Available online at www.sciencedirect.com

ScienceDirect

IFAC-PapersOnLine 48-29 (2015) 049–058

Facilitating the Creation of Virtual and Remote Laboratories for Science and Engineering Education

Francisco Esquembre

Departamento de Matemáticas, Universidad de Murcia, Murcia, Spain
(e-mail: fem@um.es).

Abstract: For roughly the past ten years, we have been working on creating and using virtual and remote laboratories for Science and Engineering education, and on providing a number of software tools that facilitate their creation. Virtual laboratories, or simulations, can be used to promote a more active role of students when studying certain phenomena. Remote laboratories add the extra value of using real hardware, typically at a distant location, which shows students the additional issues that appear when using real equipment. These pedagogical benefits are particularly effective if the laboratories are designed to be used using an Interactive Engagement approach. Our work, which received the gift of the collaboration with many other people interested in improving education, in particular in Physics and Control Engineering, has taught us what are the features and software platforms required to create virtual and remote laboratories. We review the lessons learned from the past ten years of successful outcomes and how we apply these lessons to prepare for the integration of computers, tablets, and other mobile devices with platform-independent cloud-based computing and laboratories.

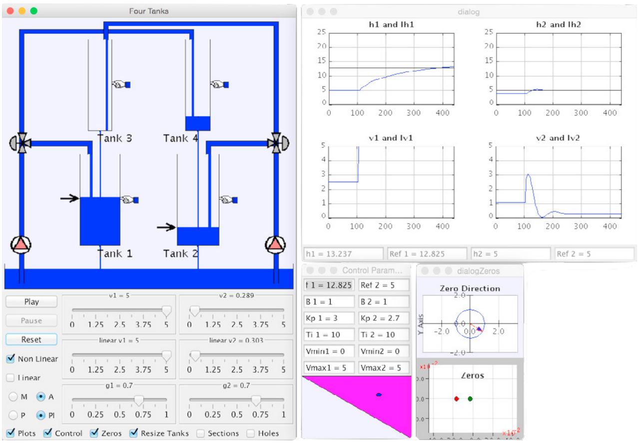
Keywords: Simulation, Virtual laboratory, Remote Laboratory, Science Education, Engineering Education.

1. INTRODUCTION

The use of computers for teaching is nowadays ubiquitous, although this term refers to a number of different activities in classroom, group and homework too varied to be described here. The same is true for the educational use of the Internet. In this paper, we concentrate in virtual and remote computer-based laboratories as particular forms of using both computers and the Internet to develop, deploy and use tools that can improve Science and Engineering education, in general, and Physics and Control education, in particular.

Virtual laboratories are computer simulations with typically high visualization and interaction capabilities, aimed to help students perform a given (simulated) scientific or engineering experiment. In virtual laboratories, the underlying physical process or engineering equipment is replaced by a software program that reproduces, approximates, or imitates it. Figure 1 shows our very first Control Engineering education virtual laboratory (Dormido and Esquembre (2003)).

Remote laboratories are computer programs that provide a graphical user interface (GUI) to interact with real hardware performing the experiment. This hardware is typically separated from the student's computer (sometimes at a distant location), but can be accessed, manipulated, and frequently visualized from the laboratory GUI using an intranet or Internet connection. Figure 2 shows one of our earliest Control education remote laboratories (Duro et al. (2008)).



h1 = 13.237 | Ref 1 = 12.825 | h2 = 5 | Ref 2 = 5

f 1 = 12.825
Ref 2 = 5
B 1 = 1
B 2 = 1
Kp 1 = 3
Kp 2 = 2.7
Ti 1 = 10
Ti 2 = 10
Vmin1 = 0
Vmin2 = 0
Vmax1 = 5
Vmax2 = 5

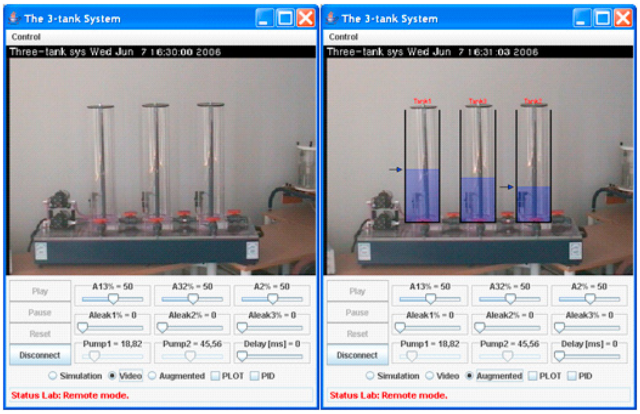
Fig. 1. The quadruple-tank process. The plant is simulated by software.

There are other types of computer-based laboratories used in Science and Engineering education (Dormido (2004)). Virtually every piece of slightly sophisticated experimentation equipment is subject to be used connected to or together with a computer in practical lectures. We do not cover these other laboratories in this paper.

Virtual and remote laboratories are a convenient and cost-effective way of improving education, because:

⋆ Research supported by the Spanish Ministry of Economy and Competitiveness (Grant MTM2014-52920-P) and the Fundación Séneca, Research Agency of the Region of Murcia, Spain, (Grant 19294/PI/14).

Peer review under responsibility of International Federation of Automatic Control.
10.1016/j.ifacol.2015.11.212



Three−tank sys Wed Jun 7 16:30:00 2006

A13% = 50
A32% = 50
A2% = 50
Aleak1% = 0
Aleak2% = 0
Aleak3% = 0
Pump1 = 18,82
Pump2 = 45,56
Delay [ms] = 0

Status Lab: Remote mode.

Three−tank sys Wed Jun 7 16:31:03 2006

A13% = 50
A32% = 50
A2% = 50
Aleak1% = 0
Aleak2% = 0
Aleak3% = 0
Pump1 = 18,82
Pump2 = 45,56
Delay [ms] = 0

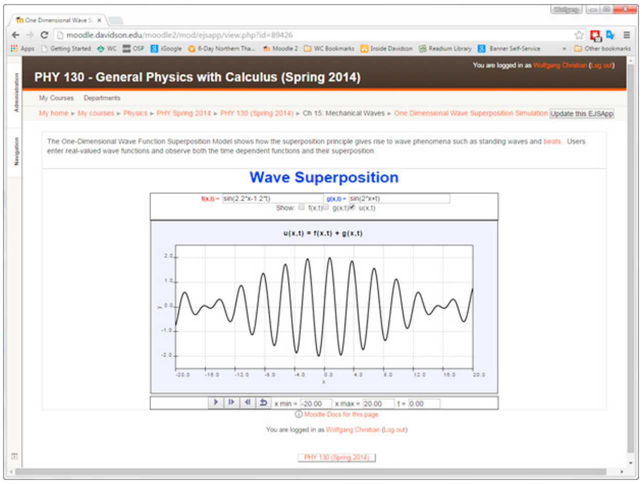
Status Lab: Remote mode.

Fig. 2. The three-tank system. The laboratory combined a virtual and a remote plant and superimposed their responses.

- They can be used to implement class and homework activities that promote Interactive Engagement (IE) educational approaches. IE encompasses a wide range of teaching methods in which students participate in hands-on, head-on activities that challenge them to learn, and has been proven to improve the student's understanding, motivation, and learning outcomes (Hake (1998)).
- They are reasonably cheap to produce and use. Even when real equipment – the most costly component of a laboratory – is used, the remote nature of the laboratory allows the consecutive use of the same equipment by a large number of students (which can access it 24 hours per day / 7 days per week), reduction of student transportation costs, and sharing of installations and maintenance expenses among several institutions.

Notice that the above list of conveniences of virtual and remote laboratories does *not* state that they are easy to create and deploy. They are not. Here is a list of possible barriers for faculty to create their own virtual and remote laboratories.

- As with any other software, programming expertise is required to create and maintain the programs that run on the students' computers (the *local* part of the laboratory). This expertise goes beyond being able to code domain-specific algorithms for simulating the process of interest. It is also frequently necessary to program numerical algorithms (e. g. for solving differential equations), graphical visualizations, and animation and interaction capabilities.
- When hardware is involved, assembling the equipment, designing and preparing the experiment, and making it accessible through an Internet server (the *remote* part of the laboratory) can be a very specialized task. Not to mention taking care of safety and security issues and providing a reservation system for scheduling students' access.
- Embedding the laboratory in an educational package that makes it usable by students requires not only writing the required narrative (instructions, theoretical background, explanations, and perhaps evaluation questions), but also presenting both parts, narrative and the experiment's user interface in a single, easy to use ensemble to the student. Figure 3 shows a virtual laboratory embedded in a Moodle-based Physics course (Christian and Belloni (2014)). Instructors may also want to allow for, or even encourage, collaboration among students.



PHY 130 - General Physics with Calculus (Spring 2014)

You are logged in as Wolfgang Christian (Log out)

The One-Dimensional Wave Function Superposition Model shows how the superposition principle gives rise to wave phenomena such as standing waves and beats. Users enter real-valued wave functions and observe both the time dependent functions and their superposition.

Wave Superposition

f(x,t) = sin(2.2*x-1.2*t)
g(x,t) = sin(2*x+t)

x min = -20.00
x max = 20.00
t = 0.00

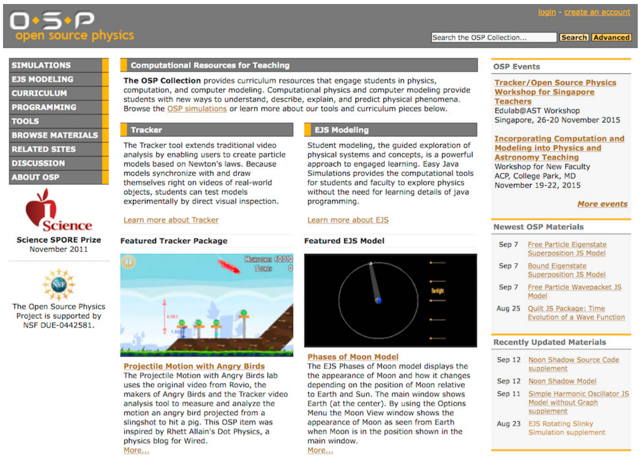
You are logged in as Wolfgang Christian (Log out)

Fig. 3. A simulation inside a Moodle course. The simulation runs embedded in instructional content.

- Successful deployment means making sure that students can access and run the laboratories at any time and in any platform they happen to use. From computers to modern tablets, under any possible operating system.

A final, important consideration when teachers decide to use a virtual or remote laboratory in their teaching, is not to find themselves *reinventing the wheel*. An important factor of the acceptance of these new teaching methods is the possibility of quickly finding already existing laboratories and associated curricular materials that are ready to use. Or close to it. A second important factor is the possibility to access them in a way that allows a teacher to *adapt and adopt them* (Redish (2000)). No matter how good the material is, it is more than likely that teachers will need or want to adapt it to their own teaching goals, methods, or tastes. Adopting a laboratory means that teachers need to get acquainted with the material and get familiar with it, until they feel completely at ease using it. Both because they think it is correctly implemented (the equations and algorithms are correct, the visualization is adequate, the interaction allows students to perform the experiments the teachers want. . .) and they can operate it effectively when using it in their lectures, or their students in the planned activities.

For roughly the past ten years, we have created and used, together with a number of collaborators around the world, a large number of virtual and remote educational laboratories, mainly in the discipline of Physics and Control Engineering. We also worked to help overcome the aforementioned barriers for faculty interested in creating new virtual and remote laboratories and to facilitate access to existing material to instructors interested in using them in their day-to-day teaching. Although attracted by new and effective teaching methods, faculty typically have heavy teaching duties and their main concern is to be able to cover the curriculum in the prescribed number of lessons.



Science SPORE Prize
November 2011

The Open Source Physics Project is supported by NSF DUE-0442581.

Computational Resources for Teaching

The OSP Collection provides curriculum resources that engage students in physics, computation, and computer modeling. Computational physics and computer modeling provide students with new ways to understand, describe, explain, and predict physical phenomena. Browse the OSP simulations or learn more about our tools and curriculum pieces below.

Tracker

The Tracker tool extends traditional video analysis by enabling users to create particle models based on Newton's laws. Because models synchronize with and draw themselves right on videos of real-world objects, students can test models experimentally by direct visual inspection.

Learn more about Tracker

EJS Modeling

Student modeling, the guided exploration of physical systems and concepts, is a powerful approach to engaged learning. Easy Java Simulations provides the computational tools for students and faculty to explore physics without the need for learning details of java programming.

Learn more about EJS

Featured Tracker Package

Projectile Motion with Angry Birds
The Projectile Motion with Angry Birds lab uses the original video from Rovio, the makers of Angry Birds and the Tracker video analysis tool to measure and analyze the motion an angry bird projected from a slingshot to hit a pig. This OSP item was inspired by Rhett Allain's Dot Physics, a physics blog for Wired.
More...

Featured EJS Model

Phases of Moon Model
The EJS Phases of Moon model displays the the appearance of Moon and how it changes depending on the position of Moon relative to Earth and Sun. The main window shows Earth (at the center). By using the Options Menu the Moon View window shows the appearance of Moon as seen from Earth when Moon is in the position shown in the main window.
More...

OSP Events

Tracker/Open Source Physics Workshop for Singapore Teachers
Edulab@AST Workshop
Singapore, 26-20 November 2015

Incorporating Computation and Modeling into Physics and Astronomy Teaching
Workshop for New Faculty
ACP, College Park, MD
November 19-22, 2015

More events

Newest OSP Materials

Sep 7	Free Particle Eigenstate Superposition JS Model
Sep 7	Bound Eigenstate Superposition JS Model
Sep 7	Free Particle Wavepacket JS Model
Aug 25	Quilt JS Package: Time Evolution of a Wave Function

Recently Updated Materials

Sep 12	Noon Shadow Source Code supplement
Sep 12	Noon Shadow Model
Sep 11	Simple Harmonic Oscillator JS Model without Graph supplement
Aug 23	EJS Rotating Slinky Simulation supplement

Fig. 4. The OpenSourcePhysics collection in ComPADRE.

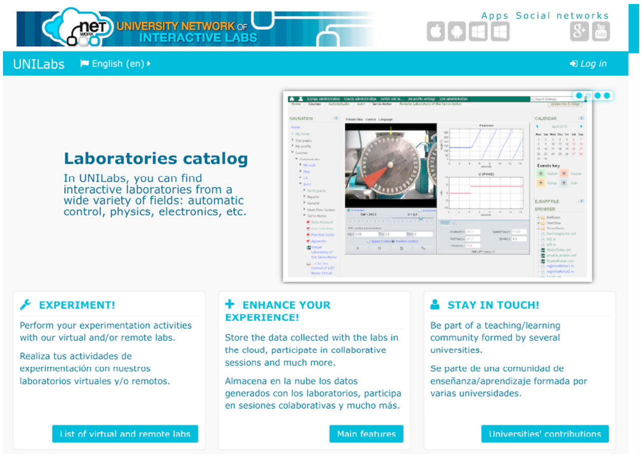
Barriers are therefore the last thing faculty need if we want to help education so that teachers and instructors adopt Interactive Engagement educational approaches.

This paper is an account of the experience accumulated as result of our work, as well as a reflection on which should be the bases for our future developments, in a moment when the advent of new hardware (tablets and big-screened smartphones on the users' side, cheap instrumentation hardware on the experimental side) and software platforms are changing the way students access and work with intranet- and Internet-accessible educational content.

The experience steams from our joint work with researchers and teachers from other groups. In the field of Physics Education, the main and invaluable collaboration has been with the OpenSourcePhysics project and the ComPADRE digital library. The OpenSourcePhysics project aims to provide curriculum resources that engage students in physics, computation, and computer modeling (Christian (2015)). Members of this team include Wolfgang Christian (Managing Editor), Mario Belloni, Doug Brown, Anne Cox, and Aaron Titus. ComPADRE is a Digital Library supported by the American Association of Physics Teachers with high quality educational resources of different types in physics and astronomy (Mason (2015)). Members of ComPADRE include Bruce Mason (Editor), Lyle Barbato, and Matt Riggsbee. Figure 4 shows the home page of the OpenSourcePhysics collection in ComPADRE.

In the field of Control Education, the main and equally invaluable collaboration has been with the group at the Computer Science and Automatic Control Department of the Spanish Distance University (UNED) lead by Sebastián Dormido. The group lists Raquel Dormido, Natividad Duro, Gonzalo Farias, Rubén Heradio, José Sánchez, Luis de la Torre, Héctor Vargas, and many others. Their most recent work includes UNILabs, a network of virtual and remote laboratories shared among different Spanish universities (UNILabs (2015), Dormido et al. (2011), de la Torre et al. (2015)). See Figure 5.

The paper is organized as follows. Section 2 lists the main lessons we obtained during our past work. We then expand on these lessons to explain how we applied them (at the same time we were learning them) to our work.



Laboratories catalog

In UNILabs, you can find interactive laboratories from a wide variety of fields: automatic control, physics, electronics, etc.

EXPERIMENT!

Perform your experimentation activities with our virtual and/or remote labs.

Realiza tus actividades de experimentación con nuestros laboratorios virtuales y/o remotos.

ENHANCE YOUR EXPERIENCE!

Store the data collected with the labs in the cloud, participate in collaborative sessions and much more.

Almacena en la nube los datos generados con los laboratorios, participa en sesiones colaborativas y mucho más.

STAY IN TOUCH!

Be part of a teaching/learning community formed by several universities.

Se parte de una comunidad de enseñanza/aprendizaje formada por varias universidades.

Fig. 5. UNILabs offers a network of virtual and remote collaborative laboratories.

Section 3 shows our pedagogical approach. Section 4 lists the technological standards we have adopted. Section 5 describes the tools we have developed or used to support our work. Section 6 explains our collaboration efforts. Finally, Section 7 provides some conclusions and future directions of our work.

2. LESSONS LEARNED

We devote a separate section to state the main lessons learned, in an attempt to make them stand very clearly from the rest of the paper (in which we provide details). If we could carve these lessons into stone, we would.

Lesson 1. Appreciate and exploit the educational power of virtual and remote laboratories.

Technology by itself does not improve education. Virtual laboratories can and should be used to prepare a wide variety of Interactive Engagement classroom activities that pose intellectual challenges to students. An appropriated use of them can therefore noticeably improve Science and Engineering education. Remote laboratories add extra interesting features. Using real hardware increments the sense of reality for the student, which increases motivation. But, also, hardware introduces additional issues, such as error in measurements or non-immediate response times, which only appear when using real equipment and that can be key in real-life situations.

Lesson 2. Start with virtual laboratories.

The extra advantages of remote laboratories come at a cost. Creating remote laboratories is harder than creating virtual ones, simply because they involve instrumentation hardware, data acquisition software and hardware, and configuration of a non-trivial Internet service. While a single person can easily create dozens of ready-to-use virtual laboratories, making remote laboratories available to students is more a teamwork.

Lesson 3. Follow standards or well-established platforms.

With respect to technology, follow industry standards whenever available. Spending a lot of time and effort (not to mention money) and finding yourself that all your work is outdated or simply unusable because the industry has decided not to support a platform that your laboratory depends on heavily to just run, is a bad, bad experience. (And it happens!) If an industry standard has not been established for some part of your work, adopt well-established platforms or de-facto standards that may increase the likelihood that your work will be valid for longer time.

Follow also educational standards. Consistent research has shown which pedagogical strategies work and which don't in Science and Engineering education. Unless pedagogical innovation is your own field of research, follow well-established paradigms.

Lesson 4. Use appropriated tools. Preferably free, if not open source.

Search for the most appropriate tool for your work, before you even start it. Choosing the right tool may not only decrease your development time. It can be the difference between failure and success. Fortunately, there is a wide variety of tools available. For every pocket.

Use affordable, if not completely free, tools. At least, choose tools that do not oblige you to pay an expensive yearly fee, unless you are sure you can afford it. This may sound like a poor-man's claim. But great laboratories, especially remote ones, created in the realm of a funded research project may suddenly die if the end of the project implies lack of resources to pay a required license or fee. Typically, school budgets do not contemplate big yearly expenses in licenses, or have other priorities. Plus, free – and specially open source – tools increment your opportunities for collaboration (see next lesson).

Lesson 5. Be open to collaborate and share.

You may feel shy, protective, or proprietary about your laboratory, but you certainly win more than you loose when you share it generously with others. Be also open to collaborate with others to improve it, even if this means sharing part of the authorship.

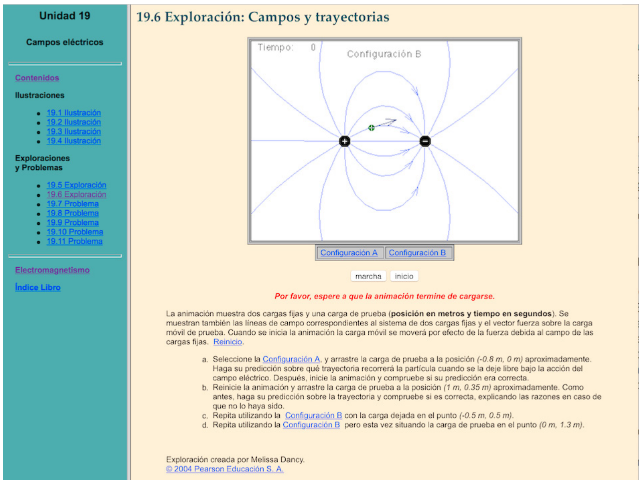
This has some basic limitations. Share your work with people who will respect and honour your authorship (giving you credit or respecting your copyright license) and who will also share their developments or ideas with you. With these basic precautions, you'll feel much happier with your work when you have others using it and providing you feedback and ideas about how to improve it.

Collaboration will also help you avoid reinventing the wheel yourself.

Lesson 6. Concentrate on good ideas, not just good implementations.

No matter how much state-of-the-art technologies you use, no matter which universal platform you adhere to, eventually, inevitably, your laboratory will stop running or will become obsolete. Industry changes, fashion (in users' preferences and in educational tendencies) changes, equipment improves, the whole society evolves.

This should not discourage you in the least. Do what you want to do, create excellent laboratories… but focus always on good ideas, not just on particular great implementations. A good implementation, even one based on the previous lessons, has an unavoidably short life. We dare to predict a lifetime of around 10 years for a good (a very good) virtual or remote laboratory. But good ideas live much, much longer.



19.6 Exploración: Campos y trayectorias

Por favor, espere a que la animación termine de cargarse.

La animación muestra dos cargas fijas y una carga de prueba (**posición en metros y tiempo en segundos**). Se muestran también las líneas de campo correspondientes al sistema de dos cargas fijas y el vector fuerza sobre la carga móvil de prueba. Cuando se inicia la animación la carga móvil se moverá por efecto de la fuerza debida al campo de las cargas fijas. Reinicio.

a. Seleccione la Configuración A, y arrastre la carga de prueba a la posición *(-0.8 m, 0 m)* aproximadamente. Haga su predicción sobre qué trayectoria recorrerá la partícula cuando se la deje libre bajo la acción del campo eléctrico. Después, inicie la animación y compruebe si su predicción era correcta.
b. Reinicie la animación y arrastre la carga de prueba a la posición *(1 m, 0.35 m)* aproximadamente. Como antes, haga su predicción sobre la trayectoria y compruebe si es correcta, explicando las razones en caso de que no lo haya sido.
c. Repita utilizando la Configuración B con la carga dejada en el punto *(-0.5 m, 0.5 m)*.
d. Repita utilizando la Configuración B pero esta vez situando la carga de prueba en el punto *(0 m, 1.3 m)*.

Exploración creada por Melissa Dancy.
© 2004 Pearson Educación S. A.

Fig. 6. This Physlets-based animation shows the electric field created by two fixed charges and provides a test charge for students to move interactively.

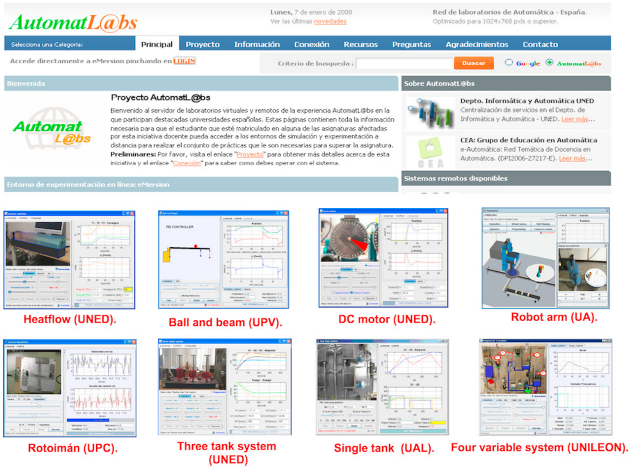
Create good, both flexible and robust, architectures for your laboratories. Design innovative, thought-provoking experiments. Create challenges for your students that motivate them to learn. Implementation details may change (will change!), but ideas may be transferred from one implementation to the future one, whatever the future may bring.

These are the main lessons distilled from our own experience. The rest of the paper provides details on our particular choices when confronted with the task of creating and using virtual and remote laboratories. Other choices are, of course, possible.

3. PEDAGOGICAL USE

We have certainly learned Lesson 1. We not only work on creating tools for people to create laboratories that they can use in their teaching. We actually use laboratories for our own teaching, as university faculty ourselves. And we do so in different disciplines and educational levels.

Our use of simulations in actual teaching has its roots back in 1995, when Wolfgang Christian created at Davidson College a collection of scriptable Java applets to help visualize and explain physics concepts. These *Physlets* simulations became tremendously popular and have had a lasting impact on Physics education. Associated originally with the Just-in-Time Teaching approach (Novak et al. (1999)), the technology was adopted by many teachers around the world and used in different teaching strategies. Perhaps the best examples of this are the *Physlets Physics* book (Christian and Belloni (2003)) and those which followed it (see an account of the history and evolution of Physlets in Christian et al. (2015)). Figure 6 shows the Spanish version of a Physlets-based *Exploration* (Esquembre et al. (2004), Christian and Belloni (2000)).



AutomatL@bs

Lunes, 7 de enero de 2008
Ver las últimas novedades

Red de laboratorios de Automática - España.
Optimizado para 1024x768 pxls o superior.

Selecciona una Categoria: Principal | Proyecto | Información | Conexión | Recursos | Preguntas | Agradecimientos | Contacto

Accede directamente a eMersion pinchando en LOGIN

Criterio de busqueda : Buscar ○ Google ● AutomatL@bs

Bienvenida

Proyecto AutomatL@bs

Bienvenido al servidor de laboratorios virtuales y remotos de la experiencia AutomatL@bs en la que participan destacadas universidades españolas. Estas páginas contienen toda la información necesaria para que el estudiante que esté matriculado en alguna de las asignaturas afectadas por esta iniciativa docente pueda acceder a los entornos de simulación y experimentación a distancia para realizar el conjunto de prácticas que le son necesarias para superar la asignatura. **Preliminares:** Por favor, visita el enlace "Proyecto" para obtener más detalles acerca de esta iniciativa y el enlace "Conexión" para saber como debes operar con el sistema.

Sobre AutomatL@bs

Depto. Informática y Automática UNED
Centralización de servicios en el Depto. de Informática y Automática - UNED. Leer más...

CEA: Grupo de Educación en Automática
e-Automática: Red Temática de Docencia en Automática. (DPI2006-27217-E). Leer más...

Entorno de experimentación en línea: eMersion

Sistemas remotos disponibles

Fig. 7. The pioneering network of Automatic Control laboratories AutomatL@bs.

Technology evolved, but the pedagogical approach and our use of virtual laboratories (commonly referred to as simulations in Physics education) remains based on Interactive Engagement techniques, in general, and the *Learning Cycle*, in particular (Christian et al. (2011)).

In the field of Control education, perhaps our best pioneering example was the collection of virtual and remote laboratories created by a Spanish group of universities in the AutomatL@bs project (Vargas et al. (2011)). See Figure 7.

The practical nature of laboratory work was carefully wrapped in a series of activities designed to achieve the desired pedagogical goal. Students had to complete a series of pre-lab activities, including reading theory, instructions, and conducting interactive experiments in virtual laboratories, before they were allowed to connect to the remote laboratories to complete the prescribed in-lab activities. This was a nice and complete example of the combined use of virtual and remote laboratories. This pioneer work has been continued and has evolved in more recent developments (de la Torre et al. (2015), Sáenz et al. (2015)).

Finally, in the field of Mathematics, the author of this paper teaches a course of Mathematical Modelling at the Faculty of Mathematics of the University of Murcia (Esquembre (2015)). Third year Math majors learn mathematical modelling through having to create complete simulations of simple physical phenomena. Figure 8 shows a students' group simulation of the *Synkope* amusement park ride (Medina et al. (2014)). This upper-division course goes one step beyond in using the pedagogical power of virtual laboratories, by asking students to create their own models and simulations in a Computational Modelling approach (see, for instance, Landau et al. (2008)).

4. TECHNOLOGY STANDARDS

One of the software tools that all computing devices, including mobile platforms, are sure to have installed is a World Wide Web (WWW) browser. Therefore, distributing and running pedagogical programs assuming the sole support of a Web browser is a safe bet.

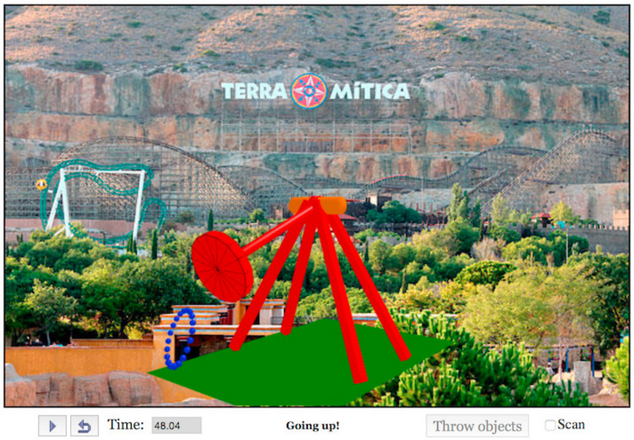


Fig. 8. A students' group simulation of an amusement park ride.

Of course, instructors can ask their students to install particular programs or platforms, including proprietary ones, in order to run their laboratories. But this requirement increases the likelihood of installation problems, need to pay licenses, or unsupported platforms. For this reason, we gradually adopted the Web browser as our preferred distribution and run-time platform.

Internet is ruled by a number of bodies which set different standards (Simonelis (2005)). The WWW standards are developed by the World Wide Web Consortium (W3C (2015a)). The standards for Web design and applications are listed in their web site, and features HTML and CSS for composing pages, and Javascript and SVG for programming and graphics. The standards also include other conveniences, such as MathML and WebGL for displaying mathematical formulas and 3D drawing, respectively.

This is the programming platform we adhere to. We recognize the power and suitability of other platforms and programming languages for creating laboratories (Matlab, LabView, Python, and Java, to name just a few). But few of them, if any, fully adhere to these standards. In particular, Java, perhaps the first programming language to be integrated in Web browsers and one which we used as run-time platform for our simulations for many years, has recently lost the favor of the big public (due to so many security problems) and Google has decided to discontinue support for Java applets in its Chrome browser (Google (2015)).

With respect to publishing standards, the International Digital Publishing Forum (IDPF) is a global organization that establishes standards in the realm of electronic publishing and content consumption (IDFP (2015)). IDFP establishes EPUB as their "distribution and interchange format standard for digital publications and documents based on Web Standards".

Respecting the W3C and EPUB standards is perhaps the best guarantee to be able to create software laboratories that can be run on almost any computing platform and that can be embedded, if desired, in electronic publications that can be read on most platforms.

Other important technologies are not industry standards, but well-established, de-facto standards. One of these is

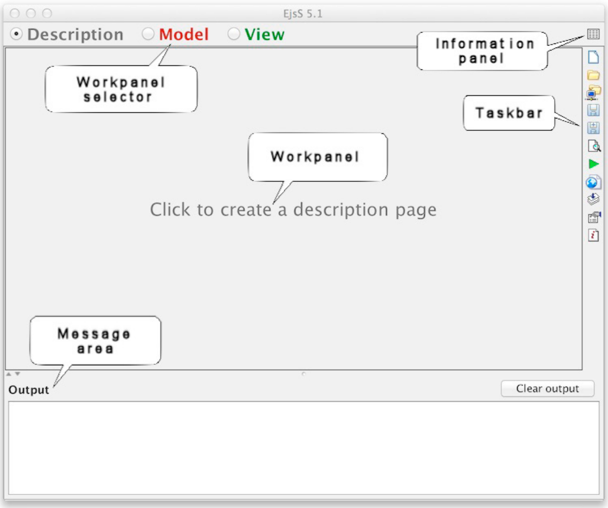


Fig. 9. The interface of EjsS in its original Java flavor. Authors fill the different panels to design their simulations.

Moodle, one of the most popular learning management systems. Moodle is a well-tested platform and has a very active supporting community. Moodle "delivers a powerful set of learner-centric tools and collaborative learning environments that empower both teaching and learning" (Moodle (2015)), and we have used it effectively to host some of our curricular material which not only included virtual or remote laboratories, but also provided a reservation system and opportunities for collaborative work (Junkin et al. (2015), UNILabs (2015), de la Torre et al. (2015), Sáenz et al. (2015)).

5. TOOLS

5.1 Modeling and authoring tool

Perhaps the central piece of our toolbox is the Easy Java/Javascript Simulations (EjsS) authoring and modelling tool. Since its public presentation (Esquembre (2004)), EjsS has been growing to include every tool and functionality that we needed to easily create both virtual and remote laboratories. From new visualization and interaction tools, to better numerical algorithms, to connection to hardware (locally or remotely). Written in Java and originally designed to generate Java simulations, EjsS is an example of how ideas can and must evolve when technology changes. Figure 9 shows the user interface of EjsS.

Three years ago, together with Félix J. García Clemente from the University of Murcia, we started adding to the then called EJS (Easy Java Simulations) a new Javascript *flavor* that generates HTML5 + Javascript simulations. Our current development efforts are devoted almost entirely to the Javascript flavor, for precisely the reasons stated in the previous section.

EjsS has kept the main features that made it a very successful tool, used by hundreds of faculty and students, to create uncountable virtual and remote laboratories in Java. Easy Java/Javascript Simulations now also offer the same capabilities for the creation of Javascript simulations. EjsS:

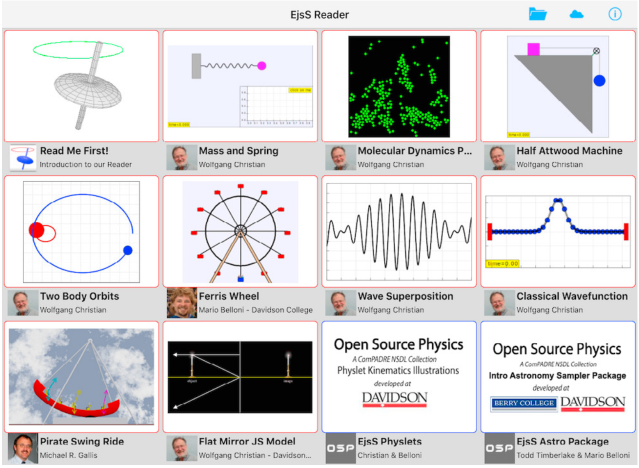


Fig. 10. The EjsS Reader App allows keeping a personal library of EjsS Javascript simulations for off-line use.

- Has a simple, yet powerful architecture for creating the model of a simulation. Authors follow a left-to-right sequence of panels where they code the logic of the simulation. This architecture is at the heart of more than 700 different simulations.
- Includes a powerful and easy to use solver for ordinary differential equations with support for events and delay differential equations.
- Provides writing HTML pages for documenting your simulation.
- Connects directly to on-line digital libraries for downloading the source code of existing simulations.
- Allows you to inspect and customize existing simulations for adaption and adoption.
- Generates advanced graphical and interactive interfaces, built by the author by using a drag and drop mechanism from a palette of existing view elements.
- Provides access to third party libraries in a simplified, object-oriented form.

Moreover, the Javascript flavor of EjsS generates documentation and user interfaces based solely on Web standards. Therefore, simulations created with EjsS can be easily included in Web pages or in EPUB documents that can be run and read on virtually any device. Besides, we have also created iOS and Android *Reader Apps* that can be used to keep a library of EjsS-generated simulations to be run off-line.[1] See Figure 10.

Security features of Web technologies prevent web pages (and the simulations in it) to access hardware connected to the host computer or tablet. This limitation is currently the subject of our research as we try to determine the most appropriated way to connect and send and receive data to and from equipment. Though there is still research and development going on in this topic, our current direction points to the WebSocket and Smart Device specifications.

The WebSocket specification includes a Javascript interface to establish a full-duplex single socket connection between equipment, and is part of the HTML5 initiative (Websocket (2015)). The WebSocket application programmer interface in the Web IDL (an interface definition language to describe interfaces to be implemented in

---

[1] Search for "EjsS" in Apple's iTunes Store or Google's Play Store.

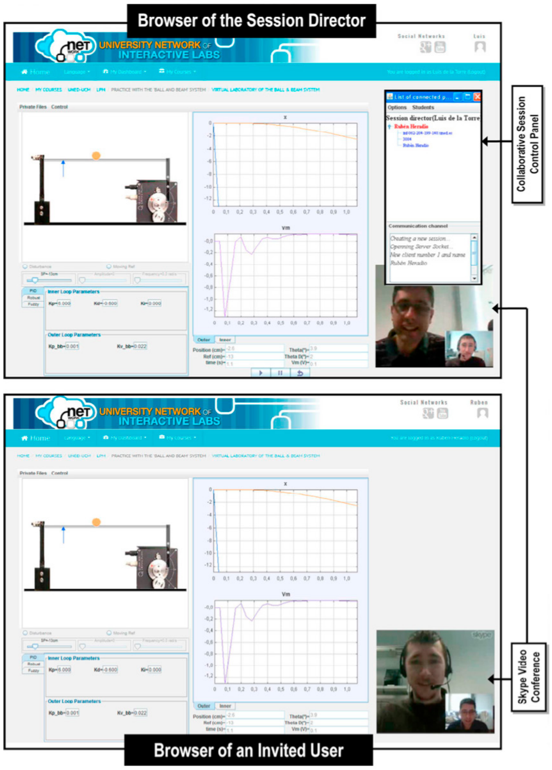


Fig. 11. A collaborative session of a UNILabs laboratory.

web browsers) is being standardized by the W3C (W3C (2015b)).

The Smart Device specification defines a set of interfaces, protocols and data formats, for exchanging data between equipment (Salzmann et al. (2015)). The IEEE Working Group P1876 on Networked Smart Learning Objects for Online Laboratories is currently working on establishing the Smart Device specification as an IEEE standard.

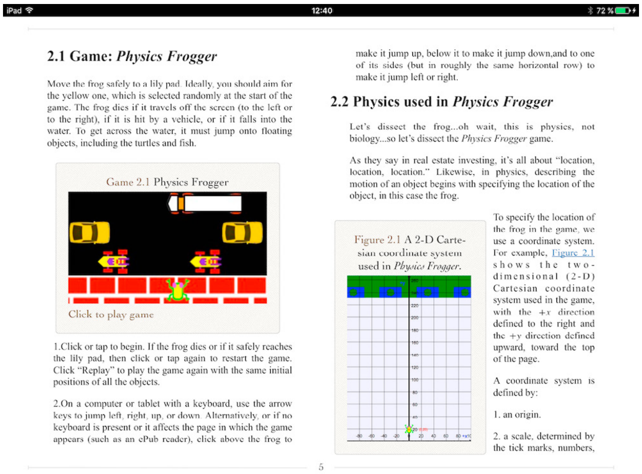
5.2 Moodle plug-ins

For the creation of educational courses, reseachers of the Spanish group at UNED (Madrid) have created Moodle plug-ins that allow easy integration of EjsS-generated virtual and remote laboratories in Moodle courses. This is possible for both Java (Heradio et al. (2014)) and, more recently, Javascript laboratories.

This group has also developed, together with researchers from the University of Huelva (Spain), Moodle extensions for handling the connection to the equipment (Márquez-Sánchez (2015)), and for facilitating collaborative support for laboratories created with EjsS (de la Torre et al. (2013), de la Torre et al. (2015)). Figure 11 shows a collaborative session of a UNILabs laboratory within Moodle.

5.3 EPUB edition

Creating EPUB (also written ePub) documents is not a trivial thing, but there exist free (as well as comercial) tools to assist with the task. Authors can create standard (text plus static images) ePubs using different free tools, such as Sigil (Sigil (2015)) or Calibre (Calibre (2015)). Or



2.1 Game: *Physics Frogger*

Move the frog safely to a lily pad. Ideally, you should aim for the yellow one, which is selected randomly at the start of the game. The frog dies if it travels off the screen (to the left or to the right), if it is hit by a vehicle, or if it falls into the water. To get across the water, it must jump onto floating objects, including the turtles and fish.

Game 2.1 Physics Frogger

Click to play game

1.Click or tap to begin. If the frog dies or if it safely reaches the lily pad, then click or tap again to restart the game. Click "Replay" to play the game again with the same initial positions of all the objects.

2.On a computer or tablet with a keyboard, use the arrow keys to jump left, right, up, or down. Alternatively, or if no keyboard is present or it affects the page in which the game appears (such as an ePub reader), click above the frog to make it jump up, below it to make it jump down,and to one of its sides (but in roughly the same horizontal row) to make it jump left or right.

2.2 Physics used in *Physics Frogger*

Let's dissect the frog...oh wait, this is physics, not biology...so let's dissect the *Physics Frogger* game.

As they say in real estate investing, it's all about "location, location, location." Likewise, in physics, describing the motion of an object begins with specifying the location of the object, in this case the frog.

Figure 2.1 A 2-D Cartesian coordinate system used in *Physics Frogger*.

To specify the location of the frog in the game, we use a coordinate system. For example, Figure 2.1 shows the two-dimensional (2-D) Cartesian coordinate system used in the game, with the $+x$ direction defined to the right and the $+y$ direction defined upward, toward the top of the page.

A coordinate system is defined by:

1. an origin.

2. a scale, determined by the tick marks, numbers,

5

Fig. 12. An iBook with an embedded EjsS simulation. Double-clicking the *Game* vignette runs the simulation full screen.

they can convert documents created in other formats to ePub. Our favorite such conversion tool is Pandoc (Pandoc (2015)), which can convert LaTeX (LaTeX (2015)) documents to ePub with a more than reasonable quality.

Including virtual laboratories in ePub documents is an extra difficulty. The Javascript flavor of EjsS can create ePub 3 small documents with one or more simulations. A standard ePub editor can then be used to edit this initial ePub and either add extra narrative to it, or to combine it with a LaTeX + Pandoc generated ePub. This is actually the approach that we use ourselves.

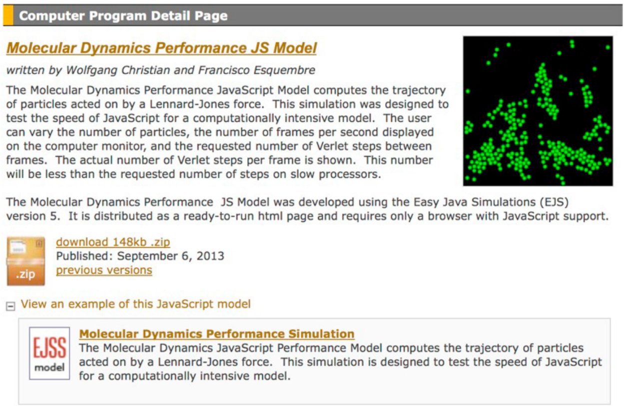
There are other, excellent ways of producing electronic books. Our favorite is Apple's iBooks Author (Apple (2015)) and we recently added to EjsS the capability of generating HTML5 Widgets that can be readily used by iBooks Author. Figure 12 shows an EjsS-generated widget embedded in an iBook (a proprietary format from Apple), running on an iPad. The quality of ePubs and iBooks created with this tool cannot be questioned. Unfortunately, electronic books generated with iBooks Author seem to only run on Mac OS X and iOS platforms.

6. COLLABORATION

From the very beginning, we decided that we would make our work freely available to all interested teachers and researchers. This decision has had implications in the design of our tools, requiring sometimes extra efforts in their implementation. But no other decision has been more rewarding than this one.

We decided that all our ready-to-use simulations would be distributed freely (for non-commercial purposes) and together with its complete source code in XML form. This code can be edited with Easy Java/Javascript Simulations for inspection and adaptation. As far as we know, no other similar project does this. The fact that EjsS has a standard architecture, common to all simulations, makes it feasible for a teacher to inspect another teacher's implementation of a given model.

We teamed up with the ComPADRE digital library to offer an on-line collection of OpenSourcePhysics simulations for



Computer Program Detail Page

Molecular Dynamics Performance JS Model

written by Wolfgang Christian and Francisco Esquembre

The Molecular Dynamics Performance JavaScript Model computes the trajectory of particles acted on by a Lennard-Jones force. This simulation was designed to test the speed of JavaScript for a computationally intensive model. The user can vary the number of particles, the number of frames per second displayed on the computer monitor, and the requested number of Verlet steps between frames. The actual number of Verlet steps per frame is shown. This number will be less than the requested number of steps on slow processors.

The Molecular Dynamics Performance JS Model was developed using the Easy Java Simulations (EJS) version 5. It is distributed as a ready-to-run html page and requires only a browser with JavaScript support.

download 148kb .zip
Published: September 6, 2013
previous versions

⊟ View an example of this JavaScript model

Molecular Dynamics Performance Simulation
The Molecular Dynamics JavaScript Performance Model computes the trajectory of particles acted on by a Lennard-Jones force. This simulation is designed to test the speed of JavaScript for a computationally intensive model.

Fig. 13. A simulation entry in the OSP collection of ComPADRE. Entries include the source code and, frequently, curricular material for using the simulation in classroom.

Physics. Figure 13 shows an entry in ComPADRE with a simulation created with EjsS. The collection has had great acceptance and was awarded the SPORE award by the Science Magazine in November 2011 (Science (2011)). The collection hosts nearly 1000 entries between Java simulations and associated curricular material and a growing number of Javascript simulations. All these ready-to-run simulations can be accessed on-line through the library services of ComPADRE or from the Reader apps. Alternatively, the library can also be accessed directly from EjsS to download the source code in order to inspect, modify, and re-run the simulations.

Institutions not affiliated with ComPADRE, or editors of small collections of simulations, can also establish their own smaller-scale digital library. Adding a simple PHP script – that we provide freely – to their web servers is all that is needed to turn their collection into a digital library that can be accessed by EjsS or our Reader apps.

These interconnections set up an EjsS collaboration environment in which instructors can access a library of simulations, communicate with others (the ComPADRE digital library provides communication tools), and contribute to the community with their own work. Figure 14 shows the EjsS ecosystem for Physics instructors interested in creating and using simulations.

The AutomatL@abs and its descendent projects are the best examples of collaboration of different organizations to create a collective set of virtual and remote laboratories in the field of Control Engineering. The Control Engineering community of EjsS users is growing considerably, but still lacks the organizational support provided by a well-established digital library, such as ComPADRE does for Physics.

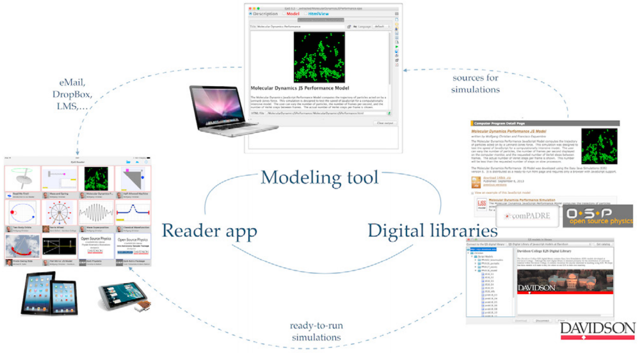


Fig. 14. The EjsS ecosystem for instructors includes a modelling tool, deployment and collaboration tools.

Finally, the open nature of our work has blessed us with the possibility of collaborating with a large number of researchers and instructors from all over the world. Two additional and excellent examples of these collaborations are the collection of EjsS simulations included in the pioneering work of Fu-Kwun Hwang in Taiwan (Hwang (2015)) and the growing, more recent, digital library of EjsS simulations lead by Loo Kang Wee in Singapore (Wee (2015)). This work certainly influenced UNESCO Bangkok, Asia and Pacific Regional Bureau for Education, to select Easy Java/Javascript Simulations to be part of the free educational compilation "UNESCO Directory of Free Educational Resources for Teachers. Vol. 1: Science" for distribution worldwide, mostly in the Asia-Pacific region targeting developing countries, in March 2015.

7. CONCLUSION

We have reviewed the main lessons we learned in the past ten years of creating and using virtual and remote laboratories for improving Science and Engineering teaching and learning, and of choosing, adapting, and developing tools for the creation of these laboratories. Designing and using the laboratories with solid, research-based pedagogical approaches, choosing the right tools and following well-established standards, and being open to collaboration are solid grounds on which to base successful work in this area. But on top of these lessons, the final one outstands: concentrate of good, long-lasting designs and teaching ideas. The use of current technologies and the Internet as we perceive it today, the implementation of a diversity of state-of-the-art pedagogical approaches, and the use of present and upcoming software and hardware platforms may influence our current work on virtual and remote laboratories. But the final goal of improving Science and Engineering education in the mid and long term must be our driving force.

We have also shown how we applied these lessons to our own work, at the same time we were learning them. Experience taught us what worked and what didn't, and lead our steps in this journey. It has also set the path for our future work.

The collection of tools available for the creation of virtual laboratories is already very powerful for generating a wide range of simulations. From simple – but perhaps perfectly appropriated for the first steps of instruction – simulations, to more sophisticated virtual laboratories required in higher education. Javascript is a flexible programming language and is very well integrated with HTML and other W3C standards. There may still be a need, however, for specialized or dedicated Javascript libraries to solve particular modelling or GUI tasks.

The creation of remote laboratories needs to deal also with the issues of communication and interfacing with actual hardware. There exists a new generation of very affordable hardware and data acquisition equipment for even the modest pockets, such as Arduino, RedPitaya and NI MyRio boards (to name a few), and the numerous cheap sensors and actuators that can be connected to them. Some of these boards also offer the possibility of establishing their own Web access point, so all that is needed to connect to them is a WiFi enabled device (that is, practically any computing device, including mobile ones). We find this a very attractive possibility. In order to deal with the diversity of new hardware platforms, it would be highly desirable the creation of standards for interfacing with them. The Smart Device specification is a promising step in this direction.

Finally, the integration of virtual and remote laboratories in actual curricular material that can be effectively deployed and used in class, group and homework is an area where some work may be needed. Besides using Internet standards, research and development work may be needed to establish pedagogical and integration standards that allow easy creation of learning environments that integrate classical curricular materials, such as narrative and evaluation tools, with other more modern ones, such as collaborative work and social networking.

Our own work will surely follow these lines.

REFERENCES

Apple. iBooks Author web page. `www.apple.com/es/ibooks-author`, 2015. Accessed: 2015-09-06.

Calibre. Calibre eBook management. `calibre-ebook.com`, 2015. Accessed: 2015-09-07.

Wolfgang Christian. Open Source Physics project. `www.compadre.org/osp`, 2015. Accessed: 2015-09-06.

Wolfgang Christian and Mario Belloni. *Physlets: Teaching Physics with Interactive Curricular Material.* Addison-Wesley, 2000. ISBN 0130293415.

Wolfgang Christian and Mario Belloni. *Physlet Physics.* Prentice Hall, 2003. ISBN 0-13-101969-4.

Wolfgang Christian and Mario Belloni. Creating and distributing simulations for tablets. *Poster at the Conference in Computational Physics, Boston 2014*, 2014.

Wolfgang Christian, Francisco Esquembre, and Lyle Barbato. Open Source Physics. *Science*, 334:1077–1078, 2011.

Wolfgang Christian, Mario Belloni, Francisco Esquembre, Bruce A. Mason, Lyle Barbato, and Matt Riggsbee. The Physlet approach to simulation design. *The Physics Teacher*, 53:419–422, 2015.

Luis de la Torre, Rubén Heradio, Carlos A. Jara, José Sánchez, Sebastián Dormido, Fernando Torres, and Francisco A. Candelas. Providing collaborative support to virtual and remote laboratories. *IEEE Transactions on Learning Technologies*, 6(4):312–323, 2013.

Luis de la Torre, María Guinaldo, Rubén Heradio, and Sebastián Dormido. The ball and beam system: a case study of virtual and remote lab enhancement with Moodle. *IEEE Transactions on Industrial Informatics*, 11:934–945, 2015.

Sebastián Dormido. Control learning: Present and future. *Annual Reviews in Control*, 28:115–136, 2004.

Sebastián Dormido and Francisco Esquembre. The quadruple-tank process: An interactive tool for control education. *Proceedings of the European Control Conference*, 2003.

Sebastián Dormido, José Sánchez-Moreno, Hector Vargas, Luis de la Torre, and Rubén Heradio. UNED Labs: a network of virtual and remote laboratories. In J. G. Zuba and G. Alves, editors, *Using Remote Labs in Education*, volume 2, pages 253–270. University of Deusto Publications, Deusto, 3rd edition, 2011.

Natividad Duro, Raquel Dormido, Héctor Vargas, Sebastián Dormido-Canto, José Sánchez, Gonzalo Farias, Sebastián Dormido, and Francisco Esquembre. An integrated virtual and remote control lab: The three-tank system as a case study. *Computing in Science & Engineering*, July:50–59, 2008.

Francisco Esquembre. Easy Java Simulations: a software tool to create scientific simulations in java. *Computer Physics Communications*, 156, 2:199–204, 2004.

Francisco Esquembre. Laboratorio de Modelización. `www.um.es/fem/PersonalWiki`, 2015. Accessed: 2015-09-06.

Francisco Esquembre, Ernesto Martín, Wolfgang Christian, and Mario Belloni. *Fislets, Enseñanza de la Física con Material Interactivo.* Prentice Hall, 2004. ISBN 8420537810.

Google. Chrome help. `support.google.com/chrome/answer/6213033`, 2015. Accessed: 2015-09-07.

Richard Hake. Interactive-engagement vs. traditional methods: A six-thousand-student survey of mechanics test data for introductory physics courses. *American Journal of Physics*, 66:64–74, 1998.

Rubén Heradio, Luis de la Torre, José Sánchez, and Sebastián Dormido. Making EJS applications at the OSP digital library available from Moodle. In *Remote Engineering and Virtual Instrumentation (REV), 2014 11th International Conference on*, pages 112–116. REV, Porto, 2014.

Fu-Kwun Hwang. NTNUJAVA virtual physics laboratory. `www.phy.ntnu.edu.tw/ntnujava`, 2015. Accessed: 2015-09-08.

IDFP. International Digital Publishing Forum web page. `idpf.org`, 2015. Accessed: 2015-09-07.

William Junkin, Wolfgang Christian, and Mario Belloni. Designing courses with Moodle workshop. `www.opensourcephysics.org`, 2015. Accessed: 2015-09-07.

Rubin H. Landau, Manuel José Páez, and Cristian C. Bordeianu. *A Survey of Computational Physics.* Princeton University Press, 2008. ISBN 9780691131375.

LaTeX. LaTeX a document preparation system. `www.latex-project.org`, 2015. Accessed: 2015-09-07.

Marco A. Márquez-Sánchez. *Un Modelo General de Referencia para el Acceso Remoto a Laboratorios Docentes y de Investigación (Ph. D. dissertation).* Universidad de Huelva, 2015.

Bruce Mason. ComPADRE digital library. `www.compadre.org`, 2015. Accessed: 2015-09-06.

Desiré Medina, Isabel Muñoz, Cristina López, Catiana Meca, and Francisco Esquembre. Synkope simulation. `www.opensourcephysics.org`, 2014. Accessed: 2015-09-21.

Moodle. Moodle web page. `www.moodle.org`, 2015. Accessed: 2015-09-07.

Gregor M. Novak, Evelyn T. Patterson, Andrew D. Gavrin, and Wolfgang Christian. *Just-in-Time Teaching: Blending Active Learning with Web Technology.* Prentice Hall, Upper Saddle River, 1999. ISBN 0130850349.

Pandoc. Pandoc a universal document converter. `pandoc.org`, 2015. Accessed: 2015-09-07.

Edward Redish. Preface. In *Christian and Belloni (2000).* Prentice-Hall, 2000.

Jacobo Sáenz, Jesús Chacón, Luis de la Torre, Antonio Visioli, and Sebastián Dormido. Open and low-cost virtual and remote labs on control engineering. *IEEE Access*, 3:805–814, 2015.

Christophe Salzmann, Sten Govaerts, Wissam Halimi, and Denis Gillet. The Smart Device specification for remote labs. In *Remote Engineering and Virtual Instrumentation (REV), 2015 12th International Conference on*, pages 199–208. REV, Bangkok, 2015.

Science. Science prize for online resources in education (SPORE) award. `www.sciencemag.org/site/special/spore`, 2011. Accessed: 2015-09-07.

Sigil. Epub eBook editor. `sigil-ebook.com`, 2015. Accessed: 2015-09-07.

Alex Simonelis. A concise guide to the major internet bodies. *Magazine Ubiquity*, February:2–2, 2005.

UNILabs. University network of interactive labs. `unilabs.dia.uned.es`, 2015. Accessed: 2015-09-06.

Héctor Vargas, José Sánchez, Carlos A. Jara, Francisco A. Candelas, Fernando Torres, and Sebastián Dormido. A network of automatic control web-based laboratories. *IEEE Transactions on Learning Technologies*, 4(3):197–208, 2011.

W3C. World Wide Web consortium. `www.w3.org`, 2015a. Accessed: 2015-09-07.

W3C. Web IDL. `www.w3.org/TR/WebIDL/`, 2015b. Accessed: 2015-09-20.

Websocket. `www.websocket.org`, 2015. Accessed: 2015-09-07.

Loo Kang Wee. Open Source Physics @Singapore. `weelookang.blogspot.com.es`, 2015. Accessed: 2015-09-08.