

Especificación Metodológica de la Implementación y Desarrollo de Entornos de Experimentación

Rafael Pastor, *Member, IEEE*, Roberto Hernández, *Member, IEEE*, Salvador Ros, *Member, IEEE*, y Manuel Castro, *Senior Member, IEEE*.

Abstract—In engineering grades, the accomplishment of experimental practices is limited to a real presence in the laboratory installations. Nevertheless, the development of remote environments has facilitated the use of experimental equipment outside the laboratory, avoiding the spatial and temporary barriers. There are so many initiatives that provides real interaction with the laboratory equipment, but no methodology exists that allows getting a unified vision of the development and access to these laboratories, that is to say, an specification of the experimental environment. This paper describes a specification proposal that relies about the definition of the laboratory components and the functionality of the experimentation environment in terms of these components. Using this approach, it will be possible to focus on the accomplishment of experiments and define them as reusable units of learning in an independent way. The component descriptions are defined in terms of XML tags and also a development framework is modelled in order to add tools for experiment designers and to provide a visual interaction with the laboratory.

Index terms—Remote experimentation, virtual and remote labs, XML, Engineering education, Interactivity, Web based education.

I. INTRODUCCIÓN

TRADICIONALMENTE la enseñanza de las asignaturas de Ingeniería se ha caracterizado por un fuerte componente práctico en forma de laboratorios, que permite desarrollar y aplicar los conceptos teóricos que se asimilan en las clases presenciales. Es un hecho constatado que la evolución del concepto de enseñanza y la aplicación de dicho modelo a la educación a distancia demanda una metodología

educativa diferente a la empleada en la enseñanza presencial, una metodología específica concreta que establece la separación física real existente entre profesor y alumno. Esto es particularmente importante en carreras eminentemente prácticas, donde la ausencia del aula presencial provoca la búsqueda de nuevas soluciones que permitan la realización de actividades desde la distancia.

Una de las principales herramientas de la enseñanza a distancia es la generación y uso de un entorno de aprendizaje que emule el funcionamiento real de un aula convencional: un conjunto de alumnos que puedan conversar entre ellos, asimilar los contenidos distribuidos oralmente por el profesor y discutir dichos contenidos entre ellos y/o el profesor. Para conseguir este objetivo se utiliza un tipo específico de elemento tecnológico denominado plataforma de e-learning, que agrupa al conjunto de estudiantes en torno al concepto de curso virtual. La interacción entre los estudiantes se realiza a través de foros de discusión y la asimilación de contenidos mediante la lectura del material que, habitualmente, se encuentra en un formato electrónico que el alumno puede descargar e imprimir. Así mismo, estas plataformas de e-learning disponen de un conjunto de herramientas que permiten la autoevaluación, o incluso la evaluación final, de los estudiantes y la creación de grupos de trabajo en torno a un objetivo docente específico.

Uno de esos objetivos docentes dentro de una disciplina eminentemente experimental lo constituye la realización de prácticas de laboratorio [1]-[2]. Sin embargo, en el modelo tecnológico de estas plataformas educativas se carece de los elementos y herramientas necesarias para dar soporte de forma directa a este objetivo. Una vez más, la solución que se propone es la emulación de dicho laboratorio, generando un entorno virtual de experimentación dotado de las características propias de un laboratorio real:

- 1) El alumno recibe el guión de las prácticas, indicándole las tareas a realizar y las actividades a seguir para finalizar la sesión experimental con éxito.
- 2) El estudiante dispone de un software específico en el laboratorio para la realización de la práctica, ejecutándose dicho software en un computador y visualizando algún tipo de interfaz gráfico que le permite interactuar con el laboratorio.

Rafael Pastor pertenece al Departamento de Informática y Automática de la UNED, C/Juan del Rosal nº 16, 28040, Madrid, España (autor de contacto, Tel.: 34-91 398 83 83; e-mail: rpastor@dia.uned.es).

Roberto Hernández pertenece al Departamento de Informática y Automática de la UNED, C/Juan del Rosal nº 16, 28040, Madrid, España (e-mail: roberto@dia.uned.es).

Salvador Ros pertenece al Departamento de Informática y Automática de la UNED, C/Juan del Rosal nº 16, 28040, Madrid, España (e-mail: sros@dia.uned.es).

Manuel Castro pertenece al Departamento de Ingeniería Eléctrica, Electrónica y de Control de la UNED, C/Juan del Rosal nº 12, 28040, Madrid, España (e-mail: mcastro@ieec.uned.es).

DOI (Digital Object Identifier) Pendiente

- 3) Las prácticas se realizan en grupo. De esta manera los estudiantes pueden discutir, generar informes y preguntar dudas al instructor del laboratorio.
- 4) Al finalizar la práctica en el laboratorio, el grupo de trabajo debe entregar un informe conforme a un formato específico, que el instructor debe verificar para validar que la práctica se ha realizado de forma correcta.

Los puntos 1, 3 y 4 pueden resolverse mediante el empleo de la misma plataforma educativa utilizada para la distribución de los contenidos teóricos y la realización de actividades de aprendizaje. Sin embargo, el punto 2 requiere un esfuerzo adicional por parte de los responsables de la materia a impartir ya que deben proporcionar al estudiante el mismo entorno visual de experimentación que se encontraría en caso de realizar la práctica de forma presencial. Dada la naturaleza del acceso a las diferentes plataformas de aprendizaje, la utilización de Internet como medio de distribución y acceso a la información del laboratorio se convierte en un factor clave en el desarrollo del entorno de experimentación a distancia.

A continuación se muestra una propuesta metodológica que aborda el desarrollo estructurado de software para la implementación de laboratorios en línea. En la sección II se definen las especificaciones deseables del entorno de experimentación y las posibles tipologías de los laboratorios experimentales en el ámbito de la experimentación a distancia (junto a los criterios de calidad medibles sobre los que se debe fundamentar el factor de éxito de la implementación del laboratorio). Una vez analizadas las características a desarrollar, en la sección III se describe el protocolo que define la metodología propuesta y que pretende implementar el conjunto de especificaciones anteriores. Para ello se define un esquema secuencial de desarrollo modular que emplea un lenguaje de definición para la estructuración en componentes de un laboratorio experimental. Dicho esquema está soportado por un marco de desarrollo que contiene las herramientas necesarias para interpretar la definición en componentes del laboratorio, además de dotarle de herramientas visuales para comprobar el correcto funcionamiento de la implementación software del laboratorio. Finalmente en la sección IV se presentan las conclusiones más importantes que se pueden extraer de los conceptos generales mostrados en las secciones II y III.

II. ESPECIFICACIÓN DEL ENTORNO DE EXPERIMENTACIÓN

A. Características del entorno

Existen desarrollos de diferentes entornos experimentales basados en Internet como los propuestos en [3]-[13] que, aunque ofrecen soluciones particulares, no engloban un conjunto de características básicas en el entorno de experimentación que deben usarse como opciones de diseño:

1. *Facilidad de uso y comprensión.* La interfaz gráfica del entorno de experimentación debe ser amigable y haber sido probada con éxito en otros entornos de simulación disponibles como, por ejemplo, el presentado en [14].
2. *El software empleado por el cliente debe ser multiplataforma.* Puesto que el acceso al entorno de experimentación se hace a través de un navegador web, la interfaz de dicho entorno necesariamente debe estar compuesta por un applet 100% compatible Java. Los estudiantes sólo necesitan un navegador web con soporte Java para poder realizar el conjunto de experimentos disponibles desde el entorno de experimentación. No debe ser necesario otro software añadido a un navegador en particular, denominado plug-in, o software desarrollado a medida para el entorno de experimentación, tal y como ocurre en los trabajos de [3], [5], [6], [11], [15]-[19] entre otros.
3. *Acceso universal al laboratorio mediante una aproximación orientada a la compartición de datos.* El intercambio de información entre el servidor y el cliente se debería realizar mediante un protocolo ligero, en término de envío de datos, facilitándose así:
 - a. Conocer el tipo de usuario que está empleando el entorno de experimentación: alumno o profesor tutor.
 - b. Enviar desde la interfaz gráfica del cliente parámetros y datos que permitan modificar el comportamiento interno del laboratorio. Por ejemplo, en el caso de representar un laboratorio real con el entorno de experimentación, poder ubicar la ejecución del controlador asociado al experimento en el software del cliente y cerrar en lazo de control a través de Internet, de forma que se consiga una interacción real entre ambos entornos.
 - c. Gestionar el ciclo de control del entorno de experimentación. Por ejemplo, en el caso de representar una simulación con el entorno de experimentación, poder detener, comenzar, asignar un nuevo tiempo de muestreo a la simulación, etc.
4. *Acceso en tiempo real al recurso representado por el entorno de experimentación.* La interacción con el entorno de experimentación debe ser dinámica y tener realimentación en tiempo real de los eventos producidos en dicho entorno. Los usuarios no esperan sólo mover barras de desplazamiento, ejecutar el experimento y examinar los datos representados en gráficas y tener que repetir los pasos anteriores si se modifica algún dato. Durante la realización del experimento, cada cambio en las variables de entrada debe ser mostrada en la interfaz del entorno. Los usuarios podrán entonces supervisar en tiempo real cómo evoluciona el comportamiento del

Esta aproximación no es nueva, tal y como se propone en [20]-[21], pero en este caso el protocolo debe ser muy simple y ligero, reduciéndose a un conjunto de órdenes. Estos deben ser independientes del tipo de representación con la que se corresponda el entorno de experimentación, ya sea una simulación o un equipo didáctico real, lo que permite independizar el desarrollo de la interfaz gráfica del cliente del modelo de acceso al sistema representado.

sistema de acuerdo a los valores de las variables interactivas que puedan modificarse.

5. *Definición explícita de experimentos.* Habitualmente, el entorno de experimentación se define en términos de qué variables pueden modificarse y cómo se presenta la interfaz gráfica del usuario en el cliente sin definir de una forma precisa cómo se deben realizar dichos cambios con el objetivo de conseguir algún fin, es decir, especificar lo que se pretende hacer en un experimento. El profesor debe ser capaz de añadir nuevos experimentos al entorno de experimentación con algún mecanismo simple como, por ejemplo, la utilización de ficheros de definición. De esta forma es posible crear diferentes experimentos: limitar las posibilidades de control presentes en la interfaz gráfica, adecuar el tiempo de experimentación a la ejecución real del experimento, cambiar el tiempo de muestreo, cambiar la estrategia de control, etc. Esta característica no está presente en los trabajos previos revisados, y proporciona una forma muy flexible de ajustar el entorno de experimentación a un curso de control en particular.
6. *Generación de eventos de fallos en el entorno de experimentación.* La mayor parte de los entornos de experimentación basados en Web sólo permiten introducir situaciones anómalas si existe una figura de supervisor que puede realizar un conjunto de acciones cuando un estudiante está llevando a cabo un experimento. Sin embargo, es deseable que dicho conjunto de acciones se pudieran programar para producir de manera automática los eventos adecuados que llevaran a situaciones anómalas y que fuera el entorno de experimentación el que supervisara la resolución de dichas situaciones anómalas. Un ejemplo de situación anómala es la presencia de perturbaciones en la planta o ruidos en las señales, por lo que se podría usar esta característica para validar en un experimento que el controlador diseñado por el estudiante sea capaz de responder de manera adecuada a dichas alteraciones.

B. Clasificación de los tipos de experimentos

Todos estos factores de diseño deben ser comunes a los diferentes tipos de sistemas de prácticas de laboratorio que se representan mediante el entorno de experimentación. Siguiendo las argumentaciones presentadas en [22], se pueden agrupar diferentes modalidades de laboratorios bajo dos criterios muy claros: la forma de acceso a los recursos sobre los que experimentar y la naturaleza del sistema sobre el que se opera. Atendiendo al primer criterio, se puede distinguir entre acceso remoto a través de una red y acceso local, es decir, que no es necesaria la presencia de una red de comunicación que permita interactuar entre sí a los diferentes componentes del laboratorio. Desde el punto de vista de la naturaleza del recurso se pueden distinguir entre modelos simulados y sistemas reales, es decir, plantas didácticas tradicionales ubicadas en el laboratorio. Al primer tipo de laboratorios se le denomina laboratorio virtual debido a que

no existe físicamente como tal y para el segundo tipo se emplea la denominación de laboratorio real. Existen otras clasificaciones que podrían usarse [23] donde se distingue el tipo de recurso en base a herramientas hardware o software.

Si se parte inicialmente del criterio por tipo de acceso, se pueden agrupar los diferentes entornos de experimentación en:

1. *Laboratorio remoto.* Se accede a través de Internet a un sistema físico real para su manipulación directa. El software utilizado para el control remoto puede ser un navegador web o una aplicación que necesita ser descargada del servidor del laboratorio. En algunas ocasiones puede que sea posible su visualización e incluso audición en tiempo real.
2. *Laboratorio virtual descargable.* Utilizando un navegador se descarga un applet, un ActiveX o una aplicación que opera localmente con un recurso simulado. Es decir, la interfaz y el núcleo de simulación constituyen un único objeto. No se necesita la instalación de ningún entorno de simulación, salvo los correspondientes plug-in o run-time (por ejemplo, los componentes necesarios para ejecutar applets Java [24], interfaces remotas de Labview [25] o simulaciones SysQuake [26]). También se incluyen las aplicaciones ejecutables independientes.
3. *Laboratorio virtual distribuido.* El cliente utiliza una página HTML, un applet, un ActiveX o una aplicación para conectarse con un servidor en el que se encuentra todo el software de simulación. El cliente ejecuta exclusivamente la interfaz en su ordenador, estableciéndose un diálogo a través de la red entre la interfaz y el servidor de simulaciones.
4. *Laboratorio virtual mixto.* Es análogo al descargable pero necesita obligatoriamente que el cliente tenga instalado en su ordenador el entorno de ejecución de los componentes que se ejecutarán en la parte cliente y se conectan a través de un canal de comunicación propio con el entorno de ejecución de la parte servidora (por ejemplo, el caso de la herramienta Matlab/Simulink [27]).

Existe un último tipo que no se ha descrito en los cuatro grupos anteriores y que se corresponde con el acceso local a un recurso local, que es lo que tradicionalmente se conoce como un laboratorio de prácticas en la educación presencial.

C. Criterios de calidad

El interés fundamental de la realización de experimentos a través de la red se centra en aquellos entornos de experimentación que se puedan aplicar de manera efectiva a la educación a distancia o a otras modalidades no ligadas al entorno educativo presencial, como servicios de formación a empresas comerciales. Un ejemplo claro de esto último se puede aplicar a la industria o a centros de investigación donde la teleoperación remota de dispositivos constituye una característica deseable en los entornos de trabajo de dichas instituciones. Este tipo de operación remota permite eliminar un conjunto de inconvenientes derivados del carácter distribuido de las empresas como es la adquisición de equipos

específicos en una localización, inaccesibles para el resto de centros de la compañía, y que se pueden compartir de manera remota con el conjunto global de ubicaciones de la empresa. Además, de esta manera se reducen costes asociados a los desplazamientos de técnicos y científicos al centro, además de aliviar los presupuestos de adquisición de dispositivos que se duplican con el objetivo de proporcionar la misma funcionalidad en diferentes centros. Este tipo de operación remota abre un mercado de servicios en el cual es posible alquilar el uso remoto de sistemas o dispositivos para obtener un rendimiento puntual, que de otra forma obligaría a adquirir el equipamiento necesario.

En cualquier caso, para facilitar el empleo de este tipo de entornos se deben satisfacer los siguientes criterios de calidad:

1. *Facilidad de instalación y uso.* Los entornos de experimentación implementados deben poseer de manera inherente una manera sencilla e intuitiva de instalarse y de usarse, sin que esto obligue a perder calidad en el trabajo desarrollado por el alumno. Las herramientas del entorno deben paliar la ausencia del tutor presencial mediante el empleo de sistemas de comunicación bidireccional entre el alumno, el tutor virtual y el resto de compañeros del aula.
2. *Acceso a través de Internet.* El empleo de Internet como medio de transmisión exclusivo para el acceso al entorno de experimentación implica que la única herramienta necesaria para un usuario, durante la realización de las prácticas, debe ser un navegador web.
3. *Coste cero.* Es muy importante que el uso del entorno de experimentación no obligue al usuario la compra o adquisición de ningún tipo de licencia software. El único gasto asociado al acceso y empleo del entorno debe ser debido a la forma de acceso a Internet, gasto que puede asumir el usuario desde su domicilio particular (soluciones ADSL o RDSI) o bien, en el caso de instituciones como la UNED, las aulas informáticas del centro asociado o los centros de cálculo equivalentes ubicados en el resto de universidades presenciales.
4. *Interactividad y realismo.* La sensación de realidad en la manipulación remota de un sistema, ya sea virtual o real, proporciona respuestas positivas por parte de los alumnos, lo que motiva el uso del entorno de experimentación como sustituto adecuado del sistema real. En el caso de un laboratorio remoto es muy relevante mostrar de manera audiovisual la evolución del sistema real y que cualquier acción realizada en el entorno de experimentación tenga una realimentación de información y un efecto visual en el entorno. Esta interactividad directa entre el entorno, el usuario y el sistema debe proporcionar una sensación de proximidad al sistema remoto.
5. *Disponibilidad total.* Los entornos de experimentación y el acceso a través de Internet permiten ofrecer al usuario una disponibilidad completa (las 24 horas del día, los siete días a la semana, etc.). Las únicas limitaciones que se deben imponer son las restricciones de acceso

simultáneo, únicamente en el caso de laboratorios remotos, al sistema físico real bien por realización de las prácticas bien por tareas de mantenimiento.

6. *Robustez y fiabilidad.* La disponibilidad del entorno de experimentación obliga a diseñar soluciones fiables que no fallen y que se puedan utilizar durante el tiempo de uso del experimento. Consecuentemente, la selección del laboratorio es muy importante ya que se debe buscar un sistema experimental autónomo que no necesite mantenimiento, o al menos, el mínimo posible.

III. METODOLOGÍA DE IMPLEMENTACIÓN Y DESARROLLO DE UN MARCO DE TRABAJO PARA EL ENTORNO EXPERIMENTAL

A partir de las especificaciones descritas en el apartado anterior surge de manera natural la idea de proponer e implementar una metodología de desarrollo unificada para entornos de experimentación, a partir de la utilización de recursos ya existentes o generados para tal fin. Para ello es necesario emplear una misma notación que permita describir los diferentes laboratorios o sistemas desde los que se tendrá acceso al entorno de experimentación. Una vez que se dispone de una forma común de especificar el laboratorio es necesario proporcionar el conjunto de herramientas, denominado marco de desarrollo, que facilite la puesta en producción de dicho entorno en Internet. Dicho conjunto de herramientas debe poder integrar desarrollos de entornos de experimentación previos y facilitar el desarrollo de nuevos entornos, por lo que es necesario representar de manera precisa la correlación entre los recursos asociados al entorno de experimentación con sus respectivas entidades software: módulos software e interfaces gráficas de usuario. Además, es necesario incorporar de forma explícita el concepto de experimento, tanto a nivel de definición como de manipulación del entorno, de forma que sea posible definir el conjunto de actividades que se pueden realizar de forma secuencial dentro del experimento. Esto último es muy importante porque permite definir tareas de aprendizaje [28] y actividades de aprendizaje colaborativo [29] que se pueden integrar en diferentes plataformas de aprendizaje que soporten lenguajes de etiquetado educativo como IMS Learning Design [30]. Finalmente, el marco de desarrollo debe ser capaz de añadir enlaces de información contextual asociados a los experimentos que permitan explicar los principios básicos del entorno de experimentación o cualquier material de apoyo necesario.

Sin embargo, dotar al conjunto de desarrolladores del entorno de desarrollo, del lenguaje de especificación y de las herramientas adecuadas no garantiza el correcto uso de las mismas. Por lo tanto, es imprescindible especificar un procedimiento secuencial de uso de las herramientas, que es lo que realmente se puede considerar como metodología de implementación de entornos de experimentación.

En la Figura 1 se muestra el diagrama correspondiente al protocolo de trabajo presentado. A partir de un lenguaje de especificación del entorno se genera el entorno experimental mediante el uso de un marco de desarrollo que proporciona un conjunto de herramientas, etiquetadas como 1), 2), 3), 4) y 5)

que se deben usar mediante un protocolo secuencial de trabajo.

De lo descrito anteriormente se deduce que se deben alcanzar los siguientes objetivos específicos para la propuesta de desarrollo metodológico:

- Definir un lenguaje de especificación que permita describir de manera completa cualquier entorno de experimentación, y que tenga en cuenta los factores de diseño comentados en el apartado anterior.
- Proporcionar un marco de desarrollo de entornos experimentales formado por las herramientas necesarias que permitan interpretar una especificación del laboratorio, realizada con el lenguaje generado en el primer sub-objetivo, y permitir la ejecución de diferentes experimentos en el entorno.
- Especificar un procedimiento de trabajo que defina cómo utilizar las diferentes herramientas para estandarizar el desarrollo de los entornos de experimentación.

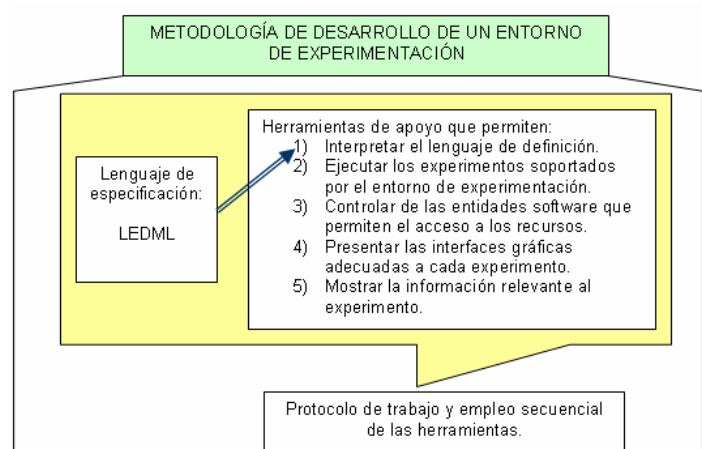


Fig. 1. Diagrama de especificación de la metodología de desarrollo.

A. Especificación del laboratorio

Para la especificación del laboratorio es necesario emplear algún tipo de lenguaje neutro, en el sentido de independiente de la implementación, que permita describir de manera autónoma las diferentes estructuras que aparecen en un entorno experimental o laboratorio. En este caso se emplea un metalenguaje denominado LEDML (Laboratory Experimentation Description Language) basado en XML para la definición de cada laboratorio [31].

La parte conceptual más importante de LEDML consiste en identificar la división del laboratorio en sus diferentes partes constituyentes y definir las propiedades concretas de las etiquetas del metalenguaje de especificación. Por este motivo conviene analizar cuáles son los elementos más comunes en la definición de todo laboratorio. Se pueden dividir los componentes básicos de un laboratorio en:

- Módulos.
- Vistas.
- Experimentos.
- Referencias.

Para clarificar la clasificación propuesta, a continuación se

describen los principios y conceptos que conllevan la definición de los componentes.

En primer lugar, en la implementación de cualquier laboratorio de ingeniería, ya sea virtual o real, se debe desarrollar un software específico que permita la comunicación entre el computador y el sistema, que, dependiendo del tipo de laboratorio, podrá ser un equipo didáctico o un modelo simulado. En cualquiera de los dos casos es necesario desarrollar el código que permita tanto obtener los valores de las magnitudes significativas del laboratorio como cambiar el conjunto de variables que puedan ser alterables como, por ejemplo en un laboratorio real, el porcentaje de apertura de la válvula que controla el caudal de líquido de entrada a un depósito. En todo caso, se puede ver a este software como una caja negra con distintos puertos de entrada y de salida que representa al sistema. Los puertos de entrada se usan para enviar los cambios en las magnitudes manipulables, mientras que los puertos de salida se emplean para observar el estado de las magnitudes de interés, mostrándose en algún tipo de interfaz gráfica creada para tal finalidad.

Una vez desarrollada la implementación, que se denominará interfaz software-sistema, normalmente, se pretende que sobre el laboratorio se ensayen una serie de estrategias de control. A tal efecto es necesario generar un nuevo componente, o reutilizar uno ya existente, para implementar la estrategia de control deseada. Las estrategias más usadas en los laboratorios, dada su sencillez, son el control manual y el control PID (Proporcional-Integral-Derivativo).

En el primer caso, el control manual, hay que facilitar mecanismos al usuario final, normalmente un estudiante, para que pueda cambiar alguna magnitud relacionada con el modelo físico y visualizar los resultados del cambio presentando algún gráfico. De esta manera sólo es necesario recurrir a la interfaz software-sistema. Si se considera el control PID, lo más habitual es implementar la estrategia PID, permitiendo al usuario definir qué magnitudes se pretenden controlar y sobre qué magnitudes se desea realizar la manipulación automática. Por supuesto, y si se considera necesario para el experimento, se deben poder cambiar los parámetros del controlador.

Desde un punto de vista práctico todo lo anterior se puede resumir diciendo que el trabajo esencial para desarrollar un laboratorio, ya sea virtual o real, consiste en escribir un conjunto de programas que de aquí en adelante se denominarán módulos, que encapsulen el software necesario para realizar una tarea específica dentro de la estructura del laboratorio. Por lo tanto, como mínimo será necesario escribir un módulo que implemente la interfaz software-sistema y tantos módulos como estrategias de control se ofrezcan en el laboratorio. Otra alternativa es la escritura de un único módulo monolítico que implemente la interfaz software-sistema junto con las estrategias de control que se ofrezcan. Las posibilidades a este respecto son ilimitadas.

Continuando con el ciclo de desarrollo de un laboratorio, el siguiente paso consiste en elaborar las interfaces gráficas que

se presentarán al usuario durante la interacción con el laboratorio. En este caso, se definirán las diferentes vistas, y se implementarán teniendo en cuenta que deben permitir el envío de cambios en las magnitudes manipulables asociadas a uno o varios módulos y la obtención de información sobre las magnitudes observables de esos mismos módulos.

Una vez que se ha desarrollado todo el software necesario para el acceso y manipulación del laboratorio, el paso final consiste en definir los experimentos. Estos experimentos necesitarán de la ejecución remota del software encapsulado en módulos y la presentación al estudiante de las interfaces gráficas que necesite para ensayar una estrategia de control determinada. Por tanto, los experimentos representarán los diferentes ensayos que se hayan definido para las estrategias de control con que se pueda trabajar en el laboratorio.

Finalmente, es necesario proporcionar las guías y ayudas que el estudiante necesita, recurriendo para ello a hipervínculos de información que se denominan referencias. De esta manera, se permitirá el acceso de los estudiantes a la documentación necesaria que les enseñe a usar de forma correcta el laboratorio.

```
<?xml version="1.0" encoding="utf-8" ?>
<!DOCTYPE system SYSTEM
"http://lab-virtual.dia.uned.es/related/dtd/ledml.dtd">
<system name="RANDOM_GENERATOR" type="0">
  Produces random data for test
  <module name="GENERATOR_MODULE">
    <var name="max_value" type="double" initial="10" max="10000" min="0"
units="N/A">max random value</var>
    <var name="min_value" type="double" initial="0" max="-10000" min="0"
units="N/A">min random value</var>
    <var name="value1" type="double" initial="0" max="10000" min="-
10000" units="N/A">random value 1</var>
    <var name="value2" type="double" initial="0" max="10000" min="-
10000" units="N/A">random value 2</var>
    <var name="value3" type="double" initial="0" max="10000" min="-
10000" units="N/A">random value 3</var>
    <var name="value4" type="double" initial="0" max="10000" min="-
10000" units="N/A">random value 4</var>
    <implementation type="JAVA"
jarfile="file:///D:/users/rafa/RLAB_Xml/RLAB/modules/Ejemplo/ModuloGe
nerador.jar" classname="es.uned.dia.related.modules.ejemplo.Generador">
Run the module for random data</implementation>
  </module>
  <experiment name="Test" sampleTime="1000"> Test
  <duration type="User"/>
  <run name="GENERATOR_MODULE">
    <interactives names="max_value,min_value"
show="true,true"/>
    <paint names="value1,value2,value3,value4"
colors=" blue,red,yellow,pink"/>
  </run>
  </experiment>
</system>
```

Fig. 2. Ejemplo de definición de componentes de un laboratorio.

En la figura 2 se muestra un ejemplo de definición de un laboratorio mediante etiquetas del metalenguaje LEDML. El laboratorio representa un generador de números aleatorios que expone cuatro variables (value1, value2, value3, value4) cuyo valor se genera de manera aleatoria entre los valores

max_value y min_value. La generación de los valores se realiza cada segundo dentro del módulo GENERATOR_MODULE. En este fichero de especificación se define un experimento llamado Test que permite verificar el funcionamiento del módulo GENERATOR_MODULE y muestra gráficamente (etiqueta <paint>) la evolución temporal de las cuatro variables. Es el ejemplo que se distribuye con el marco de desarrollo y que permite verificar que el funcionamiento correcto del marco.

B. Marco de desarrollo

El marco de desarrollo proporciona un conjunto de funcionalidades a través de unas herramientas software que permiten el acceso al laboratorio y cuyos objetivos son las siguientes:

- 1) Validar y leer la especificación LEDML del laboratorio.
- 2) Proporcionar métodos de acceso remoto al laboratorio.
- 3) Posibilitar la utilización de diferentes interfaces gráficas que permitan un uso sencillo del acceso remoto.
- 4) Proporcionar mecanismos de monitorización.
- 5) Facilitar un servicio de búsqueda de laboratorios y experimentos integrados en la red de entornos de experimentación.

Las herramientas que se facilitan con el marco de desarrollo realizan las siguientes funciones:

1. Visualizar la estructura del laboratorio y realizar acciones sobre el mismo. El applet de experimentación permite visualizar los componentes definidos mediante la especificación. En la figura 3 se muestra el applet de experimentación y, tal y como se puede apreciar en dicha figura, contiene una vista jerárquica de los componentes definidos en el fichero de especificación LEDML. Además permite realizar una sesión experimental seleccionando el experimento, presentándose al inicio del mismo aquellas vistas que sean necesarias para realizar la manipulación del laboratorio virtual o remoto. El applet se presenta cuando se accede al entorno de experimentación y permite ejecutar los diferentes experimentos definidos en el fichero de especificación y adicionalmente permite comprobar la estructura modular del laboratorio especificado. Cuando se comienza la ejecución de un experimento se presentan las vistas indicadas en la especificación del experimento, lo que permite la interacción entre el usuario y el laboratorio en el escenario experimental definido. En el caso de la figura 3 se muestra la estructura de un laboratorio correspondiente a un motor de corriente continua donde existen varios escenarios de control: manual, control de velocidad en lazo abierto y en lazo cerrado y control de posición en lazo abierto y lazo cerrado.
2. Visualizar la información del experimento. Esto se realiza mediante las propias vistas definidas como componentes propios, mediante el componente de variables interactivas (que permite modificar las variables del experimento en curso, véase la figura 4) y

mediante el componente de gráfico de tendencias, que representa la evolución temporal del experimento, tal y como se muestra en la figura 5. El componente de variables interactivas permite una manipulación simple de las variables definidas en los módulos de laboratorio. Se puede apreciar en la figura 4 que se indica el tipo y nombre de variable a la izquierda y se proporciona un editor del valor a la derecha (en este caso al ser un número de tipo double, una caja de texto). La evolución temporal de las variables de los módulos involucrados en un escenario experimental se muestran en el gráfico de tendencias. En la figura 5 se presenta la evolución de los valores de las variables definidas en el fichero de especificación de la figura 2 durante la ejecución del experimento Test. Para usar esta herramienta basta con declarar que variables se desean representar en el experimento mediante la etiqueta <paint>.

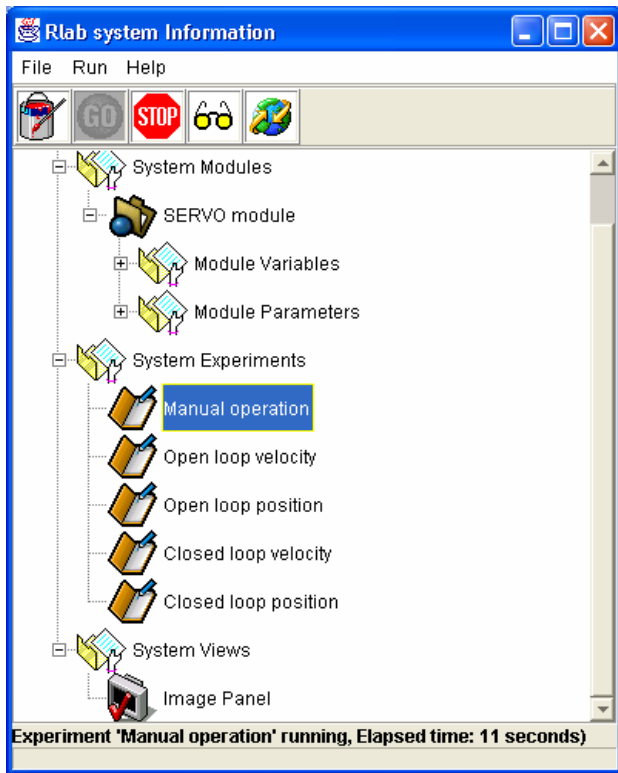


Fig. 3. Applet de experimentación.

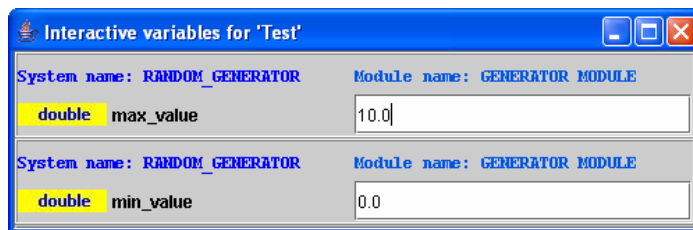


Fig.4. Componente de variables interactivas.

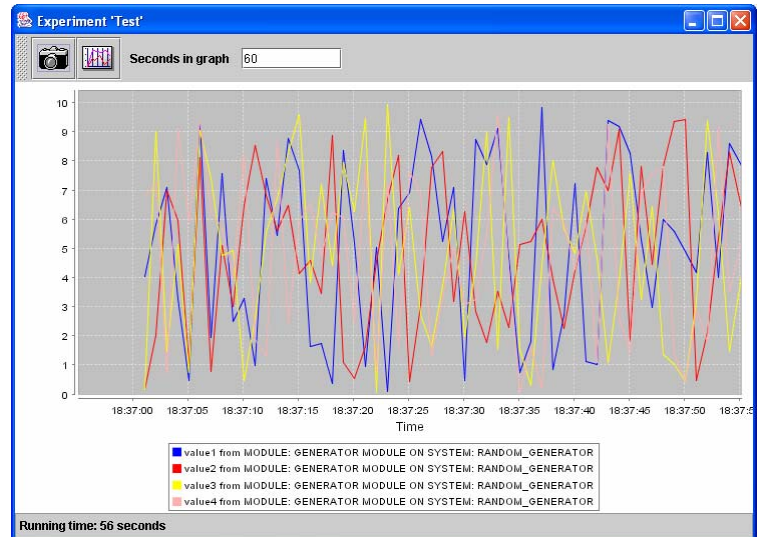


Fig.5. Componente de gráfico de tendencias.

C. Protocolo de desarrollo de un entorno de experimentación

Los pasos del procedimiento son los siguientes:

1. *Definición de la conducta estática del laboratorio en términos de variables y parámetros.* Se debe realizar un trabajo de clasificación de estas variables para agruparlas en conjuntos con una funcionalidad o método de acceso similar. Por ejemplo, las variables de estado de un motor de corriente continua se leen mediante la correspondiente tarjeta de adquisición de datos, mientras que las variables del controlador serán accesibles desde el código software que lo implementa.
2. *Programación de módulos.* Para cada de grupo de variables y parámetros definidos en el paso anterior se desarrolla el módulo adecuado. El administrador del sistema puede optar para cada módulo por una implementación nativa o Java para su posterior programación, posiblemente reutilizando código ya desarrollado. Por lo tanto, por cada grupo de variables se debe implementar un módulo.
3. *Desarrollo de las vistas.* Las distintas ventanas gráficas ayudan a describir y entender de forma visual la conducta dinámica del sistema, por lo que constituyen una pieza fundamental en la construcción del laboratorio.
4. *Tarea de documentación.* Los resultados de la misma son documentos HTML (o cualquier URL válida) que se incluyen en el sistema como referencias.
5. *Definición de la conducta dinámica del sistema.* En primer lugar el administrador decide qué propiedades del sistema pueden ser manipuladas por los usuarios. A continuación se deben definir los experimentos empleando los módulos disponibles en el entorno y todos aquellos accesibles en otros laboratorios. Finalmente, se define la duración del experimento y las distintas conexiones de puertos de entrada y salida entre los módulos que se ejecutarán en el experimento.

IV. CONCLUSIONES

El uso de la metodología estructurada de desarrollo propuesta permite definir y clasificar el conjunto de experimentos que se pueden ofrecer en línea a través de diferentes entornos de experimentación. Esta clasificación proporciona un método para globalizar la totalidad de los experimentos ofertados dentro de una red de laboratorios, a la vez que permite agruparlos por funcionalidades.

Además, de la propia separación de tareas del protocolo de trabajo se deduce que la realización independiente de las vistas y módulos produce modelos de reutilización de software que presentan beneficios inmediatos de disponibilidad de los componentes a otros desarrolladores de laboratorios en línea. Se pueden reutilizar tanto el software de los módulos como las interfaces gráficas o vistas, por ejemplo las mostradas en las figuras 6 y 7, que fueron desarrolladas para un entorno de experimentación con actividades de manipulación sobre un motor de corriente continua [32]. Para dicho laboratorio se empleó como herramienta de desarrollo el software Easy Java Simulations [33] que permite reutilizar los modelos de simulación que se preparan en la propia herramienta y las interfaces gráficas, como la mostrada en la figura 6, dentro de un entorno Java. La vista representa la posición del motor (en rojo) y la evolución de dicha posición, y se puede reusar para cualquier tipo de motor de características parecidas. Por otra parte se pueden emplear técnicas de programación de interfaces para desarrollar componentes de presentación de video en tiempo real, como la mostrada en la Figura 7, que permiten dar una sensación de realidad y usabilidad a la teleoperación de un laboratorio real.

De esta manera si se genera un repositorio central con control de versiones que permita agrupar y catalogar los componentes reutilizables de vistas y módulos se pueden acelerar los tiempos de desarrollo de los proyectos que tengan como objetivo final la elaboración de un laboratorio en línea empleando el protocolo propuesto, sin más que añadir las líneas de descripción LEDML correspondientes.

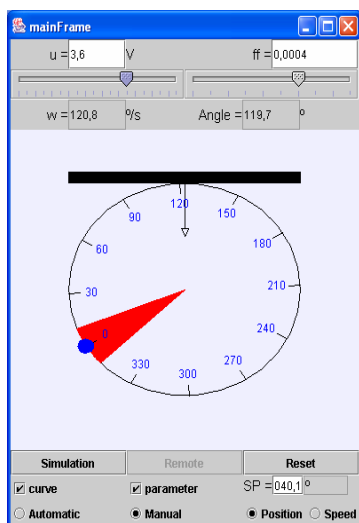


Fig.6. Vista virtual del entorno de experimentación desarrollado para el caso del motor de corriente continua.

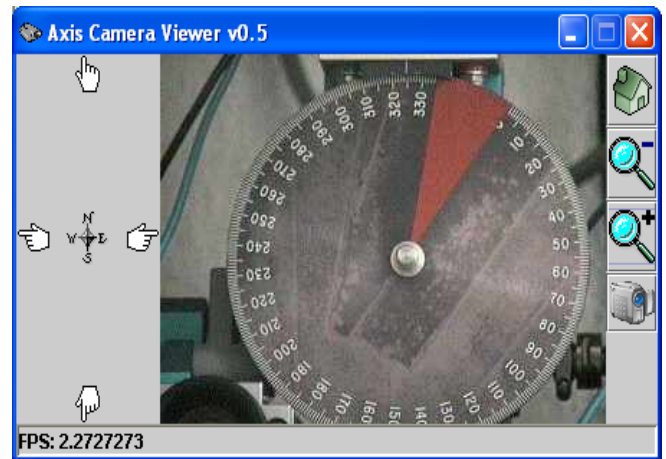


Fig. 7. Vista virtual del entorno de experimentación desarrollado para el caso del motor de corriente continua.

AGRADECIMIENTOS

Este trabajo ha sido subvencionado por la CICYT (Comisión Interministerial de Ciencia y Tecnología) bajo el proyecto DPI2004-05903.

REFERENCIAS

- [1] R. Pastor; *Especificación Formal de Laboratorios Virtuales y Remotos: Aplicación a la Ingeniería de Control*. Tesis doctoral, UNED, 2006.
- [2] S. Dormido; F. Torres, *Algunas consideraciones sobre las TIC's y la educación en Automática*. Revista Iberoamericana de Automática e Informática Industrial, vol. 2, nº 2, pp. 3-7, 2005.
- [3] A. Leva, *A simple and flexible experimental laboratory for automatic control courses*. Control Engineering Practice, vol. 14, nº 2, Feb. 2006, pp. 167-176.
- [4] K. Arzen, A. Blomdell and B. Wittenmark, *Laboratories and Real Time Computing*. Control Systems Magazine, Vol. 25, nº 1, Feb. 2005, pp. 30-34.
- [5] J.A. Asumadu, R. Tanner, J. Fitzmaurice, M. Kelly, H. Ogunleye, J. Belter, S.C. Koh, *A Web-based electrical and electronics remote wiring and measurement laboratory (RwmLAB) instrument*. IEEE Transactions on Instrumentation and Measurement, Vol. 54, nº 1, Feb. 2005, pp. 38-44.
- [6] L. Gomes, A. Costa, *Remote laboratory support for an introductory microprocessor course*. Proceedings of the IEEE International Conference on Microelectronic Systems Education, 12-14 Junio 2005 pp. 21-22.
- [7] S.C. Sivakumar, W. Robertson, M. Artimy, N. Aslam, *A web-based remote interactive laboratory for Internetworking education*. IEEE Transactions on Education, Vol. 48, nº 4, Nov. 2005, pp. 586-598.
- [8] A. Valera, J.L. Diez, M. Valles, P. Albertos, *Virtual and remote control laboratory development*. IEEE Control Systems Magazine, Vol. 25, nº 1, Feb. 2005, pp: 35-39.
- [9] G. Viedma, I.J. Dancy, K.H. Lundberg, *A Web-based linear-systems iLab*. American Control Conference 2005. Proceedings of the 2005. 8-10 Junio 2005 pp. 5139-5144.
- [10] M. Basso, G. Bagni, *ARTIST: a real-time interactive Simulink-based telelab*. IEEE International Symposium on Computer Aided Control Systems Design, 2004. pp. 196-201.
- [11] M. Casini, D. Prattichizzo, A. Vicino, *The automatic control telelab*. IEEE Control Systems Magazine. Vol. 24, nº 3, Jun. 2004 pp. 36-44.
- [12] F. Cennamo, F. Fusco, M. Inverno, A. Masi, A. Ruggiero, *A remotely controlled measurement system for education and training of experiments in wind tunnel*. Instrumentation and Measurement Technology Conference, 2004. IMTC 04. Proceedings of the 21st IEEE. Vol. 2, 18-20 May 2004 pp. 991-995.
- [13] M. Casini, D. Prattichizzo, and A. Vicino; *The automatic control telelab: A user-friendly interface for distance learning*. IEEE Trans. Educ., vol. 46, nº 2, pp. 252-57.

- [14] J. Sánchez, F. Morilla, S. Dormido, J. Aranda, P. Ruipérez; *Virtual control lab using Java and Matlab: A qualitative approach*. IEEE Control Systems Magazine, vol. 22, n° 2, 2002, pp. 8-20.
- [15] A. Ferrero, S. Salicone, C. Bonora, and M. Parmigiani; *ReMLab: A Java-based remote, didactic measurement laboratory*. IEEE Trans. Instrum. Meas., vol. 52, no. 3, pp. 710–715, Jun. 2003.
- [16] J.W. Overstreet and A. Tzes; *An Internet based real-time control engineering laboratory*; IEEE Control Systems Magazine, vol. 19, no 5, pp. 19-34, Oct. 1999.
- [17] C. Schmid; *A Remote Laboratory Using Virtual Reality on the Web*. Simulation, Vol. 73, n° 1, pp. 13-21, Jul. 1999.
- [18] W. Sheng, L. Choo-Min, and L. Khiang-Wee. *An integrated Internet based control laboratory*. IFAC/IEEE Symposium on Advances in Control Education, Gold Coast (Australia), Diciembre 2000.
- [19] K. Ling, Y. Lai, and K. Chew.; *An online Internet laboratory for control experiments*. IFAC/IEEE Symposium on Advances in Control Education, Gold Coast (Australia), Diciembre 2000.
- [20] D. Gillet, H. A. Latchman, Ch. Salzmann, and O. D. Crisalle; *Hands-On Laboratory Experiments in Flexible and Distance Learning*. Journal of Engineering Education, pp.187-191, Abril 2001.
- [21] D. Gillet, C. Salzmann, and P. Huguenin; *A Distributed Architecture for Teleoperation over the Internet with Application to the Remote Control of an Inverted Pendulum*. Libro "Lecture Notes in Control and Information Sciences 258: Nonlinear Control in the year 2000", vol. 1, pp. 399-407, Springer-Verlag, London, 2001.
- [22] J. Sánchez. *Un nuevo enfoque metodológico para la enseñanza a distancia de asignaturas experimentales: Análisis, Diseño y Desarrollo de un laboratorio virtual y remoto para el estudio de la automática a través de Internet*. Tesis doctoral, UNED, 2002.
- [23] Luís Anido, Martín Llamas, Manuel J. Fernández. "Internet-based Learning by Doing". IEEE Transactions on Education. ISSN 0018-9359. Vol. 44, No. 2. CD-ROM Folder 17. May, 2001.
- [24] <http://java.sun.com/> última consulta noviembre de 2006.
- [25] <http://www.ni.com/labview/> última consulta noviembre de 2006.
- [26] <http://www.calerga.com/products/Sysquake/index.html> última consulta noviembre de 2006.
- [27] <http://www.mathworks.com/> última consulta noviembre de 2006.
- [28] D. Edelson, R. Pea and L. Gomez; *Constructivism in the Collaboratory*. In B. G. Wilson (Ed.), *Constructivist learning environments: Case studies in instructional design*, (pp. 151-164). Englewood Cliffs, NJ: Educational Technology Publications.
- [29] E. Gaudioso, J.G. Boticario; *Towards web-based adaptive learning communities*. Proceedings of the 11th International Conference on Artificial Intelligence in Education. Sidney, Australia, Julio 20-24, 2003 (<http://www.ia.uned.es/~elena/publi.htm#internacionales>).
- [30] <http://www.imsglobal.org/learningdesign/index.html> última consulta noviembre de 2006.
- [31] R. Pastor, J. Sánchez, S. Dormido; *Related: a framework to publish web-based laboratory control systems*. First IFAC Workshop on Internet Based Control Education (IBCE'01). Diciembre de 2001, Madrid, España.
- [32] R. Pastor, C. Martín, J. Sánchez and S. Dormido; *A XML framework customization for a servo motor web-based laboratory*. Conferencia Anual del Instituto Americano de Ingenieros Químicos (AIChE). San Francisco, CA, Nov. 16-21, 2003.
- [33] <http://www.um.es/fem/Ejs/> última consulta noviembre de 2006



Rafael Pastor es Doctor en Ingeniería Informática y Titular de Escuela Universitaria. Es Director de Innovación del Centro de Innovación y Desarrollo Tecnológico (CInDeTec) de la UNED, dónde desarrolla labores de transferencia de conocimiento sobre plataformas educativas. Ha participado en varios proyectos de investigación como investigador asociado así como ha dirigido proyectos PROFIT del Ministerio de Industria, Comercio y Turismo. Además ha publicado en revistas y congresos, tanto nacionales e internacionales. Es miembro de la Sociedad de Educación de IEEE y Member del IEEE.



Roberto Hernández Berlinches es Doctor en Ciencias y Titular de la Escuela Técnica Superior de Ingeniería Informática de la UNED. Ha participado en numerosos proyectos de investigación como investigador y ha publicado en revistas y congresos, tanto nacionales e internacionales. Además, ha publicado diversos libros y material multimedia dentro de sus líneas de investigación y docencia. Es miembro de la Sociedad de Educación de IEEE y Member del IEEE.



Salvador Ros es Licenciado en Ciencias Físicas y Titular de Escuela Universitaria de la ETSII de la UNED. Actualmente es Director de Tecnologías Educativas del Centro de Innovación y Desarrollo Tecnológico (CInDeTec) de la UNED. Ha participado en numerosos proyectos de investigación como investigador y ha publicado en revistas y congresos, tanto nacionales e internacionales. Ha publicado igualmente diversos libros y material multimedia dentro de sus líneas de investigación y docencia. Es miembro de la Sociedad de Educación de IEEE y Member del IEEE.



Manuel Castro es Doctor Ingeniero Industrial y Catedrático de Universidad. Ha sido Vicerrector de Nuevas Tecnologías de la UNED, así como Subdirector de Ordenación Académica y de Investigación en la Escuela de Ingenieros Industriales de la UNED, y Director del Centro de Servicios Informáticos de la UNED, siendo actualmente Director de Departamento. Ha participado en numerosos proyectos de investigación como investigador, coordinador y director y ha publicado en revistas y congresos, tanto nacionales e internacionales. Ha publicado igualmente diversos libros y material multimedia dentro de sus líneas de investigación y docencia. Es Senior Member del IEEE así como miembro del Comité de Administración de la Sociedad de Educación del IEEE.