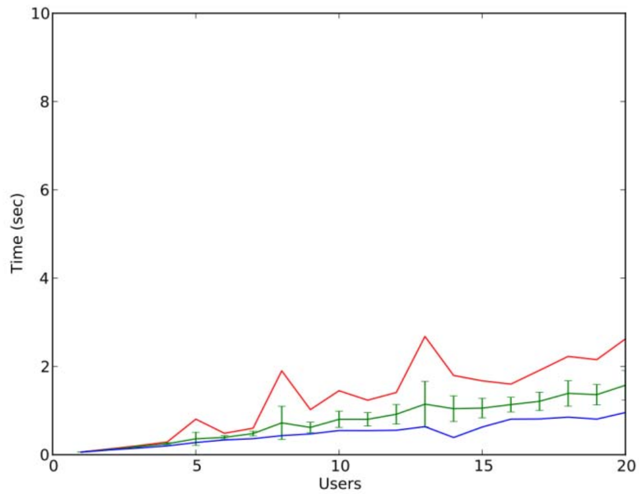

Figure 3.- WebLab-Deusto platform performance executed on a Raspberry Pi, supporting a service request getUserInfo () on JSON

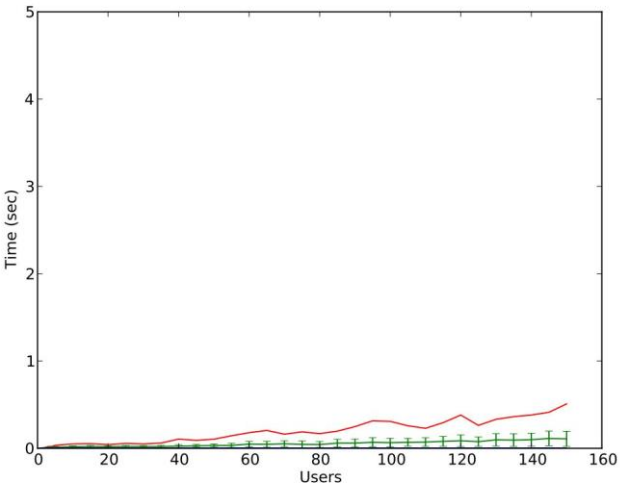

Figure 4.- WebLab-Deusto platform performance executed on a single core desktop computer, supporting a service request getUserInfo () on JSON

The performance of WebLab-Deusto platform has been tested over the popular board Raspberry Pi, Rev. B ($35) running a Debian 6 based Linux distribution (figure 3). Although the response time is considerably higher than that obtained on a desktop computer with a single core processor and 4GB of RAM (figure 4), the stress tests performed validate the operation of the system with a user load reduced. The main constraint in the performance of the raspberry pi is the RAM memory containing only 512Mb. Performance of the system improves substantially implemented over other embedded platforms like PandaBoard with 1Gb of RAM.

In addition, embedded platforms provide hardware resources that facilitate data acquisition relating to the experiment and allow remote interaction. These resources include GPIO, serial communications (UART, I2C, SPI, USB, etc.) or Pulse-width modulation (PWM) ports that facilitate controlling sectors or actuators. The common architecture of a remote laboratory [4] includes an experiment server deployed on an embedded development kit to provide the logic specific to the experiment. This server implements data acquisition and control tasks capable of digitizing the physical signals proper to the experiment and activating actuators to allow the user interact remotely with the experiment. The use of these platforms allows the implementation of main tasks corresponding to the core server, the laboratory server and experiment server in a single low cost device. Figure 5 shows the differences between the architecture of the remote laboratory WebLab-PIC, deployed at the University of Deusto for learning PIC microcontrollers [6], and the Fish Tank laboratory which will be detailed in section 4 as proof of concept. Some administration tasks of the Fish Tank laboratory, as authentication, fall on the main backend server of the WebLab-Deusto platform deployed at the University of Deusto.

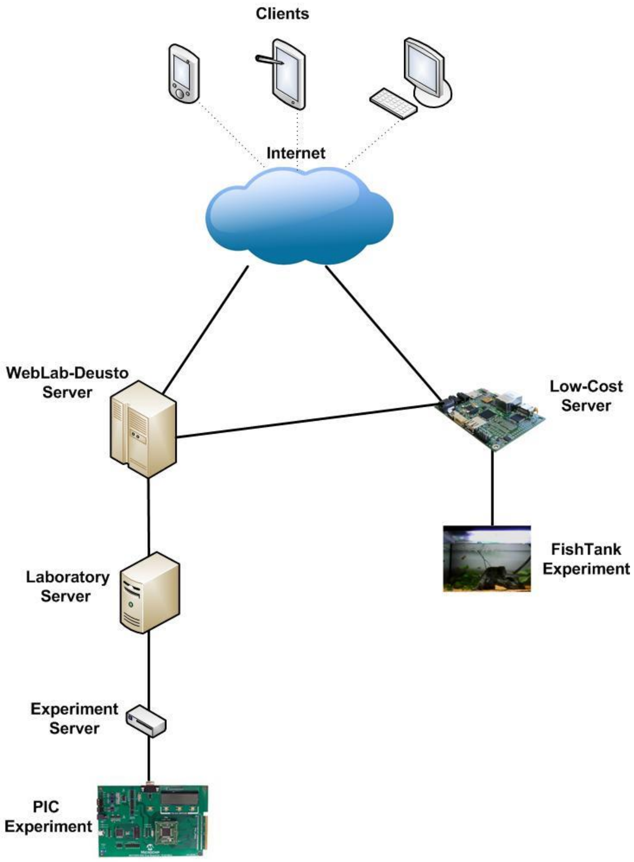

Figure 5.- Low-Cost server performs all management and control tasks needed to deploy the FISH TANK Laboratory

IV. CASE STUDY: FISH TANK LABORATORY

As a proof of concept, a fish tank laboratory has been developed, being accessible through the demo of WebLab-Deusto[1]. In this laboratory, users can connect and see several fish in a real fish tank in real time. Users can feed them, turn on and off the lights and even control a RC submarine, as long as it is in the fish tank and it has battery at that moment. In order to keep the fish safe, the system feeds them automatically if nobody has fed them recently, and avoids that users feed them too frequently.

[1] http://www.weblab.deusto.es/weblab/

Figure 6.- IgepV2 Board From ISEE©

The fish tank laboratory targets primary school students who are learning how to become responsible and to work in teams. Students should feed them with the different user accounts of a group, without killing them by starvation or food overdose. While the system internally prevents any of these conditions, it records all the attempts of each user, so it is possible for the teacher to evaluate if the group was working correctly and detect irresponsible behavior of the students in the care of the fish tank. The control of the fish tank alternates between the different groups within a class so that each week, the group at charge will investigate correct feeding and lighting patterns, based on the amount and type of existing fishes, and negotiate which participant should undertake the resultant tasks. The teacher can evaluate the quality of group work, active participation of each member and cooperation between members. The use of live pets in the classroom contributed positively to increased empathy, as well as socio emotional development, in students [7].

From a technical perspective, the laboratory has been developed using an IGEPv2 board [8]. Main features of the platform are:

- DM3730 Texas Instruments processor
- ARM Cortex A8 1GHz
- C64+ DSP 800MHz
- 3D Accelerator SGX530 @ 200 MHz
- Camera ISP
- 512 Megabytes RAM / 512 Megabytes FLASH
- Ethernet 10/100 Mb BaseT
- Wifi 802.11 b/g
- Bluetooth BC4 - Class 2.0
- Video: DVI-D (HDMI Connector) programmable panel size
- 2 x USB
- MicroSD card reader
- 3 x UART
- Stereo audio in / out
- Expansion connectors (GPIO, I2C, SPI, etc.)
- All included inside 65mm x 95mm form factor

The platform has been optimized to run Ubuntu 8.04 implementing and autonomous WebLab-Deusto platform with Apache 2 as web server and MySQL as database management system.

This system is federated with the WebLab-Deusto main server, but it could also be federated with other WebLab-Deusto servers (in primary schools). This way, the fish tank could be located somewhere else –e.g. in a primary school or aquarium centers- without a relevant deployment -small board, and the fish tank-, and be consumed by different entities –e.g. other primary schools-. These consumer entities could be deployed in regular servers with the advantages of it (supporting LDAP, a higher load of users, etc.).

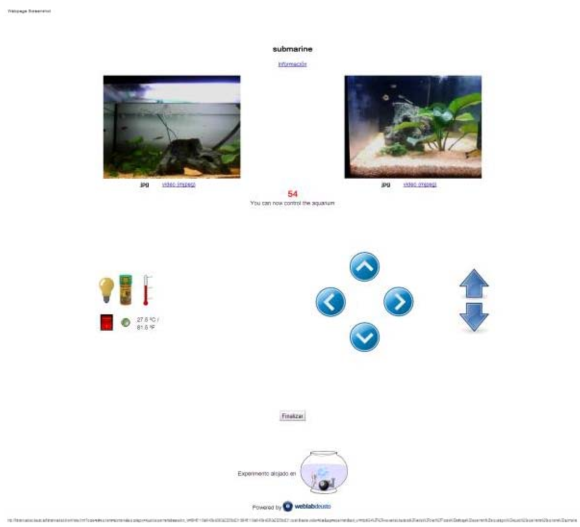

Figure 7.- Fish Tank laboratory User interface

The experiment has been implemented on a fish tank of 120 liters [8], with natural plants and live fish of the species "Brachydanio rerio", "Poecilia reticulata" and "Corydora" selected for their resistance. The experiment allows monitoring of the temperature of the aquarium water level and the control and feeding of fish and aquarium lighting. For the acquisition of the temperature sensor uses a Dallas Semiconductor

DS18B20 connected to the card via an ADC MCP3208 IgepV2 by SPI bus. Feeding is performed by a commercial automatic feeder conetado a GPIO V2 IGEP card through a darlington transistor that provides a TTL level. The original lighting system of the aquarium is controlled by a relay SPDT sealed controlled from another GPIO.

To promote access of the elementary students, the laboratory allows remote control of a commercial rc submarine that can move freely throughout the fish tank. The submarine is powered by a battery lasting about an hour, forcing to be replaced daily for carrying out the experiment in a school. User interface includes buttons to control the submarine. Six GPIOs directly connected to the buttons in the original remote control system allow control of the submarine.

Figure 8 Fish Tank laboratory

All materials used are easily affordable and the adequacy thereof has been published on the WebLab-Deusto project website to facilitate the replication of the laboratory.

V. CONCLUSIONS

The deployment of a remote laboratory requires a major economic and development effort in the provider institutions. The software infrastructure required for the provision of a remote laboratory demands complex workflow that forces acquisition of dedicated systems. The implementation of all necessary software services for developing a remote laboratory in low-cost embedded platforms allows institutions to deploy new remote laboratories without major investment (1). Furthermore the federation services provision regulates access to remote laboratories from different institutions (2). This paper shows how the combination of these two features will facilitate the deployment of remote laboratories that do not depend on the deployment location and do not require large financial investments for implementation.

Sharing remote laboratories between institutions makes profitable the development endeavor, improves collaboration between institutions entering the lab and allows students access to innovative teaching materials not available in their own institutions.

VI. FUTURE WORK

Another short term test concept is being developed consisting of an incubator for chicken eggs. This new remote laboratory will be deployed in a farm school in Bizkaia province in northern Spain [9]. Students can access during the gestation period of an egg (21 days) and observe the evolution of the process daily. Eggs undergo an ovoscope that allows monitoring the evolution of the chick inside the egg. The laboratory will measure the size of the embryo, and monitor the evolution of factors such as temperature and humidity inside the incubator.

Also in the medium term in order to facilitate the deployment of remote laboratories by institutions with no previous experience in the development of these, is being optimized the implementation of WebLab-Deusto in an open source hardware platform. A new expansion board will be designed to facilitate the interconnection of the system with the experiment providing regular inputs and outputs and the middleware that integrates the experiment into the client.

Figure 9 Baratze Farm School.

ACKNOWLEDGMENT

This work was supported by Lifelong Learning Programme of the European Union within Key Activity 3 - ICT (OLAREX project 518987- LLP-1-2011-1-ES -KA3-KA3MP). However, any opinions, findings, conclusions, and/or recommendations are those of the investigators and do not necessarily reflect the views of the European Commission.

REFERENCES

[1] V. J. Harward, J. A. del Alamo, S. R. Lerman, P. H. Bailey, J. Carpenter, K. DeLong, C. Felknor, J. Hardison, B. Harrison, I. Jabbour 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.

[2] P. Orduña, L. Rodriguez-Gil, D. López-de-Ipiña, J. Garcia-Zubia. Sharing the remote laboratories among different institutions: a practical case. In Remote Engineering Virtual Instrumentation 2012, REV 2012, July 2012.

[3] Harnett, C. Open source hardware for instrumentation and measurement. IEEE Instrumentation and Measurement Society, IEEE Instrumentation and Measurement Society ISSN: 1094-6969, Volume: 14, Issue: 3. June 2011.

[4] J. García Zubia, P. Orduña, I. Angulo, J. Irurzun, U. Hernández. Towards a Distributed Architecture for Remote Laboratories. Online Journal iJOE (International Journal of Online Engineering. ISSN: 1861-2121. Special Issue REV 2008, vol. 4.). 2008.

[5] S. Kuznetsov, and E. Paulos, Rise of the expert amateur: DIY projects, communities, and cultures. In Proc. NordiCHI '10, ACM Press (2010), 295-304.

[6] J. García-Zubia, I. Angulo, P. Orduña, U. Hernández, J. Irurzun, J. Ruiz, M. Castro, E. Sancristobal. Easily Integrable platform for the deployment of a Remote Laboratory for microcontrollers. IEEE EDUCON 2010 (ISBN: 978-1-4244-6570-5). 2010.

[7] B. Daly, S. Suggs. Teachers' experiences with humane education and animals in the elementary classroom: implications for empathy development. Journal of Moral Education, Volume 39, Issue 1, 2010.

[8] ISEE ©, IGEPV2 HARDWARE REFERENCE MANUAL, http://www.isee.biz/products/processor-boards/igepv2-board, April 2012.