

EVOLUCIÓN DEL WEBLAB DE LA UNIVERSIDAD DE DEUSTO

J. GARCÍA ZUBÍA, D. LÓPEZ DE IPIÑA, U. HERNÁNDEZ JAYO, P. ORDUÑA, I. TRUEBA
Dpto. Arquitectura de Computadores, Automática y Electrónica Industrial. Universidad de
Deusto. Bilbao (Spain) zubia@eside.deusto.es

La comunicación presenta la evolución del WebLab-Deusto en estos dos últimos años. El trabajo presenta el estado actual y las mejoras en el diseño software y hardware del WebLab-Deusto. Además se presenta el resultado de su aplicación real en dos asignaturas de los cursos 2004/2005 y 2005/2006.

1. Introducción

La sociedad occidental de hoy en día no tiene nada que ver con la de tiempos pasados en aspectos fundamentales como el tipo de vida familiar, la integración de personas discapacitadas, la organización de las tareas, la aplicación de nuevas tecnologías tanto en el ocio como en el trabajo, el aprendizaje a distancia, la distribución del tiempo, etc. Todos estos aspectos hacen necesario que cambiemos nuestros hábitos de vida. La universidad debe vivir este cambio y tratar de ofrecer los servicios que la sociedad demanda en cada momento.

Siempre ha sido un objetivo de las universidades el descentralizar parte de sus actividades y facilitar los grupos de trabajo o el trabajo colaborativo: llevar la universidad en cualquier momento a cualquier lugar. Además, la Unión Europea está teniendo en cuenta estos aspectos al amparo de la Declaración de Bolonia. En este nuevo marco educativo, los estudiantes tendrán más tiempo para organizarse el tiempo, la educación será menos estricta en cuanto a lo que horarios se refiere, y consecuentemente, la organización de los laboratorios se deberá adaptar a dichos cambios, siendo por tanto su organización más complicada.

El concepto de WebLab apareció a principio de los noventa. Su desarrollo está extendido en laboratorios de electrónica analógica [1] y digital [2], lógica programable [3] o procesos de control [4]. Podemos encontrar buenos ejemplos de WebLabs en diferentes países: USA [5], Colombia [6], España [3,7], Portugal [8], Italia [9], Corea [4], etc.

Un WebLab puede ser estudiado desde diferentes puntos de vista:

- Didáctico: logros didácticos, calidad y conformidad del WebLab, integración en una plataforma didáctica, etc. [9,10,11,12]
- Tecnología hardware: tarjetas, prototipos electrónicos, adquisición de datos, etc. [6,10]
- Tecnología software: diseño cliente/servidor, seguridad, integración, etc. [10]
- Plataforma de desarrollo software: Servicios-Web [4], LabView [2,13], aplicaciones en C [14], JAVA [15], Matlab [8], etc.
- Comunicación: mediante RS-232 [16], TCP/IP [11], XML [17], etc.
- Social: solidaridad internacional, adaptación de gente incapacitada, etc. [3].

La reciente popularidad del concepto WebLab en el mundo académico, los diferentes acercamientos de las universidades y las abundantes referencias existentes, no son más que pruebas de la gran actividad que rodea a esta idea que está llamada a ser una piedra angular en la enseñanza de la ingenierías a través de World Wide Web.

El diseño de un WebLab debe conjugar de forma efectiva hardware y software. Los WebLabs son diseñados generalmente por ingenieros electrónicos y de control que, como es natural, presentan mayor atención al lado hardware del sistema. Normalmente siguen tres pasos en el proceso: (1) elegir un dispositivo programable, (2) conectarlo a un servidor que sea accesible a través de web o mediante

un simple socket TCP/IP, y (3) diseñar un protocolo sencillo para grabar permitir grabar programas en el dispositivo remoto, enviar entradas y recibir las salidas. Desafortunadamente, al desarrollo software involucrado en los últimos dos pasos no se le presta la suficiente atención y por lo tanto se hace un uso muy pobre del hardware programable disponible. Nosotros creemos que si se plantean unas arquitecturas software mejoradas para los WebLabs, conseguiremos que tengan una apariencia más amigable, que presenten unos costos más eficientes y que serán por tanto más reales y escalables.

Por otra parte, los WebLab suelen ser diseñados pero rara vez suelen ser puestos a servicio del alumno y de la comunidad investigadora. Y además, aquellos que son usados, rara vez son contrastados en su calidad y resultados didácticos. El WebLab desarrollado en la Facultad de Ingeniería de la Universidad de Deusto es usado en dos asignaturas: Lógica Programable y Diseño Electrónico en 3º Ing. Técnica Industrial Esp. Electrónica Industrial y 5º de Ing. Automática y Electrónica Industrial, y en cada curso los alumnos responden a una encuesta para valorar el servicio ofrecido.

La estructura de este artículo es el siguiente: en los apartados 2 y 3 se revisará en concepto de WebLab y sus ventajas. En los apartados 4 y 5 se describe el WebLab-Deusto y se muestra las evoluciones de la arquitectura software de nuestro WebLab y propone un nuevo modelo de arquitectura “ideal” para WebLabs que permite entre otras cosas el trabajo colaborativo. El apartado 6 muestra la opinión que nuestros alumnos tienen sobre el WebLab-Deusto y el apartado 7 presenta finalmente las conclusiones.

2. ¿Qué es un WebLab?

Actualmente, las prácticas que realizan los estudiantes, se llevan a cabo en laboratorios, con los problemas que ello acarrea en cuanto a organización y equipamiento especializado. Para solventar estos problemas, se dispone de dos soluciones frecuentemente usados: simuladores y laboratorios virtuales que intentan reproducir ante el alumno un laboratorio real, pero que son solamente la representación software del mismo.

Hay una tercera solución: los llamados WebLabs o laboratorios remotos. Un WebLab es una aplicación con la que un alumno, a través de una página Web puede interactuar de forma remota con los diferentes elementos que puedan localizarse en un laboratorio: dispositivos programables (FPGA, CPLD, DSP, PIC, etc), instrumentación (generadores de señales, analizadores de espectros, osciloscopios, etc), u otros elementos bajo control (robots, maquetas de control PID, etc).

Por tanto, un WebLab no es ni un simulador ni solamente un sistema software: el usuario mediante TCP/IP, tiene comunicación con el hardware real que hay en el laboratorio. De esta forma se permite al usuario acceder, mediante Internet, al laboratorio desde cualquier parte en cualquier momento en vez de tener que acudir a las instalaciones de la universidad.

3. Beneficios de los WebLabs.

El diseño y el uso de un WebLab en una facultad de ingeniería proporciona los siguientes beneficios :

- Mejor aprovechamiento de los equipos del laboratorio, ya que están disponibles para los alumnos las 24 horas del durante los 365 días del año.
- Organización de los laboratorios. No es necesario mantener los laboratorios abiertos todo el tiempo, solo es necesario mantener operativo el WebLab.
- Organización del trabajo del estudiante. Haciendo uso del WebLab, tanto el alumno como el profesor pueden organizar mejor sus tiempos, incluyendo los tiempos de clase.
- Aprendizaje autónomo. Los WebLabs promueven el autoaprendizaje, fundamental en el nuevo espacio europeo universitario.
- Acercamiento a la sociedad. Los WebLabs acercan los laboratorios al resto de la sociedad.

- Cursos a distancia. Los WebLabs permiten la organización de cursos de ingeniería a distancia sin tener la necesidad de que los alumnos se encuentren presentes físicamente en el aula, evitando muchos de los problemas actuales.
- Integración de estudiantes discapacitados. Dado que todos los equipamientos hardware están controlados por un ordenador, pueden ser usados por estudiantes incapacitados utilizando software diseñado especialmente para sus necesidades particulares.
- Máximo rendimiento de las instalaciones de la universidad. El uso de WebLabs se evitan los tiempos muertos en los que el laboratorio no está siendo utilizado.
- Beneficios económicos. Evitando los tiempos muertos, se pueden obtener los mismos resultados minimizando la inversión necesaria en hardware. Y por otra parte, se evita el coste que conlleva el mantenimiento de los laboratorios tradicionales.

4. Descripción del WebLab-Deusto

El WebLab-Deusto (Fig.1) lleva cinco años desarrollándose en el Dpto. De Arquitectura de Computadores, Automática y Electrónica Industrial y Telecomunicaciones de la Universidad de Deusto. El WebLab desarrollado permite el acceso y control remoto vía Internet de dos dispositivos lógicos programables, CPLD (XC9572PLCC44) y FPGA (XC2SPQ144). De esta forma el alumno puede completar sus prácticas desde casa, sin necesidad de acudir al laboratorio en la facultad. Las ventajas son claras y evidentes, tanto para el alumno como para el profesor, y tanto para la universidad y la facultad como para la propia sociedad.

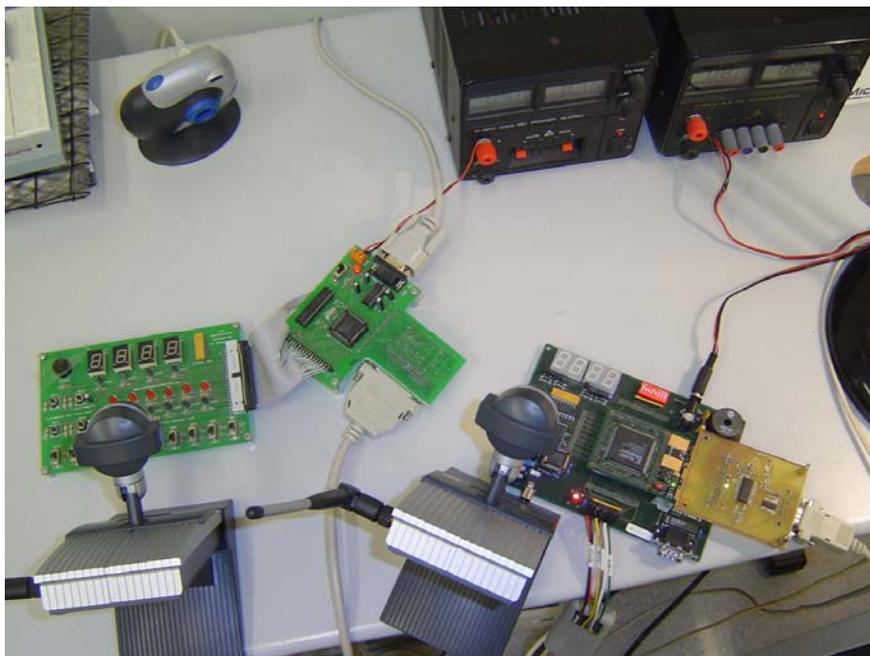


Figura 1. Imagen del WebLab-Deusto

La secuencia característica de una práctica comienza con el alumno en su casa programando y simulando en VHDL. Cuando está seguro de lo programado, genera el fichero .jed o .bit. Seguidamente entra en la página web del WebLab para descargar el fichero generado sobre el PLD elegido. En este momento, el servidor ofrece al cliente/alumno una página para controlar las entradas (interruptores, reloj, pulsador, etc.) y una imagen Webcam para observar las salidas. Si lo visto por el alumno es satisfactorio, la práctica ha finalizado, en caso contrario deberá volver al VHDL y a la secuencia anterior.

5. Evolución del WebLab-Deusto

La arquitectura software de nuestro WebLab ha ido evolucionando a través de las siguientes versiones:

1. Solución propietaria basada en un Socket y un Applet [3].
2. Solución Web [11].
3. Solución Web basada en AJAX [11].
4. Solución Web basada en AJAX y en microservidores.

5.1 Solución propietaria basada en un Socket y un Applet

Esta fue la primera versión de la arquitectura software del WebLab-Deusto (Fig. 2). Consistía en un cliente desarrollado en C y la comunicación se realizaba mediante un simple socket usando la librería SDLnet en el servidor. El servidor, mediante RS-232, se comunicaba con el PIC, el cual actuaba de puente con el dispositivo programable, en este caso un PLD. En paralelo con lo aplicación de línea de comandos que controlaba remotamente el dispositivo programable, se usaba un applet ActiveWebCam de Pysoft, para observar en tiempo real el estado del dispositivo. En cada momento solo podía acceder un usuario al dispositivo remoto. Éste era un prototipo solo accesible por invitados.

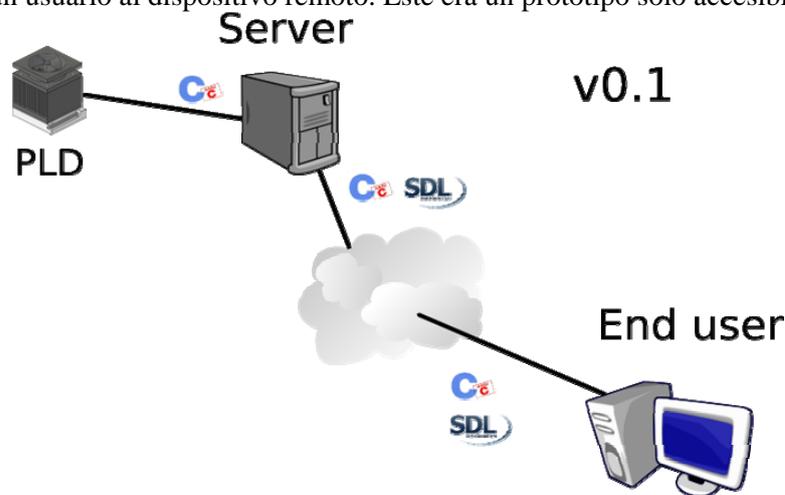


Figura 2. Primera versión de la arquitectura software

Las principales desventajas de esta solución eran:

- Interoperatividad. Tanto las soluciones adoptadas en el cliente como en el servidor, únicamente podían ser ejecutadas bajo Microsoft Windows.
- Entorno. El usuario necesitaba arrancar dos aplicaciones diferentes, la aplicación de control basada en C y el applet de visualización de Java. Además, el control ofrecido al cliente era un primitivo interfaz de comandos.
- Seguridad. En el lado del servidor, el firewall debía ser configurado para permitir el tráfico sobre dos puertos no well-known en lugar de utilizar puertos ya abiertos como el 80 para HTTP. En consecuencia, no había un desarrollo para el control de usuarios y por tanto, el WebLab no estaba abierto al público, y solo estaba disponible para demostraciones dentro de la LAN de la universidad.

5.2 Solución Web

La Fig. 3 muestra la segunda versión de nuestra arquitectura hardware. Aquí, el lado del servidor está compuesto por tres elementos: a) un servidor web Apache que contiene una aplicación web controlando y visualizando applets, b) un servidor Python que se comunica a través del puerto serie con el PIC que controla el PLD y c) un servidor de broadcast para enviar las imágenes de la Webcam. En esta versión, la aplicación del cliente está desarrollada totalmente en JAVA, accesible a través de un navegador web con un plug-in de Java preinstalado.

La lógica del servidor ha sido actualizada para permitir el acceso de usuarios y el control de acceso. En cada momento solo puede haber un usuario accediendo al dispositivo remoto durante un tiempo máximo de 120 segundos. El único requerimiento que debían satisfacer los estudiantes era tener preinstalado el plug-in de Java.

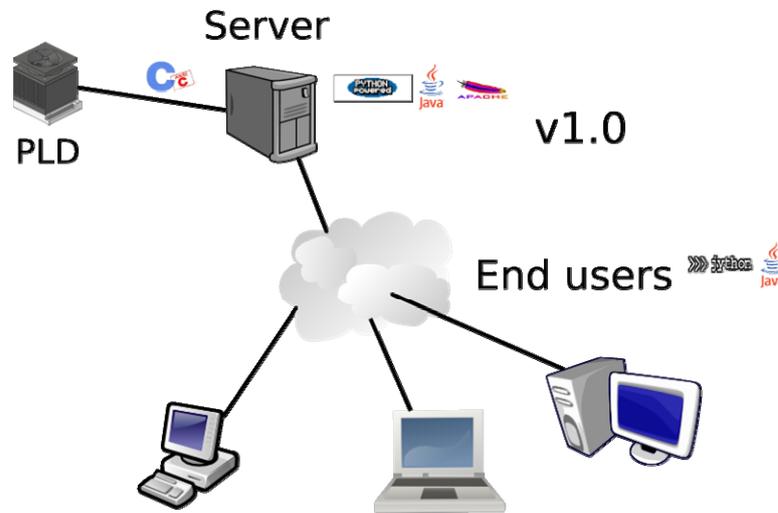


Figura 3. Segunda versión de la arquitectura software

Esta solución aún presentaba ciertos inconvenientes en cuanto al entorno y la seguridad:

- Entorno. Teníamos dos applets independientes ejecutándose en la misma página web. La descarga de los applets llevaba un tiempo y requerían tener instalado el plug-in de Java, con el inconveniente de que muchos usuarios no aceptan la instalación del mismo.
- Seguridad. Cada vez que el usuario se descargaba el applet de control para cargar el fichero con la nueva lógica a descargar al PLD, saltaba una alarma de seguridad. Además aún teníamos que mantener dos puertos abiertos en el firewall: uno para el servidor de la webcam y otro para el control del servidor. Esto suponía un inconveniente para el mantenimiento del firewall.

Con esta versión, ya se permitió el acceso al sistema desde fuera de la universidad a los estudiantes de la asignatura de “Lógica Programable”.

5.3 Solución Web basada en AJAX

La tercera versión del WebLab-Deusto, actualmente en uso, se muestra en la Fig. 4. A medida que las tecnologías web que hacen posible el desarrollo de WebLabs, nosotros tratamos de hacer evolucionar nuestro WebLab-Deusto. De esta forma surge la versión 2.0 desarrollada en tecnologías Web 2.0 como AJAX, Python, y SOAP.

La aplicación mostrada en el navegador del cliente se comunica con el servidor mediante HTTP. Ahora tenemos una solución web con un sistema de seguridad firewall programada en AJAX (Asynchronous JavaScript and XML). El principal beneficio de este desarrollo web es que solo utiliza herramientas disponibles en cualquier navegador web, como por ejemplo XHTML, DOM y JavaScript. De este modo, no se requiere la instalación de ningún plug-in en el navegador del usuario.

El servidor está compuesto por los siguientes elementos: a) un servidor Java que está continuamente capturando y salvando imágenes de la WebCam en un directorio exportado mediante un servidor web Apache, b) un servidor Python controlando la comunicación con el dispositivo programable y c) un aplicación ASP.Net basada en Mono y ejecutándose en el servidor web Apache ofreciendo un interfaz de servicios web para las aplicaciones del cliente.

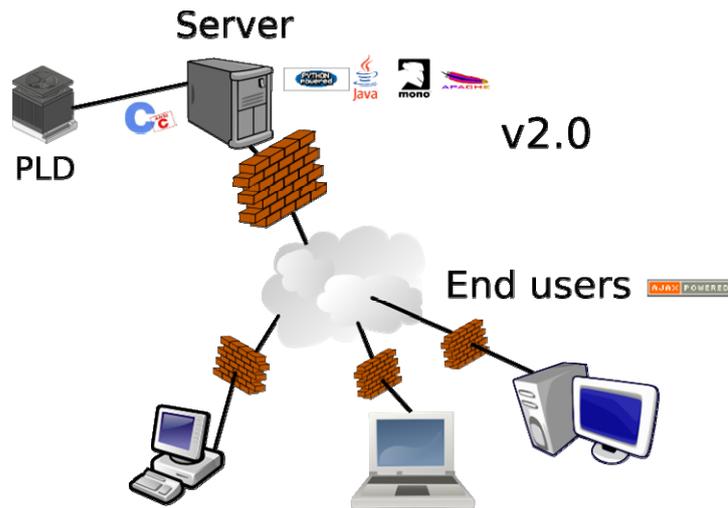


Figura 4. Tercera versión de la arquitectura software

La aplicación del cliente ahora es una solución puramente HTML/JavaScript que sigue el modelo de interacción web de AJAX, como por ejemplo en el caso de que cambie el contenido de una parte de la página web, en vez de cambiar toda la página, únicamente cambia la parte modificada, siendo una interacción entre el cliente y el servidor. Esta tecnología está siendo utilizada de manera muy satisfactoria en aplicaciones tales como Gmail, Google Maps o Flickr. La clave de esta tecnología es que los comandos de control, las respuestas y las imágenes son transmitidas de forma asíncrona, sin interrumpir la interacción del usuario con el sistema, por medio de un objeto JavaScript XMLHttpRequest [18].

El intercambio de datos entre el cliente AJAX y el servidor basado en Mono se realiza sobre el estándar del protocolo de transporte de Servicios Web llamado SOAP. El servidor basado en tecnología Mono delega la invocación de los métodos que llegan al servidor Python que controla el dispositivo programable. La última imagen capturada está siendo continuamente recuperada mediante HTTP por el cliente con tecnología AJAX accediendo a una URL well-known.

Los principales inconvenientes de esta solución son:

- Mantenimiento del software del servidor. Hay demasiadas tecnologías usadas en el servidor: Java, Python y ASP.NET. Para el mantenimiento sería interesante concentrar todas las funcionalidades en un único componente desarrollado en una única tecnología.
- Escalabilidad. El servidor proporciona únicamente servicio a un usuario que accede al dispositivo programable. Idealmente, nos gustaría trabajar con una red de N dispositivos controlables mediante el mismo servidor y permitir el acceso simultáneo de N usuarios.
- Flujo de imágenes. La recepción de las imágenes del dispositivo programable aún no es óptima. Cada imagen es enviada como un fichero JPEG en lugar de tratarse como un flujo continuo de imágenes, solución que nos dotaría de imágenes actualizadas en cada momento del dispositivo programable.
- Seguridad. Esta versión aún no realiza una verificación de los programas descargados en el dispositivo programable con el objeto de prevenir posibles daños en el mismo.

Por otro lado, el uso de AJAX presenta las siguientes ventajas frente a las alternativas anteriormente utilizadas:

- Es una herramienta multiplataforma que permite que las aplicaciones tanto del cliente como del servidor puedan ejecutarse sobre Windows y sobre Linux.
- Para la comunicación entre el cliente y el servidor únicamente se utiliza el puerto 80, por lo que se consigue una mayor seguridad en el firewall.
- El cliente no tiene que instalarse ningún plug-in y únicamente necesita un navegador web para interactuar con el WebLab, consiguiendo de este modo que el cliente sea un "thin-client".

- Permite la utilización de todos los servicios que proporciona la utilización SOAP.
- El WebLab es accesible desde cualquier dispositivo móvil como por ejemplo un teléfono móvil o una PDA sin tener que realiza ninguna modificación en el servidor.

5.4 Solución Web basada en AJAX y microservidores

Ahora estamos trabajando en la versión del WebLab-Deusto que se puede ver en la Fig. 5. esta solución estará basada como las anteriores en una plataforma web, con un firewall seguro, más escalable que las anteriores (permitirá acceder a más dispositivos programables) y posibilitará el trabajo colaborativo. En este caso será posible que exista un grupo de N clientes conectados desde diferentes plataformas, conectados a N dispositivos, cada uno de ellos identificado por la IP proporcionada por el microservidor al cual se encuentra conectado.

El adoptar microservidores hace que nuestro WebLab sea un sistema más flexible y escalable:

- El servidor WebLab deja utilizar la comunicación serie RS-232 para pasar a utilizar una comunicación basada íntegramente en HTTP, usando una codificación de datos estándar como XML.
- Los microservidores posibilitan la creación de una red LAN de dispositivos hardware conectados al WebLab. Para la comunicación con el servidor se podrá utilizar una red basada en Ethernet o bien una red Wireless.
- Los prototipos electrónicos conectados a los microservidores también serán capaces de intercambiar información entre ellos. La información no solo fluirá entre el prototipo y el servidor, sino que también podrá circular entre los prototipos gracias a los microservidores.

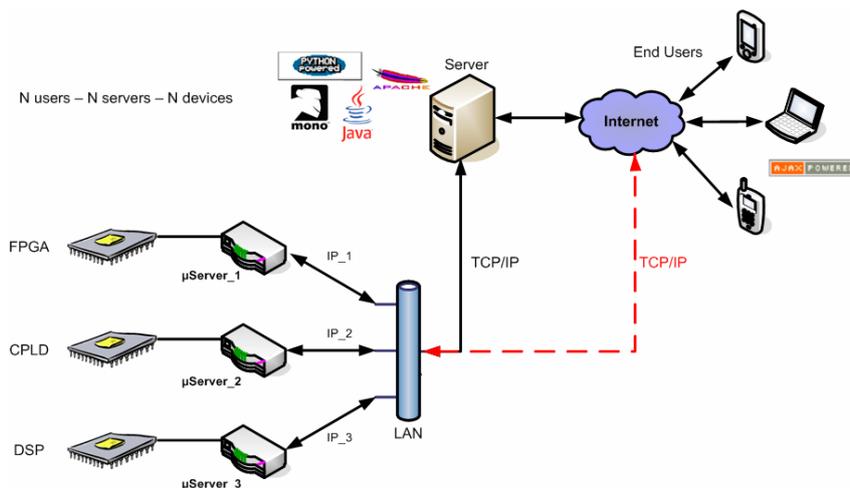


Figura 5. Cuarta versión de la arquitectura software

En un futuro, se podrá seguir teniendo un servidor como puerta de acceso al WebLab o bien las parejas dispositivo-microservidor, ya que son nodo identificado de una red, podrían estar conectadas directamente a Internet, paso indicado en la figura Fig. 5 mediante la línea discontinua. Esto último sería ir un paso más allá, de tal forma que en un futuro se podrían crear buscadores de hardware en Internet (por ejemplo, hardware-google), posibilitando que diferentes instituciones compartan los recursos de sus laboratorios. Por ejemplo, teniendo en consideración las diferencias horarias entre España y Sudamérica, el WebLab-Deusto podría ser accesible desde Sudamérica mientras en España es de noche. Pero para poder dar este paso, aún se deben valorar las ventajas y desventajas, así como las posibilidades que los microservidores proporcionan.

En conclusión, esta última versión propuesta, proporcionará el escenario necesario para el trabajo colaborativo siendo una plataforma multiusuario y multidispositivo, que maximiza el uso del hardware disponible en un laboratorio. En nuestra opinión, esta solución se aproxima a la arquitectura ideal de un WebLab.

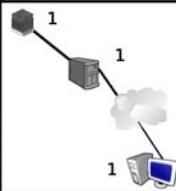
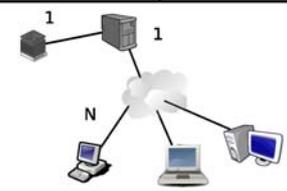
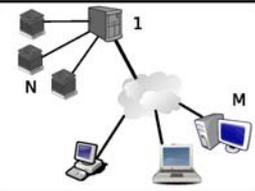
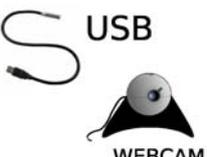
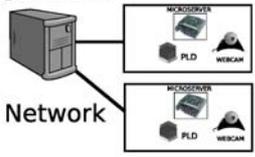
WebLab	0.1	1.0	2.0	3.0
Device Server Client Proportion				
Connection with devices	RS-232 	SERVER USB 	SERVER Network 	
Client side technology				
Server side technology				
Protocol	proprietary		SOAP	
Does it use HTTP for transporting everything?	No		Yes	
Data protection	-			

Figura 6. Evolución de tecnologías en el WebLab-Deusto

6. Resultados académicos

Actualmente el WebLab-Deusto (<http://weblab.deusto.es>, <http://weblab-pld.deusto.es>, <http://weblab-fpga.deusto.es>) se utiliza para llevar a cabo tareas sobre CPLD y FPGA (CPLD de Xilinx XC9572 y FPGA de Xilinx XC2S144). El WebLab-Deusto se utiliza en la asignatura de “Lógica Programable” en el tercer año de Ingeniería Automática y Electrónica y en “Diseño Electrónico” el quinto año de la misma carrera.

Si analizamos la Tabla 1, vemos las altas puntuaciones recogidas el primer año que se utilizó el WebLab-Deusto, posiblemente por su novedad y porque también el número de alumnos era menor. Actualmente la pregunta 2 no tiene mucho significado ya que hoy por hoy todos los alumnos utilizan el WebLab-Deusto, no como en años anteriores en el que su uso era opcional.

Vemos que para la mayoría de los usuarios encuestados es una buena idea el uso de este tipo de aprendizaje y relativamente sencilla su utilización, por lo que su inclusión como herramienta en otras

asignaturas les parece una buena iniciativa. En cambio, vemos que en el apartado de calidad, este año nuestra puntuación no es muy alta. Esto se debe a tres motivos principalmente: a) la nueva versión con la que se comenzó a trabajar, resultó al principio inestable, como todo cambio, b) se produjo un cambio en el ISE de Xilinx, por lo que se tuvieron que realizar cambios en el WebLab-Deusto, c) aunque ya han sido superados, al principio hubo ciertos problemas con las webcam, lo que entorpecía el uso del WebLab-Deusto. Hemos apreciado que uno de los aspectos más criticado ha sido el tiempo de conexión, por lo que para el próximo curso se ha decidido ampliarlo de 99 a 150 segundos por conexión.

Preguntas	Medida (1)	Medida (2)	Medida (3)
Número de accesos al WebLab	1.706	495	632
1. ¿Te ha ayudado el WebLab en la asignatura?	4.6	4.1	3.8
2. ¿Te has sentido beneficiado por estar en el grupo WebLab?	4.7	3.9	3.9
3. ¿Te parece una buena idea extenderlo a todos los alumnos?	4.7	4.6	4.2
4. ¿Es fácil de usar?	4.4	4.4	3.9
5. ¿Qué tal es la calidad de lo visto (la WebCam)?	3.2	2.4	2.7
6. ¿Te has sentido cómodo con la gestión de entradas?	3.7	3.1	3.0
7. ¿Qué te parece el tiempo asignado para cada conexión?	3.7	2.7	3.1
8. ¿Qué te parecen las entradas/salidas seleccionadas?	3.8	3.2	3.4
9. Al estar alejado de la maqueta, ¿has tenido sensación de control?	4.1	3.7	3.6
10. ¿Te gustaría usar WebLab en otras asignaturas?	4.3	4	3.9
11. ¿Cuál es tu satisfacción global con el WebLab?	4.7	3.9	3.7

(1) Resultados obtenidos en el curso 2004/2005 en la asignatura de "Lógica Programable"
(2) Resultados obtenidos en el curso 2005/2006 en la asignatura de "Diseño Electrónico"
(3) Resultados obtenidos en el curso 2005/2006 en la asignatura de "Lógica Programable"

Tabla 1. Resultados del cuestionario realizado entre los usuarios del WebLab-Deusto

7. Conclusiones y trabajo futuro

El trabajo aquí presentado proporciona principalmente tres contribuciones: a) es importante prestar más atención a la parte software que a la hardware a la hora de diseñar un WebLab, ya que muchos problemas que aparecen en el desarrollo de WebLabs se deben a un diseño software pobre (accesibilidad, seguridad, etc). b) El uso de microservidores en la parte hardware del sistema supondrá una revolución y una evolución en el desarrollo y uso de WebLabs. c) Académicamente es obvio que la utilización de WebLabs mejora el aprendizaje de los alumnos y la opinión que éstos tienen de los laboratorio, de las clases y de las asignaturas. De cualquier manera es importante contrastar la calidad de los desarrollos en los WebLabs, consultando periódicamente la opiniones de los usuarios.

Los WebLabs tal y como hemos dicho anteriormente, deben seguir los avances que los elementos que lo hacen posible experimenten. Por ello, los futuros WebLabs deberían estar regidos por tecnologías como Web 2.0 y microservidores, posibilitando que el alumno pueda interactuar con diferentes dispositivos de forma natural, sintiendo que tiene el pleno control de los mismos. Y de forma ambiciosa, el paso siguiente será el conseguir WebLabs plug&play y buscadores de hardware.

Actualmente, nuestro grupo de investigación está trabajando en tres direcciones: a) extender el uso del WebLab-Deusto a microcontroladores y DSP, b) rediseñar el WebLab-Deusto para adaptarlo al uso de microservidores y c) documentar las características académicas del WebLab-Deusto.

Referencias

- [1] Gustavsson, I. et al. "A flexible remote electronics laboratory". *2nd International Symposium in Remote Engineering and Virtual Instrumentation, REV 2005*, Brasov (Romania), July 2005
- [2] Barrón, M. "Laboratorios virtuales para enseñanza por internet" en *I Jornadas de Tendencias sobre eLearning, TEL 2005*, Madrid (Spain), febrero de 2005.

- [3] García Zubía, J. "Programmable Logic and WebLab" V European Workshop on Microelectronics Education, Proceedings of the 5th European Workshop on Microelectronics Education, Lausanne (Switzerland) ISBN: 1-4020-2072-4, 2004, pp: 277-282.
- [4] Ko, C.C.; Chen, Ben. M.; Chen, Jianping. *Creating Web-Based Laboratories*, Springer-Verlag 2004, London, ISBN: 1-85233-837-7
- [5] Alamo, J.A., MIT Microelectronics Weblab, Marzo, 27, 2001. <http://web.mit.edu>
- [6] Pérez M. et al. "Laboratorios de acceso remoto. Un nuevo concepto en los procesos de Enseñanza-Aprendizaje". <http://digital.ni.com/worldwide/latam.nsf/web/all/F54369A0EC8C0B4486256B5F006565A9>
- [7] Kahoraho Bukubiye, E., Larrauri Villamor, J.I. "A WebLab System for the Study of the Control and Protection of Electric Motors", Proceedings of Telecommunication, Electronics and Control, pp. 7. Cuba 2002. ISBN: 84-8138-506-2, 2002.
- [8] Almeida, P., Viera Coito, F., Brito Palma, L. "An Environment for Remote Control". 1st International Workshop on e-learning and Virtual and Remote Laboratories, VIRTUAL-LAB'2004, Setubal (Portugal), August 2004.
- [9] Casini, M.; Prattichizzo, D. y Vicino, A. "e-Learning by Remote Laboratories: a new tool for control education". The 6th IFAC Conference on Advances in Control Education, Finland, 2003.
- [10] Cabello, R. et al. "EMERGE: A European Educational Network for Dissemination of OnLine Laboratory Experiments". Innovations 2004, Ed. iNNER, 2004.
- [11] García-Zubia, J et al. "A new approach for implementing remote laboratories: a practical case". 2nd International Symposium in Remote Engineering and Virtual Instrumentation, REV 2005, Brasov (Romania), July 2005
- [12] Soysal, O. "Computer Integrated Experimentation in Electrical Engineering Education over Distance" Proceedings of ASEE 2000 Annual Conference, Saint Louis, MO, June 2000.
- [13] Gomes, C. "Distance Learning Remote Laboratories using LabVIEW". 1st International Workshop on e-learning and Virtual and Remote Laboratories, VIRTUAL-LAB'2004, Setubal (Portugal), August 2004
- [14] Aliane, N.; Martínez, A.; Fraile, A.; Ortiz, J. "LABNET: laboratorio remoto para control de procesos". Actas de las XI Jornadas de Enseñanza Universitaria de la Informática, JENUI 2005, Madrid (Spain), ISBN: 84-9732-421-8, Julio 2005, pp: 515-522,.
- [15] Pelcz, A. et al. "Remote experiments using JAVA: Implementations in the Virtual Electro Lab project". 1st International Workshop on e-learning and Virtual and Remote Laboratories, VIRTUAL-LAB'2004, Setubal (Portugal), August 2004
- [16] Gomes, L.; Costa, A. "Embedded systems introductory course supported by remote experiments ". 1st International Workshop on e-learning and Virtual and Remote Laboratories, VIRTUAL-LAB'2004, Setubal (Portugal), August 2004.
- [17] Bagnasco, A.; Chirico, M.; Scapolla, A.M. "XML Technology to Design Didactical Distributed Measurement Laboratory (RmwLAB) Instrument", IEEE Transactions on Instrumentation and Measurement, VOL. 54, N° 1, February 2005
- [18] McLellan, D. "Very Dynamic Web Interfaces", O'Reilly Xml.com, <http://www.xml.com/pub/a/2005/02/09/xml-http-request.html>, February 2005.