

Novática, revista fundada en 1975, es el órgano oficial de expresión y formación continua de ATI (Asociación de Técnicos de Informática)

ATI es miembro de CEPIS (Council of European Professional Informatics Societies) y tiene un acuerdo de colaboración con ACM (Association for Computing Machinery). Tiene asimismo acuerdos de vinculación o colaboración con AdaSpain, AI² y ASTIC

<<http://www.ati.es/novatica/>>

CONSEJO EDITORIAL

Antoni Carbonell Noguera, Francisco López Crespo, Julián Marcelo Chocho, Celestino Martín Alonso, Josep Molas i Bertrán, Roberto Moya Quiles, Gloria Nistal Rosique (Presidenta del Consejo), César Pérez Chirinos, Mario Piattini Velthuis, Fernando Píera Gómez, Miquel Sàrries Griñó, Carmen Ugarte García, Asunción Yturbe Herranz

Coordinación Editorial
 Rafael Fernández Calvo <rfcalvo@ati.es>

Composición y autoedición
 Jorge Llácer

Administración
 Tomás Brunete, María José Fernández

SECCIONES TÉCNICAS: COORDINADORES

- Arquitecturas**
 Antonio Gonzalez Colás (DAC-UPC) <antonio@ac.upc.es>
- Bases de Datos**
 Coral Calero Muñoz, Mario G. Piattini Velthuis (Escuela Superior de Informática, UCLM) <Coral.Calero@uclm.es>, <mpiatini@inf-cr.uclm.es>
- Calidad del Software**
 Juan Carlos Granja (Universidad de Granada) <jcgranja@goliat.ugr.es>
- Derecho y Tecnologías**
 Isabel Hernando Collazos (Fac. Derecho de Donostia, UPV) <ihernando@legaltek.net>
- Enseñanza Universitaria de la Informática**
 Cristóbal Pareja Flores (Dep. Sistemas Informáticos y Programación-UCM) <cpareja@sip.ucm.es>
- Informática Gráfica**
 Roberto Vivó (Eurographics, sección española) <rvivo@dsic.upv.es>
- Ingeniería del Software**
 Luis Fernández (PRIS-E.L./UEM) <lufern@dpris.esi.uem.es>
- Inteligencia Artificial**
 Federico Barber, Vicente Botti (DSIC-UPV) <fvbotti, fbarber@dsic.upv.es>
- Interacción Persona-Computador**
 Julio Abascal González (FI-UPV) <julio@si.ehu.es>
- Internet**
 Alonso Álvarez García (TID) <alonso@ati.es>
 Llorenç Pagés Casas (Atlante) <pagés@ati.es>
- Lengua e Informática**
 M. del Carmen Ugarte (IBM) <cugarte@ati.es>
- Lenguajes informáticos**
 Andrés Marín López (Univ. Carlos III) <amarin@it.uc3m.es>
 J. Ángel Velázquez (ESCET-URJC) <a.velazquez@escet.urjc.es>
- Libertades e Informática**
 Alfonso Escolano (PIR-Univ. de La Laguna) <aescolan@ull.es>
- Lingüística computacional**
 Xavier Gómez Guinovart (Univ. de Vigo) <jgomez@uvigo.es>
 Manuel Palomar (Univ. de Alicante) <mpalomar@dlsi.ua.es>
- Profesión informática**
 Rafael Fernández Calvo (ATI) <rfcalvo@ati.es>
 Miquel Sàrries Grinyó (Ayto. de Barcelona) <msarries@ati.es>
- Seguridad**
 Javier Areitio (Redes y Sistemas, Bilbao) <jareitio@orion.deusto.es>
- Sistemas de Tiempo Real**
 Alejandro Alonso, Juan Antonio de la Puente (DIT-UPM) <[alonso.jpunte]@dit.upm.es>
- Software libre**
 Jesús M. González Barahona, Pedro de las Heras Quirós (GSYC, URJC) <[jgb, pheras]@gsyc.escet.urjc.es>
- Tecnología de Objetos**
 Esperanza Marcos (URJC) <e.marcos@escet.urjc.es>
 Gustavo Rossi (LIFIA-UNLP, Argentina) <gustavo@sol.info.unpl.edu.ar>
- Tecnologías para la Educación**
 Benita Compostela (F. CC. PP. - UCM) <benita@diad.eunet.es>
 Josep Sales Rufí (ESPIRAL) <jsales@pie.xtec.es>
- Tecnologías y Empresa**
 Pablo Hernández Medrano <phmedrano@terra.es>
- TIC para la Sanidad**
 Valentín Masero Vargas (DI-UNEX) <vmasero@unex.es>

Las opiniones expresadas por los autores son responsabilidad exclusiva de los mismos. Novática permite la reproducción de todos los artículos, salvo los marcados con © o copyright, debiéndose en todo caso citar su procedencia y enviar a Novática un ejemplar de la publicación.

Coordinación Editorial y Redacción Central (ATI Madrid)
 Padilla 66, 3º, dcha., 28006 Madrid
 Tlf.914029391; fax.913093685 <novatica@ati.es>

Composición, Edición y Redacción ATI Valencia
 Palomino 14, 2º, 46003 Valencia
 Tlf./fax 963918531 <secreval@ati.es>

Administración, Suscripciones y Redacción ATI Cataluña
 Via Laietana 41, 1º, 1ª, 08003 Barcelona
 Tlf.934125235; fax 934127713 <secregen@ati.es>

Redacción ATI Andalucía
 Isaac Newton, s/n, Ed. Sadiel, Isla Cartuja 41092 Sevilla
 Tlf./fax 954460779 <secreand@ati.es>

Redacción ATI Aragón
 Lagasca 9, 3-B, 50006 Zaragoza
 Tlf./fax 976235181 <secreara@ati.es>

Redacción ATI Asturias-Cantabria <gp-astucant@ati.es>

Redacción ATI Castilla-La Mancha <gp-clmancha@ati.es>

Redacción ATI Galicia
 Recinto Ferial s/n, 36540 Silleda (Pontevedra)
 Tlf.986581413; fax 986580162 <secregal@ati.es>

Publicidad: Padilla 66, 3º, dcha., 28006 Madrid
 Tlf.914029391; fax.913093685 <novatica.publicidad@ati.es>

Imprenta: 9 Impressió S.A., Juan de Austria 66, 08005 Barcelona.
Depósito Legal: B 15.154-1975
ISBN: 0211-2124; CODEN NOVAEC

Portada: Antonio Crespo Foix / © ATI 2002

SUMARIO

En resumen: ¿eXtremismo? Sí, gracias 3
Rafael Fernández Calvo

Monografía: «eXtreme Programming / Programación eXtrema»
 (En colaboración con **Informatik/Informatique** y **Upgrade**)

Editor invitado: *Luis Fernández Sanz*
Presentación: eXtreme Programming y la mejora en el desarrollo del software 5

Luis Fernández Sanz
Referencias útiles sobre eXtreme Programming 7

Luis Fernández Sanz
eXtreme Programming (XP): un nuevo método de desarrollo de software 8

César F. Acebal, Juan M. Cueva Lovelle
La necesidad de velocidad: automatización de las pruebas de aceptación en un entorno de Programación Extrema 13

Lisa Crispin, Tip House, con la contribución de *Carol Wade*
Estudios cualitativos sobre XP en una empresa de tamaño mediano 22

Robert Gittins, Sian Hope, Ifor Williams
XP en proyectos complejos: algunas extensiones 27

Martin Lippert, Stefan Roock, Henning Wolf, Heinz Züllighoven
XP e Ingeniería del Software: una opinión 32

Luis Fernández Sanz
Programación Extremista 36
Michael McCormick

Secciones Técnicas

Informática Gráfica
Modelado geométrico para visualización en tiempo real 39
Inmaculada Remolar, José Ribelles, Óscar Belmonte, Miguel Chover, Cristina Rebollo

Interacción Persona-Computador
Modelos y herramientas para diseño y evaluación de la interfaz de usuario 44
Fabio Paternò, Laila Paganelli, Carmen Santoro

Ingeniería Semiótica y comunicabilidad de las interfaces de usuario 49
Clarisse Sieckenius de Souza

Software libre
Un entorno para enseñanza basado en software libre 55
José Alfonso Accino

Tecnologías y Empresa
Knowledge Management en su organización 61
Pablo Hernández Medrano

Referencias autorizadas 65

Sociedad de la Información

Personal y transferible
Mi opinión sobre las patentes de software en Europa 70
Jesús M. Gonzalez-Barahona

Programar es crear
¿Queso! 71

25º Concurso Internacional de Programación ACM (2001): problema B
«Configuración de un aeropuerto»: solución 72
Manuel Carro, Ángel Herranz, Julio Mariño, Pablo Sánchez

Asuntos Interiores

Programación de Novática 76
Normas de publicación para autores / Socios Institucionales 77

Monografía del próximo número: «Recuperación de la información y la Web»

Software libre

José Alfonso Accino

Dirección de Enseñanza Virtual, Universidad de Málaga; Socio de ATI

<jaccino@ieev.uma.es>

Resumen: como en otros ámbitos, el software libre puede constituir una solución eficaz para el desarrollo de entornos de enseñanza virtual. Su principal ventaja, más allá de su bajo coste, es la posibilidad de construir entornos abiertos, modulares, escalables, de rápida puesta en servicio, versátiles y de gran capacidad de adaptación, especialmente adecuados, por tanto, para comunidades amplias y dinámicas en las que se requiera un elevado grado de interacción y una respuesta ágil a las necesidades de los usuarios.

Palabras clave: software libre, enseñanza virtual, campus virtual, entornos de formación.

1. Introducción

El objetivo del presente artículo es exponer las líneas generales del modelo de entorno de enseñanza virtual, basado en software libre o abierto, actualmente en uso en la Dirección de Enseñanza Virtual (DEV) de la Universidad de Málaga (UMA) <<http://www.ieev.uma.es>>.

La consideración de las nuevas tecnologías como herramienta básica de comunicación y formación ha sido una constante ya desde sus orígenes en el Instituto de Ciencias de la Educación (ICE) de la UMA, acorde con la tarea, prioritaria entonces, de actuar como elemento catalizador de la renovación pedagógica entre el profesorado no universitario. Corresponde a esta etapa (1988) la instalación de un BBS accesible por red telefónica y por medio del cual, además de los habituales servicios de transferencia de archivos y mensajería electrónica, se ponía, con software elaborado ex-profeso, la documentación y bases de datos del Servicio a la disposición de aquellos profesores y centros de enseñanza que, con un cierto carácter pionero, desearan incorporar la tecnología de las redes a su trabajo diario.

No obstante, es especialmente a partir de 1994, con la reestructuración del ICE y la reorientación de su actividad hacia el profesorado universitario, primero como Dirección de Innovación Educativa y luego de Enseñanza Virtual, cuando la utilización de las herramientas de red adquiere un carácter prioritario.

Sucesivas convocatorias de Proyectos de Innovación Educativa permitieron pulsar las preferencias y opiniones del profesorado más proclive a la utilización de nuevas tecnologías, información que se completó con encuestas anuales y que puso de manifiesto una amplia variedad de necesida-

Un entorno para enseñanza virtual basado en software libre

des, expectativas, grados de motivación y niveles de formación en el uso de nuevas herramientas.

La constatación de que tal capital humano constituía el más importante de los activos hizo patente la necesidad de disponer no sólo de un simple web informativo, sino de un completo soporte de asesoramiento, diseño, formación y experimentación para todo tipo de proyectos educativos innovadores. A lo largo de este proceso y aunque se optó por una solución rápidamente operativa, no por ello se han dejado de tomar en consideración y probar en su momento otras alternativas, tanto basadas en software abierto como en plataformas comerciales integradas, desde la plataforma WEST (*Web Educational Support Tools*, University College, Dublin) en 1995, hasta WebCT en sus versiones 1.3, 2.2 y 3.1 en 1999, 2000 y 2001, respectivamente.

2. Plataformas comerciales vs. desarrollo propio

Puede decirse que las alternativas en cuanto a servicios de formación virtual se polarizan en torno a estas dos posibilidades. Las ventajas de las plataformas comerciales integradas son bien conocidas en cuanto que son comunes a todo el software comercial: el poder disponer de una herramienta ya realizada y casi lista para usar.

Sin embargo, también sus inconvenientes son comunes: cualquier herramienta adquirida, más o menos «llave en mano», tiene una curva de aprendizaje que, en algunos casos, puede ser bastante inclinada. Este inconveniente es más significativo si se pretende su utilización por un colectivo, como el académico, con una amplia variedad de experiencias y prácticas educativas previas. Así, se da el caso de que los menos motivados, a quienes no interesa la «virtualización» de la enseñanza, consideran excesivo aprender el manejo de tales aplicaciones mientras que quienes dominan la tecnología y gustan de experimentar con sus posibilidades, prefieren conocer diversas alternativas y poner a prueba sus propias ideas, ya sea buscando o desarrollando sus propias herramientas, o instalando por sí mismos sus propios servidores de información. A esto hay que añadir, más allá de la mera traducción, el fuerte sesgo anglosajón de la práctica totalidad de las plataformas comerciales, ya tengan su origen en entornos empresariales o propiamente educativos.

Es preciso considerar que la enseñanza no es un proceso normalizado como lo es, por ejemplo, la gestión contable, y aunque, en general, las plataformas comerciales ofrezcan

más o menos posibilidades de adaptación, en cualquier caso son las predefinidas por el fabricante. Esto puede ser un inconveniente, tanto para los usuarios menos motivados y más reacios a cambiar su práctica educativa y que, como mucho, estarían dispuestos a seguir utilizando sus herramientas habituales, como para los más avanzados, precisamente aquellos que, por su dinamismo, pueden enriquecer el sistema con sus ideas, hallazgos y aportaciones y que podrían verse obligados, de lo contrario, a moverse fuera del entorno marcado.

Estas limitaciones de los también llamados Sistemas de Gestión de Aprendizaje (LMS, *Learning Management Systems*) han empezado a hacerse patentes entre usuarios e instituciones que comienzan a cuestionar su validez como herramientas pedagógicas [MIT-www-1], lo que ha llevado a algunas de ellas, como el Instituto de Tecnología de Massachusetts (MIT), las universidades de Stanford y Carolina del Norte, entre otras, a promover conjuntamente proyectos como la *Open Knowledge Initiative* (OKI), con la intención explícita de elaborar un nuevo modelo de sistemas de enseñanza virtual, basado en software libre, que permita superar «las frustraciones del software comercial para desarrollo de cursos» [MIT-www-2].

En estas condiciones y en un marco cambiante a medio plazo, el coste de adquisición o de pago de licencias —de medio a muy elevado— puede suponer un inconveniente añadido por el riesgo que conlleva el no poder forzar su utilización en grado suficiente como para garantizar la rentabilidad de la inversión.

En el otro extremo del espectro se encuentra el desarrollo de entornos propios. Desde hace algunos años han sido múltiples las propuestas de entornos de formación virtual desde diferentes instancias educativas. La mayor parte suelen tener su origen en proyectos llevados a cabo por un departamento o facultad determinados —de carreras Técnicas, especialmente— apuntando al doble objetivo de realizar una tarea de investigación y de disponer de una herramienta que facilite la tarea académica diaria.

Puede decirse entonces que este enfoque solventa buena parte de los inconvenientes aludidos anteriormente. Prueba de ello es que, prácticamente, cada universidad ha elaborado algún proyecto de este tipo.

Por un lado, representan una clara ventaja económica en cuanto que, incluidos en la tarea de investigación, no suponen, en principio, un desembolso añadido. Por otro, es evidente que al proceder de un departamento determinado, su utilización por los miembros del mismo estará, también en principio, fuera de toda duda. También como labor investigadora presentan notables ventajas al ofrecer la posibilidad de conocer e integrar la tecnología más avanzada.

Sin embargo, por la misma razón y dado que a veces se opta por partir de cero, el plazo de desarrollo puede llegar a ser largo y, cuando se termina, el producto suele estar hecho a medida de un departamento determinado, de sus requeri-

mientos y de su práctica docente. Aunque al disponer del código fuente de sus componentes se podría adaptar a otros departamentos, tal adaptación no sería ya rentable para el departamento de origen, una vez cubiertas sus expectativas, por lo que habría que dedicar a ello personal específico.

Alternativamente, en algún caso se ha planteado la elaboración de plataformas globales, con inclusión de todos los agentes y elementos de la vida académica: personal docente, de administración y servicios, alumnos, personal colaborador exterior, cursos, tutorías y todo tipo de servicios. Aunque atractivo en principio, el problema de este modelo suele ser su sobredimensionamiento. La migración de la estructura actual a un esquema tal no está exenta de problemas ni de costes por lo que a veces no pasa del estado de proyecto. También aquí, el dividir el problema en módulos y resolverlos por etapas, puede ser una solución más operativa.

De todo lo anterior, puede concluirse que una posible tercera vía podría apuntar a combinar las ventajas de ambos, sin sus costes respectivos, en un modelo propio de entorno centrado en el usuario final, con un tiempo mínimo de puesta en servicio, estructura modular, abierta, ágil para adaptarse a diversos requerimientos, con posibilidad de integración de elementos no previstos inicialmente, escalable, de alta operatividad y bajo coste, construido a partir de elementos basados en estándares. Los componentes que debería incluir un sistema tal son sobradamente conocidos [Gisbert et al 1997]: creación de contenidos, comunicación personal y de grupo síncrona y asíncrona, herramientas de trabajo, gestión, herramientas auxiliares ... En nuestra opinión, la mayor parte o todo ello puede realizarse satisfactoriamente, con más rapidez y menor coste, a partir de software libre o abierto.

3. Estructura funcional

El marco propuesto es el representado en la **figura 1**. Básicamente, el profesor que solicita los servicios de la DEV, además de recibir asesoramiento personalizado y apoyo específico, se convierte en usuario de una cuenta que le permite disponer de una, al menos, o varias áreas de trabajo. Para utilizar su cuenta puede emplear sus herramientas habituales (navegador, cliente de correo o FTP favoritos) o bien un entorno de trabajo de fácil utilización construido a partir de diversos elementos elaborados con software libre.

El elemento central es el «área de trabajo» que generalmente corresponderá a un curso, seminario o grupo de investigación. En resumen, a cualquier actividad diferenciada que genere unos contenidos específicos. Cada área de trabajo dispone de diversas herramientas y recursos, la utilización de los cuales queda al arbitrio del usuario.

Son recursos a su disposición una cuenta de correo (que, naturalmente, puede utilizar o no, o redirigir a cualquier otra cuenta de que disponga), un buzón específico para cada actividad, un servicio de *webmail* para acceder a esas cuentas desde cualquier sistema, una o más listas de correo de cuya administración puede encargarse directamente si así lo desea, *chat*, foro de discusión, calendario de actividades,

glosario y preguntas frecuentes (FAQ) para ese área, estadísticas de acceso a los contenidos, un buscador o una base de datos SQL que soporta tanto la información requerida por los diferentes recursos —por ejemplo, el calendario— como aquella específicamente solicitada por el usuario. La lista de recursos no está cerrada ya que cualquier usuario puede proponer la inclusión de otros nuevos que considere necesarios.

Entre las herramientas puede disponer, por ejemplo, de elaboración de cuestionarios de autoevaluación, mantenimiento de FAQ, gestión de archivos del área o la gestión de permisos de accesos a los contenidos, para los cuales el usuario puede optar por establecer acceso libre, restringido por una contraseña genérica o acceso específico para alumnos inscritos. De igual forma podrá definir si se admite la autoinscripción por parte del alumno y si es de acceso reservado para miembros de la universidad.

Todos los documentos de un área de trabajo —por ejemplo, los materiales de un curso— residen en el directorio base del mismo y son accesibles por medio de herramientas estándar (como FTP, lo que se recomienda para transferir grandes cantidades de documentos) o del gestor de archivos incorporado en el sistema. El gestor de archivos ofrece una interfaz unificada para las operaciones más comunes: crear

nuevos archivos y subdirectorios, renombrar, borrar, editar archivos de texto simple o HTML, y enviar archivos.

Ya que no se requiere estructuración previa, es el usuario el que establece la ordenación de sus materiales dentro de su espacio de trabajo y el modelo de presentación de sus contenidos. Junto al contenido elaborado por el usuario, se incluyen diversos directorios de uso interno que contienen los recursos del área: chat, FAQ, foros, etc. accesibles a través de las herramientas correspondientes.

El resultado de la actividad del área de trabajo puede tomar, a decisión del usuario, la forma de contenidos abiertos, de cursos reservados, en cuyo caso las condiciones de acceso serán las definidas anteriormente, o bien, como es el caso más frecuente, una mezcla de ambos, razón por la que el control de acceso se establece sobre la base de subdirectorios.

La entrada a los contenidos se realiza a través de una página de acceso que puede presentar cualquier aspecto, desde la más simple página web a un portal complejo con enlaces directos a recursos y contenidos. Es el usuario el que decide acerca de su mayor o menor complejidad de diseño o de los elementos a incluir, si bien, de estimarlo necesario, puede requerir el apoyo y asesoramiento especializado del personal de la DEV sobre este aspecto específico.

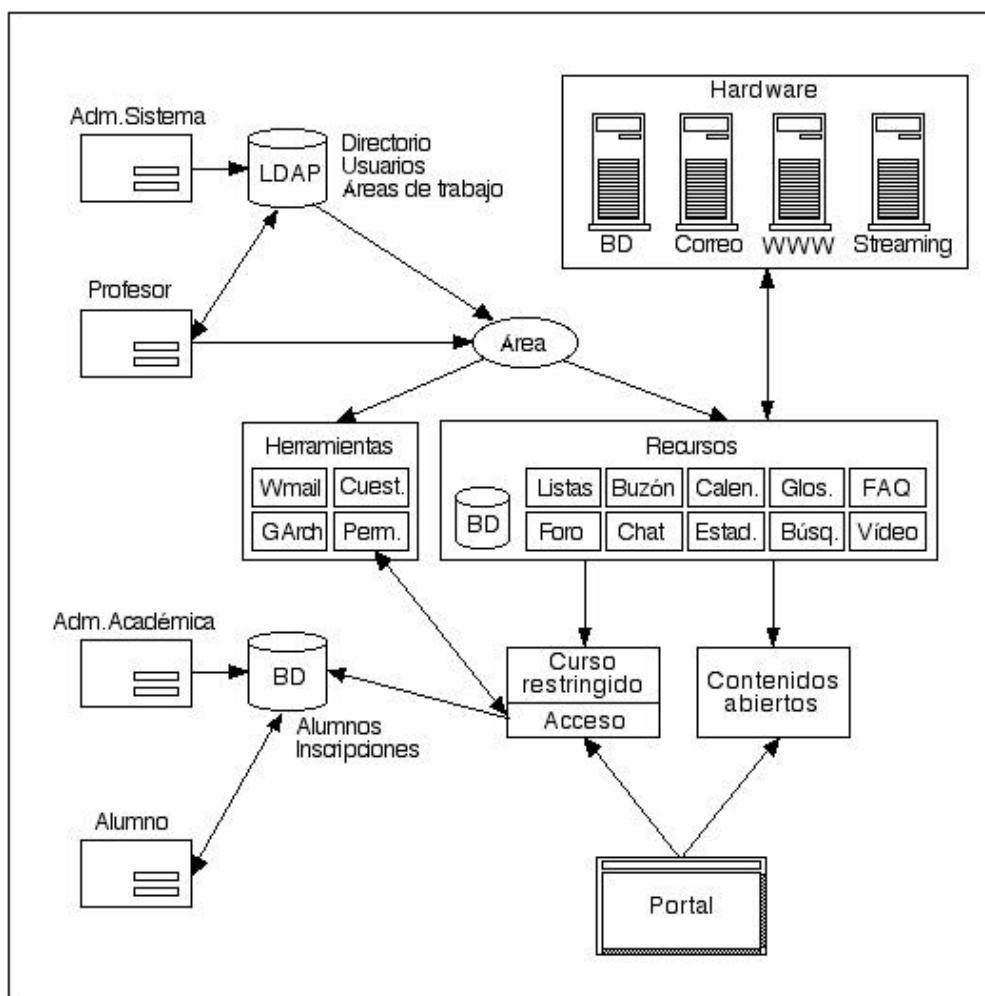


Figura 1. Estructura del DEV

Para acceder a los contenidos restringidos, el alumno, por su parte, debe estar registrado como tal e inscribirse en esa actividad, si así ha sido configurada. Esta posibilidad, que ocupa escaso tiempo para cada uno de los alumnos, descarga al profesor de la tarea, mucho mayor, de realizar decenas o cientos de inscripciones. En las inscripciones realizadas desde la UMA, el alumno debe utilizar como identificador su cuenta de correo, a donde se le enviará la confirmación de la admisión y la contraseña para el acceso. Los restantes ámbitos corresponden a la administración del sistema y a la gestión académica. En el nivel que ahora nos ocupa es tarea del administrador del sistema, entre otras, la creación de cuentas de usuario y áreas de trabajo, mientras que la gestión académica se dirige básicamente al mantenimiento de las cuentas e inscripciones de alumnos.

4. Descripción técnica

Como ya es habitual, todo el entorno se sustenta en un modelo cliente-servidor regido por protocolos estándar. El elemento habitual, aunque no único, de interacción son los formularios en HTML y aplicaciones en PHP4.

En la actualidad, el modelo contempla tres grupos de datos: un servicio de directorio LDAP para la gestión, entre otros, de usuarios (profesores) y grupos de trabajo, una base de datos SQL centralizada para la gestión de alumnos e inscripciones, y bases de datos, también SQL, para utilización por cada área de trabajo.

Puesto que ya se disponía con anterioridad de un servicio de directorio LDAP para la autenticación de usuarios, se ha optado por mantenerlo en dicha forma toda vez que hasta el momento se ha mostrado como una buena alternativa para la autenticación en entornos mixtos, particularmente adecuada para el acceso a datos de baja volatilidad que requieren pocas actualizaciones y muchas consultas, además de constituir una opción con futuro [López Murcia 2001]. Si fuera necesario porque la carga del sistema lo exigiera o por cualquier otro motivo, nada impediría, por otra parte, utilizar como *backend* un servidor SQL. Por razones de seguridad, el servicio maestro de directorio se replica en una segunda máquina esclava.

Por otras razones parece aconsejable el mantener una base de datos administrativa separada: de una parte, la UMA es una universidad con una clara orientación presencial. La enseñanza «virtual», si se dirige al alumnado propio, no deja de ser una actividad todavía complementaria. De otra, es preciso contemplar también la posibilidad de contenidos abiertos —sin requisitos de inscripción— y, como ya se ha hecho en varias ocasiones, de cursos dirigidos a participantes no pertenecientes a la comunidad universitaria que verían vetado su acceso si se utiliza como fuente de registro la BD general de la universidad, tal como requieren algunas plataformas comerciales. De ser necesario, siempre sería posible en el futuro establecer la conexión adecuada.

Por último, el mantener bases de datos separadas para cada

área de trabajo se debe a razones de modularidad, seguridad y simplicidad de mantenimiento.

Los elementos software utilizados para construir este entorno son ampliamente conocidos: el sistema operativo para todos los servidores es Debian GNU/Linux, elegido, después de probar varias alternativas, por su estabilidad, facilidad de actualización y carácter multiplataforma (actualmente está en uso en máquinas Intel y PowerMac), que proporciona la funcionalidad básica mientras que para la mayor parte de las aplicaciones de los servicios se ha optado por instalarlas desde código fuente, por ofrecer mayores posibilidades de adaptación y control. Cada cliente en la DEV, por su parte, utiliza el sistema operativo de su elección: Linux, Mac OS o Windows.

En cuanto a los servicios, el servidor HTTP es Apache con diversos módulos, entre ellos SSL y autenticación contra LDAP y PostgreSQL. Como analizador de *logs* empleamos una versión ligeramente modificada de Webalizer, de amplia aceptación entre los usuarios por su características gráficas, y HtDig como motor de búsqueda. Aunque se han realizado algunos *scripts* en Python y se utilizan otros en Perl, el lenguaje de elección es PHP4 por su carácter de módulo de Apache, facilidad de uso, orientación a objetos y rapidez de desarrollo, además de disponer de una amplia biblioteca de software libre o abierto capaz de resolver casi cualquier problema imaginable, todo lo cual proporciona una excelente relación coste/rendimiento.

Como MTA, y tras utilizar SendMail durante algún tiempo, se ha optado por Postfix con Procmail; como webmail se empezó utilizar NOCC por su rapidez de instalación y sencillo interfaz, si bien se están estudiando diversas alternativas, como Postman [López, Rulot 2001], a fin de migrar en un futuro próximo a un sistema de mayor rendimiento. GNU/Mailman como servidor de listas, OpenLDAP como servicio de directorio LDAP, ProFTPd como demonio FTP o Darwin Streaming Server, distribuido por Apple como código abierto, son otros componentes de este entorno.

A todo ello hay que añadir herramientas menores solicitadas específicamente por determinados usuarios, tales como utilidades sencillas para mantenimiento de FAQs o los generadores de cuestionarios del Mellon Language Project [Mellon-www], y la múltiples utilidades de uso interno necesarias para la gestión y mantenimiento del sistema.

En el momento presente, la mayor parte de los módulos de recursos y herramientas están ya disponibles para los usuarios si bien la integración efectiva de algunos de ellos todavía requiere una activa participación de los gestores del sistema. Por ello, el desarrollo actual se dirige —una vez establecidas la herramientas básicas— a aumentar la capacidad de autogestión del profesor usuario, a mejorar la interfaz de acceso y a la integración de una herramienta de gestión de alumnos, también basada en software libre, elaborada por miembros del SCI para el Departamento de Economía Aplicada (Matemáticas). Las posibilidades de ampliación son, por lo demás, infinitas ya que difícilmente se puede

poner hoy en cuestión el dinamismo de la comunidad del software libre y abierto que ha puesto a disposición de desarrolladores y usuarios miles de componentes software.

En lo que se refiere al hardware, la libertad de elección es absoluta. La opción de repartir las diversas tareas en distintas máquinas permite optimizar los recursos disponibles y aparece como la más práctica, segura y de mayor rendimiento. En nuestro caso, se dispone de cinco máquinas Intel que, en líneas generales, se reparten las tareas de la siguiente forma: la primera de ellas, con SMP, atiende el servidor HTTP y todos los servicios que le están directamente ligados; la segunda, también con SMP, actúa como servidor de *streaming* para vídeo bajo demanda; la tercera, atiende los servicios relacionados con el correo y las listas de distribución; la cuarta, soporta el servidor LDAP, las bases de datos y actúa como servidor de backup nocturno, y la quinta, además de servir como réplica del directorio LDAP, se utiliza como plataforma de pruebas antes de poner un nuevo elemento en producción. Todas las máquinas disponen de tarjetas de interfaz de red 3Com a 100 MB que las conectan al conmutador CISCO que une la subred de la DEV al resto de la red de la UMA.

Naturalmente, este esquema es sólo uno de los muchos posibles ya que la arquitectura cliente-servidor y la utilización de protocolos estándar de comunicación entre los diferentes servicios permite que se puedan migrar con facilidad a otras máquinas a fin de optimizar el rendimiento global.

5. Evaluación

Ya que no constituye un producto «llave en mano» como las plataformas comerciales, no cabe duda de que el entorno presentado requiere una cuidadosa tarea de análisis, planificación y selección. En contrapartida, satisface plenamente, en nuestra experiencia, los requisitos antes expuestos:

Modularidad: cada recurso o herramienta es independiente de las demás, aunque accedan a datos comunes. El esfuerzo de integración de una nueva herramienta se reduce, desde un punto de vista funcional, en la mayoría de los casos, a incorporar el módulo de autenticación común previamente desarrollado y, en su caso, modificar el acceso a los datos por medio de una capa de abstracción en caso de que el módulo no disponga ya de ella, abstracción que debe facilitar la tarea de migración a otra BD si en algún momento fuese necesario.

Estrechamente vinculadas a la estructura modular se encuentran otras dos características:

Escalabilidad: la sustitución de un elemento software por otro de prestaciones más adecuadas puede hacerse, generalmente, sin mayores problemas. Como muestra, el servidor de listas de correo ha pasado sucesivamente desde un básico MailShare en Macintosh PowerPC a Majordomo sobre Linux, SmartList y, finalmente, GNU/Mailman, sin mayores molestias –e incluso pasando desapercibido– para los usuarios. En lo que respecta al hardware, gracias a la independencia de los módulos, no representa problema alguno

migrar servicios, ahora agrupados, a máquinas distintas o de superiores prestaciones cuando los recursos lo permiten (por ejemplo, razones de seguridad pueden aconsejar instalar el servicio de *chat* en una máquina separada). Este ha sido el procedimiento seguido para ir incorporando los sucesivos equipos.

Versatilidad: como se deduce de lo anterior, es adaptable a distintas situaciones, con diferentes requerimientos y con diversos grados de recursos y equipamiento. Para el usuario final es un modelo válido tanto si sólo se desea poner una página web en Internet como si se pretende elaborar un portal de complejo diseño con múltiples recursos y servicios. Puede construirse tan libre o restrictivo como se quiera y tan específico como se necesite, y su capacidad de adaptación es, en todo caso, muy superior a la de las plataformas comerciales, lo que resulta de inapreciable valor en un contexto con vocación prospectiva y con un alto grado de retroalimentación por parte de los usuarios.

Por último, el empleo de protocolos y herramientas estándar representa una garantía para la inversión, tanto en equipamiento como en horas de trabajo, al tiempo que la modularidad alarga la vida útil del hardware al permitir su reciclaje en tareas de menor carga. Para el usuario final, la utilización de estándares representa la posibilidad y la libertad de poder seguir utilizando sus herramientas habituales sin necesidad, por ejemplo, de tener que recurrir a clientes de correo específicos de una u otra plataforma.

6. Conclusión

Iniciativas como OpenCourseWare [MIT-www-3], (no confundir con la antes citada OKI, también del MIT), vienen a poner de relieve que, para las instituciones más avanzadas, su principal activo no es tanto el contenido como el capital humano. En un contexto tal, se hace patente la conveniencia de optar por entornos de trabajo que primen el servicio a los usuarios sobre la mera comercialización de contenidos [Ishii, Lutterbeck 2001].

Mientras que las herramientas comerciales, cerradas, pueden resultar útiles en ambientes de menor tamaño u homogéneos, o donde la elaboración de una plataforma a medida supondría un coste excesivo, el entorno aquí presentado propone un modelo de trabajo plural, abierto a la colaboración, y con la agilidad y elevada capacidad de respuesta que requiere una comunidad como la académica, amplia, con una rica variedad de prácticas, contenidos, grados de motivación y preparación, perspectivas e intereses.

No hay que satisfacer grandes requisitos ni la superación de una larga curva de aprendizaje para empezar a «ver resultados», lo que, unido a la capacidad de personalización y adaptación a los gustos y preferencias específicos del usuario y a la posibilidad de ver integradas sus propias aportaciones, constituye, todo ello, un incentivo para su utilización.

La utilización de software libre y abierto permite elaborar con relativa rapidez y facilidad un entorno modular propio

absolutamente abierto –a diferencia de las plataformas comerciales–, pero sin los costes de un desarrollo a medida desde cero, con una rápida puesta en servicio, óptima relación costo/rendimiento y mínimo riesgo, que estimula la colaboración de los usuarios y, con ello, la generación de valor añadido.

7. Referencias

- [MIT-www-1] <http://web.mit.edu/oki/oki_library/LMSgood_pedagogy.ppt> (Consultado: diciembre 2001)
- [MIT-www-2] «Universities Begin Creating a Free, Open-Source Course-Management System». <En <http://web.mit.edu/oki/news.html>> (Consultado: diciembre 2001)
- [Gisbert et al 1997] **Gisbert, M; Adell, J.; Anaya, L.; Rallo, R.**: «Entornos de formación presencial virtual y a distancia». Boletín de RedIRIS, nº 40, octubre 1997. También: Britain, Sandy; Liber, Oleg: A Framework for Pedagogical Evaluation of Virtual Learning Environments. <<http://www.jtap.ac.uk/reports/htm/jtap-041.html>> (consultado: noviembre 2001)
- [López Murcia 2001] **López Murcia, Alfonso**: «Servicios basados en LDAP: Migración desde X.500», en Boletín de RedIRIS, nº 54-55, enero 2001.
- [López, Rulot 2001] **López Bueno, J.A.; Rulot, H.**: «POSTMAN: un cliente Webmail de alto rendimiento», en Boletín de RedIRIS, nº 54-55, enero 2001.
- [Mellon-www] <<http://lang.swarthmore.edu/mellon>>. (Consultado: noviembre 2001)
- [MIT-www-3] El pasado mes de junio, el MIT anunció su decisión de hacer todos sus contenidos libremente accesibles en la web. <<http://web.mit.edu/ocw/>> (Consultado: noviembre 2001)
- [Ishii, Lutterbeck 2001] Acerca de la viabilidad de propuestas similares, puede verse: Ishii, Key; Lutterbeck, Bernd: «Unexploited resources of online education for democracy - Why the future should belong to OpenCourseWare», en First Monday, noviembre, 2001. <http://www.firstmonday.org/issues/issue6_11/ishii/index.html> (Consultado: diciembre 2001)