

**Sumario**

<i>Editorial</i>	3
<i>Domótica</i>	4
<i>Software Libre</i>	6
<i>Integridad de datos relacionales</i>	10
<i>Primera Jornada de Testeo</i>	14
<i>Noticias Breves</i>	16
<i>Oferta y Demanda Tecnológica</i>	17
<i>Ayudas y Subvenciones</i>	18

**EDITA:**

ITI – Instituto Tecnológico de Informática

Universidad Politécnica de Valencia  
Camino de Vera s/n  
46071 ValenciaTel.: 96 387 70 69  
Fax: 96 387 72 39  
<http://www.iti.upv.es>  
e-mail: [actualidadtic@iti.upv.es](mailto:actualidadtic@iti.upv.es)**DISEÑA:**domino publicidad  
<http://www.dominopublicidad.com>**IMPRIME:**GRUPO QUATREMEDIA  
<http://www.quatremedia.com>  
Tel.: 902 903 987**Depósito Legal:** V-3279-2003**ISSN:** 1696 - 5876

# Actualidad

**Boletín Trimestral del Instituto Tecnológico de Informática, dedicado a las Tecnologías de la Información y las Comunicaciones.**Número > 4  
Junio 2004**Formación Continua ITI 2004****JULIO**

Programación Orientada a Objetos: Lenguaje JAVA- 30 h\*

Del 5 al 14 de julio. De lunes a jueves, de 16:00 a 20:00 h.

JAVA: Bases de Datos y RMI -30h\*

Del 19 al 28 de julio. De lunes a jueves de 16:00 a 20:00 h.

**SEPTIEMBRE**

Administración de Sistemas LINUX- 30h\*

Del 6 al 15 de septiembre. De lunes a jueves de 16:00 a 20:00 h.

Seguridad de los Datos en la Empresa-30h\*

Del 6 al 15 de septiembre. De lunes a jueves de 9:30 a 13:30 h.

Bases de Datos Relacionales y el lenguaje SQL. - 30 h

Del 20 de septiembre al 06 de octubre. Lunes, miércoles y viernes de 16.00 a 20.00 h.

**OCTUBRE**

Desarrollo de aplicaciones para dispositivos móviles con J2ME.- 30h

Del 13 de octubre al 10 de noviembre. Lunes y miércoles, de 16.00 a 20.00 h.

**NOVIEMBRE**

Java: Servlets y JSPs.- 20h

Del 15 al 19 de noviembre. De Lunes a viernes, de 16.00 a 20.00 h.

Programación Gráfica 2D/3D con Java.- 30h

Del 22 de noviembre al 02 de diciembre. De Lunes a jueves, de 16.00 a 20.00 h.

\* Cursos subvencionados al 85% por IMPIVA y FSE

Más información: [otri@iti.upv.es](mailto:otri@iti.upv.es)

## Editorial

**En este cuarto número de la revista Actualidad TIC continuamos ofreciendo contribuciones técnicas, información y novedades a nuestros asociados. En el ánimo de reforzar el carácter de vehículo de comunicación de esta publicación, queremos reiterar la permanente invitación a las empresas asociadas para que presenten artículos técnicos representativos de su actividad y la del sector.**

Como nuevo director científico-técnico del ITI es una satisfacción presentar este cuarto número de ActualidadTIC. Desde este espacio quiero manifestar mi objetivo de ayudar en lo posible a mantener la línea de actuación del Instituto y a potenciar las actividades que en él se vienen realizando.

No puedo soslayar una referencia a la magnífica labor realizada por el anterior director científico, José Bernabéu, que desde el inicio mismo de la andadura del Instituto contribuyó de forma crucial a la transformación de la idea original de creación de una entidad al servicio del sector informático, investigación, desarrollo y transferencia de tecnología, en una realidad activa y eficiente, como es el ITI en la actualidad. Este reconocimiento ha de hacerse extensivo, evidentemente, a Laura Olcina que, como he podido constatar en estos primeros meses en la dirección científica, desde la gerencia realiza cada día un extraordinario trabajo para que el Instituto siga aumentando en un rápido progreso su presencia en el sector y mantenga esa vocación de revulsivo de la innovación y servicio del más alto nivel en el ámbito tecnológico para las empresas de nuestro país y en particular para nuestros asociados. También destacaría la magnífica impresión que he recibido en cuanto a la profesionalidad y entusiasmo del personal y los colaboradores científicos y técnicos que constituyen la verdadera fuerza motriz del ITI.

Las colaboraciones técnicas de este número de la revista abordan aspectos muy interesantes y representativos de las tecnologías de la información y particularmente de las áreas de interés del Instituto y sus empresas asociadas, como son el software libre, el mantenimiento de la integridad en bases de datos, la domótica y la calidad del software.

Podemos encontrar en este número un primer artículo descriptivo y sugerente sobre la tecnología domótica, en el que se da un repaso al mundo de la automatización doméstica que sigue siendo hoy en día un sector en desarrollo con grandes posibilidades.

El software libre está cobrando creciente importancia en la sociedad y prueba de ello es la atención que le están prestando las administraciones públicas, así como el aumento del uso de soluciones de fuentes abiertas en los entornos corporativos. En el Instituto

hemos mantenido durante estos diez años una actitud clarísima a favor del software abierto, indudablemente beneficioso en muchos casos, en nuestro particular entorno tecnológico. En el artículo de este número pueden quedar resueltas muchas dudas respecto de este curioso fenómeno que tanta influencia ha ejercido en la última década y probablemente ejercerá en los años próximos.

Por otro lado, como sabemos, las bases de datos son el corazón de algunas de las aplicaciones informáticas más importantes y, sobre todo, más críticas. El mantenimiento de la integridad de los datos almacenados en ellas resulta por tanto importantísimo. Este extremo concreto y algunas soluciones propuestas desde el ITI se analizan en la segunda colaboración técnica.

Finalmente, en una última sección técnica encontramos un relato de las impresiones obtenidas y de los contenidos desarrollados en las "Primeras Jornadas de Testeo de Software" organizadas por el Instituto y acogidas con un extraordinario entusiasmo por parte de las empresas y profesionales participantes. Un notable acicate que nos ayudará a seguir impulsando en el futuro inmediato la adopción de tecnologías y procesos que contribuyan a la mejora de la competitividad y eficiencia de las empresas de nuestro entorno.

**Juan Carlos Pérez**

**Director Científico-Técnico  
del Instituto Tecnológico  
de Informática**

## Domótica

*Los sistemas domóticos incorporan a las viviendas las últimas tecnologías informáticas y de comunicaciones, proporcionando cuantiosas ventajas a sus usuarios. Estas tecnologías ayudan a reducir consumos energéticos y a aumentar la seguridad, las posibilidades de comunicaciones y sobre todo, la comodidad. Sin embargo, la situación y cultura actuales conceden poco optimismo a la domótica. Desde el Instituto Tecnológico de Informática se está investigando en tecnologías que se podrían incorporar a los sistemas domóticos para mejorar los servicios ofrecidos y facilitar su penetración en el mercado.*

### Introducción

La palabra domótica deriva de la unión de domus (casa) y de Informática, y hace referencia a la incorporación a la vivienda de un conjunto de tecnologías informáticas y de comunicaciones que permiten gestionar y automatizar desde un mismo sistema las diferentes instalaciones de uso cotidiano en una vivienda, proporcionando una mejor calidad de vida de los usuarios de la misma y una mejor conservación y cuidado del edificio.

Las primeras iniciativas en el mundo de la domótica intentaron abrirse paso en el mercado resaltando aspectos superficiales, que realmente no aportaban valor añadido al usuario, como podía ser el mecanizado de ciertos elementos de la vivienda (persianas, riego, iluminación) o la posibilidad de “jugar” con ellos. Actualmente, con una tecnología mucho más madura y con una nueva visión de las necesidades a cubrir, cuatro aspectos fundamentales permiten justificar la inversión en una instalación domótica:

#### 1. Ahorro energético

Cada vez son más los aparatos eléctricos que se incorporan a la vivienda, de forma que el consumo de energía puede llegar a ser importante. Mediante un sistema domótico es posible implementar mecanismos que regulen y optimicen dicho consumo, como el control de la climatización y regulación de la temperatura por zonas; la utilización de electrodomésticos en tarifa nocturna; la iluminación por detección de presencia; el riego controlado por sensor meteorológico; la desconexión automática de dispositivos, etc.

#### 2. Seguridad

La evolución y el crecimiento económico del “mundo occidental” en los últimos años ha hecho que poco a poco se incorporen a las viviendas elementos más sofisticados. Para asegurar la integridad de las personas y de los edificios, una instalación domótica puede proporcionar mecanismos como detección de intrusos; simulación de presencia; conexión con centrales de alarma; alarmas de salud o alertas médicas (teleasistencia); alarmas técnicas: Incendios, fugas de agua o gas; control de accesos, etc.

#### 3. Comunicaciones

Las comunicaciones juegan un papel fundamental en el momento actual, y constituyen un elemento estratégico

**La domótica mejora la comodidad, seguridad y ahorro de energía.**

para el desarrollo de servicios impensables hace algunos años. Mediante un adecuado sistema de comunicaciones integrado con el sistema domótico es posible establecer comunicaciones en el interior del edificio, desde el interior hacia el exterior y viceversa. Por ejemplo, se podría distribuir imágenes y sonido por el interior del edificio; manejar internamente el sistema a través de mando a distancia; enviar alarmas y señales hacia el exterior; realizar un control remoto del sistema a través de teléfono fijo, móvil, PDA, Internet, etc.

#### 4. Confort y comodidad

Es un hecho constatado que la calidad de vida actual es muy superior a la de un par de generaciones anteriores. El simple hecho de automatizar los elementos del edificio y poder gestionarlos de forma remota proporciona unos niveles de comodidad hasta ahora desconocidos. Con una instalación domótica el usuario se libera de invertir tiempo y energía en realizar acciones mecánicas y cotidianas y de preocuparse por aspectos que el sistema resuelve automáticamente. Simulación de escenarios; automatización y programación del riego; regulación de temperatura; atención particularizada por persona o control remoto y a distancia son una pequeña muestra de las numerosas posibilidades con las que la domótica mejora la comodidad de las personas.

Probablemente el confort sea el aspecto más valorado de un sistema domótico para usuarios residenciales, pues es el que perciben directamente, y también sea el que puede llevar a pensar en aquella idea inicial de “domótica = juguete”, que debe ser abandonada, pues la domótica, como se ha visto, es mucho más que eso. Es una forma de alcanzar comodidad, seguridad, ahorro de energía e incluso, por qué no, felicidad.

Sin embargo, y pese a las ventajas que reporta a las personas en el interior de los edificios, la domótica todavía no ha calado en el mercado español, en pleno “boom inmobiliario” en el que por un lado, los promotores de nuevas viviendas no tienen la necesidad de añadir una instalación domótica, pues la demanda las acepta

sin ese valor añadido, y por otro lado, los usuarios finales, que todavía no han asimilado los rápidos avances tecnológicos y la incorporación a la Sociedad de la Información, se muestran escépticos y temerosos ante problemas de manejo, coste, mantenimiento y averías del sistema y no perciben una verdadera necesidad de incorporarlo en sus viviendas. Es por ello que actualmente está limitada a edificios inteligentes (no residenciales) y a unos pocos privilegiados que ven en la domótica una forma de diferenciarse del resto.

La disminución de la demanda de viviendas, unida a una mayor penetración de las tecnologías en la vida cotidiana, harán que a medio-largo plazo la domótica se perciba como un valor añadido al edificio y como una necesidad para sus usuarios, y ese será el momento en el que despegue definitivamente, abaratando costes y mejorando servicios.

## ITI y Domótica

El Instituto Tecnológico de Informática desarrolla labores de investigación en áreas que pueden ser de gran ayuda para las instalaciones domóticas. Sistemas de reconocimiento de patrones, sistemas de distribución de audio y video, sistemas operativos a la medida o sistemas adaptativos complejos son algunos ejemplos de nuestras investigaciones en este campo.

### 1. Reconocimiento de patrones

El desarrollo de sistemas inteligentes basados en técnicas de reconocimiento de patrones y algoritmos de optimización es una de las áreas del Instituto que más puede aportar al mundo de la domótica. Dentro de este grupo, la rama dedicada al procesamiento del habla es capaz de desarrollar un sintetizador de voz y un sistema de reconocimiento continuo de voz que se puede introducir en una instalación domótica para proporcionar al usuario un control del sistema a través de la voz. El sistema parte de un conjunto de órdenes preestablecidas a las que obedece y puede diferenciar entre la voz de varios usuarios de forma que puede servir también para realizar un control de acceso de usuarios al sistema.

La rama dedicada al reconocimiento de imágenes ofrece sistemas de reconocimiento facial y dactilar, que aportan seguridad en los accesos a recintos. En el primer caso, el sistema almacena en una base de datos un conjunto de caras conocidas. Una cámara a la entrada del edificio graba la imagen facial y la contrasta con la base de datos usando un algoritmo de optimización, permitiendo o denegando el acceso al recinto. Cuando los sistemas domóticos son gestionados a través de dispositivos PDA, se puede introducir un sensor de huellas dactilares que proporcione el acceso al control del sistema.

### 2. Distribución de Audio y Vídeo

Uno de los cuatro pilares básicos de la domótica es el aumento de las comunicaciones tanto en el interior del edificio como hacia y desde el exterior. En el instituto

hemos desarrollado un sistema que permite recibir canales de televisión y audio digital, almacenarlos y distribuirlos desde un servidor a varios clientes repartidos por el edificio formando una red de área local. También permite la reproducción de DVD y otras muchas opciones. Así, por ejemplo, podría servir para proporcionar video bajo demanda a cada una de las habitaciones de un hotel o para almacenar canales de televisión con calidad óptima (DVD).

### 3. Sistemas operativos a medida

Desde el Instituto siempre se ha intentado defender el uso de herramientas de Software Libre y GNU/Linux como sistema operativo y por ello se desarrollan distribuciones de este sistema a la medida de los clientes. Actualmente las instalaciones domóticas cuentan con un sistema informático que controla los elementos de la vivienda. Este sistema informático se basa por lo general en un sistema operativo propietario (no libre) por el que se tiene que pagar una licencia de uso y del cual no se conoce realmente su funcionamiento interno, lo que hace difícil adaptarlo a las necesidades propias de cada aplicación. Una forma abaratar el coste y al mismo tiempo conseguir un sistema optimizado, robusto y estándar adecuado a las necesidades de cada tipo de instalación domótica sería utilizar sistemas basados en Software Libre diseñados a la medida.

### 4. Sistemas adaptativos complejos

Dentro de este grupo se llevan a cabo investigaciones para desarrollar sistemas que permitan controlar dispositivos simplemente pensando en ello (véase el número anterior de ActualidadTIC). Si se piensa en personas discapacitadas queda clara la utilidad de este sistema que, por ejemplo, permitiría controlar una silla de ruedas, el ratón en una pantalla de ordenador o el propio sistema domótico completo a través de la mente.

## Conclusión

Aunque todavía no esté muy extendida, las ventajas y la mejora de calidad de vida que la domótica aporta harán que su integración en las viviendas españolas sea la norma general a medio plazo.

Para que ese momento llegue en el menor tiempo posible, es necesario que los usuarios perciban la necesidad real de usar un sistema domótico. En este sentido, iniciativas como DOMOGAR son muy positivas. Al mismo tiempo, la incorporación de las últimas tecnologías informáticas y de comunicaciones a estos sistemas mejorará sus prestaciones y eficiencia, y por este lado, las investigaciones realizadas en el Instituto son sin duda una pieza fundamental.

Autor: Daniel Sáez

Más información: [otri@iti.upv.es](mailto:otri@iti.upv.es)

## Software Libre

En este artículo se presentan los conceptos fundamentales relacionados con el software libre: qué es, qué no es, quién produce software libre, qué licencias existen, dónde hay software libre, quién y cómo lo usa, etc. También se tratan aspectos prácticos: qué tipos de negocio se pueden hacer, cuáles son las ventajas de software libre y cuáles sus inconvenientes. Las diversas direcciones web incluidas se pueden utilizar como punto de partida para obtener más información acerca de todos estos temas.

### ¿Qué es el software libre?

Software libre es aquel software cuyos términos de uso permiten al usuario ejecutarlo, copiarlo, distribuirlo, estudiarlo y mejorarlo. Además, el usuario tiene la posibilidad de redistribuir sus modificaciones de manera que otras personas puedan aprovecharlas y, a su vez, modificarlas y distribuirlas. Para ello, el principal requisito que se debe cumplir es garantizar el acceso al código fuente, tanto del programa original como de las modificaciones posteriores.

Hay quien confunde *software libre* con *software gratuito*. Esta confusión está motivada generalmente por la ambigüedad del término *free software* original. En la práctica son dos conceptos distintos y a menudo complementarios.

Existe otro término, *código abierto*, que habitualmente se utiliza como sinónimo de *software libre*. Aunque existen diferencias entre los dos términos, a nivel práctico ambos vienen a expresar la misma idea.

Otro concepto fundamental es el de *licencia*. Una licencia de software libre es un conjunto de *derechos y deberes* a los cuales se acogen tanto los desarrolladores de un programa como sus usuarios. Existen muchas licencias distintas de software libre. Las más conocidas han sido propuestas por asociaciones, empresas o grupos de desarrolladores o usuarios. Aunque generalmente difieren en algunos términos, todas ellas coinciden en algunos puntos esenciales como, por ejemplo, el libre acceso al código fuente del programa.

Algunas licencias permiten que un usuario tome un cierto programa y añada a la licencia original ciertas restricciones de su interés. En cambio otras licencias obligan al usuario a ofrecer los mismos derechos que ha recibido. En estos casos, se dice que la licencia es de tipo *copyleft*. La idea principal que hay detrás de estas licencias no es otra que la de fomentar en el mayor grado posible la libre difusión y evolución del software libre.

El antónimo de software libre es *software propietario* o *software privativo*. En general, es aquel software que no es libre. El término *privativo* se utiliza para enfatizar el hecho de que este tipo de software *priva* al usuario de las libertades ofrecidas por el software libre.

Otra de las confusiones habituales se da entre los términos *software propietario* y *software comercial*. Habitualmente, el software propietario es software comercial, en tanto en cuanto se compra por un determinado precio. Sin embargo no todo el software comercial es propietario. Existe software libre que también se vende por un precio u otro. Esto es posible porque las licencias de software libre no suelen imponer restricciones a este respecto.

Tampoco es cierto que todo el software propietario sea comercial. Valga como ejemplo el software distribuido como *freeware*. Este es propietario, ya que el autor

El concepto de software libre no debe confundirse con el de software gratuito.

no ofrece el código fuente públicamente, pero no es comercial, ya que el autor no cobra ningún dinero por él.

#### Términos y clasificación del software:

<http://www.fsf.org/philosophy/categories.html>

#### Relación de términos:

<http://www.fsf.org/philosophy/words-to-avoid.html>

### ¿Quién está detrás?

Una de las mayores y más importantes comunidades defensoras del software libre es la *Free Software Foundation* (FSF), creada en 1985 por Richard M. Stallman para promocionar el software libre y los derechos relacionados de los usuarios.

La FSF mantiene una definición de software libre basada en cuatro libertades que el software debe ofrecer al usuario para que pueda ser considerado libre:

- Libertad de ejecutar el software
- Libertad de acceder al código fuente, estudiarlo y adaptarlo a sus necesidades particulares
- Libertad de distribuir copias del software
- Libertad de mejorar el software y distribuir las modificaciones a otros usuarios

Otra de las entidades importantes es la *Open Source Initiative* (OSI), creada en 1998 y dirigida por Eric S. Raymond. La OSI mantiene su propia definición de software libre (aunque utiliza preferentemente el término *código abierto*).

Además mantiene un catálogo de licencias libres compatibles con dicha definición, correspondientes a empresas, asociaciones, etc. Actualmente esta lista cuenta con más de cincuenta licencias distintas.

La FSF también mantiene una lista de licencias libres y una lista adicional de licencias que no pueden ser consideradas libres según la filosofía de la FSF.

#### Free Software Foundation (FSF):

<http://www.fsf.org>

#### Open Source Initiative (OSI):

<http://www.opensource.org>

## Software Libre

### Licencias ampliamente difundidas

La FSF definió dos de las licencias de software libre más utilizadas en todo el mundo: la GNU GPL (*General Public License*) y la GNU LGPL (*Lesser General Public License*, inicialmente conocida como *Library General Public License*).

La licencia GPL se utiliza en programas de uso cotidiano en todo el mundo: Linux (núcleo del sistema operativo GNU/Linux), Gnome (entorno de ventanas), Emacs (editor de texto), MySQL (gestor de bases de datos), GCC (colección de compiladores), CVS (sistema de control de versiones), etc.

La licencia LGPL es una variante de la anterior, utilizada en aquellos casos en que la licencia GPL resulta ser demasiado restrictiva. Concretamente, la GPL obliga a seguir usando esa misma licencia en los programas (publicados y distribuidos) que *usen* o estén basados en programas publicados bajo la licencia GPL. Así, los autores de software propietario no podrían usar ningún software libre publicado bajo GPL pues estarían violando la licencia, por lo que se verían obligados a usar algún otro software, tal vez también propietario. La LGPL evita este problema a la vez que permite la mayor difusión posible del software libre.

Otras licencias muy utilizadas son BSD License (versiones inicial y modificada), Apache License y Apache Software License, Apple Public Source License (APSL) o Common Public License.

Otros ejemplos de amplia difusión podrían ser las licencias directamente relacionadas con un producto o conjunto de productos concretos como las licencias de Perl, Python, PHP, Mozilla o Netscape. En el caso de licencias de lenguajes de programación, a menudo los desarrolladores deciden aplicar esa misma licencia a los productos que desarrollan.

### ¿Dónde encontrar software libre?

Existen algunos sitios en la web dedicados exclusivamente a albergar proyectos de software libre.

Uno de los más conocidos y utilizados es SourceForge, con más de 845000 usuarios registrados y 80000 proyectos, entre los que se encuentran JBoss (servidor de aplicaciones J2EE), vim (editor de texto), Compiere (gestor ERP y CRM) o Gaim (sistema de mensajería).

Este lugar no solo sirve de almacén del software creado por los desarrolladores sino que da otros servicios como el alojamiento de la página web de cada proyecto y una gran cantidad de recursos a los desarrolladores (repositorio CVS, listas de correo, listas de *bugs* y parches, gestor de documentación, gestor de tareas, etc.).

Otro de los repositorios de software libre más utilizados es Freshmeat, que alberga cerca de 33000 proyectos y reúne a más de 272000 usuarios.

Existe otro portal muy conocido, Savannah, creado y mantenido por la FSF. De hecho sirve de repositorio de los programas del proyecto GNU. Además aloja los proyectos creados por otros usuarios particulares. Actualmente tiene cerca de 2000 proyectos y 27000 usuarios registrados.

En España, existe un portal similar a los anteriores,

software-libre.org, de muy reciente creación, impulsado y mantenido por la asociación Hispalinux.

#### SourceForge:

<http://www.sourceforge.net>

#### Freshmeat:

<http://freshmeat.net>

#### Savannah:

<http://savannah.gnu.org>

#### software-libre.org:

<http://software-libre.org>

### Uso actual del software libre

En sus comienzos, el software libre fue especialmente utilizado en universidades y en centros de investigación, no solo por el personal docente e investigador sino también por parte del alumnado.

Uno de los primeros sectores en utilizar software libre de manera destacable fue el del desarrollo de software y servicios relacionados, debido principalmente a que inicialmente una buena parte del software libre disponible estaba orientado precisamente a este tipo de tareas (por ejemplo, bibliotecas de código, lenguajes de programación, editores de texto, entornos de desarrollo, etc.).

Con el tiempo este escenario ha cambiado. Actualmente existen programas libres prácticamente de cualquier tipo. Así encontramos tanto programas genéricos (navegadores de web, paquetes de ofimática, editores gráficos, gestores de bases de datos, etc.) como más específicos y orientados a temas concretos (bolsa, medicina, arquitectura, ingenierías, etc.), así como herramientas para la gestión de servicios y establecimientos comerciales (supermercados, librerías, bibliotecas, restaurantes, etc.). Por ello el software libre está llegando a cada vez más sectores de la sociedad, aunque en determinados entornos y colectivos su grado de implantación es todavía reducido, como por ejemplo, las administraciones públicas, y los sistemas personales de una gran mayoría de usuarios.

Últimamente estamos presenciando un cambio en dirección al software libre en el entorno de las administraciones públicas y gobiernos. Durante el año pasado, varios países de latinoamérica tomaron la decisión de implantar software libre en los organismos públicos. En Europa, Alemania fue uno de los países pioneros en decidir implantar software libre en entornos públicos.

En España, últimamente se está avanzando de forma importante en este sentido. Por una parte los gobiernos de varias comunidades autónomas han decidido implantar progresivamente el uso de software libre en las administraciones y organismos públicos.

Además, varias comunidades autónomas están desarrollando sus propias distribuciones del sistema operativo GNU/Linux, orientadas en algunos casos al sector de la educación, o a cualquier usuario en el caso más general. Algunos ejemplos son gnuLinEx de Extremadura, GuadaLinux de Andalucía, MAX de Madrid o Lliurex de la Comunidad Valenciana.

Es de esperar que iniciativas de este tipo sirvan para

## Software Libre

dar a conocer a los usuarios la filosofía del software libre y extender el uso de todo un conjunto de programas y herramientas libres que les ayuden y sirvan.

### Vivir del software libre

Una de las críticas que se le suele hacer al software libre, especialmente por parte de las empresas y desarrolladores de software alude a la forma de hacer negocio con el software libre.

El negocio de las empresas de software propietario consiste precisamente en vender cuantas más copias de sus programas mejor. Si el software puede ser copiado libremente, nadie querrá pagar por él. ¿Cómo es posible que en la actualidad haya empresas que produzcan software libre y ganen dinero?

El secreto consiste, ni más ni menos, en cambiar de modelo de negocio. Ahora ya no se vende software sino servicios. Servicios de puesta en marcha y configuración, asistencia técnica ante incidencias, hospedaje de aplicaciones, formación, consultoría y asesoría, etc.

A modo de ejemplo, cabe proponer diversos escenarios:

#### Escenario 1

Una empresa desarrolla un programa libre de propósito muy específico. Otras empresas consideran útil el programa porque les permite obtener la solución a un problema (por ejemplo, producir un determinado producto) con menos recursos (tiempo y/o dinero, principalmente). Sin embargo, para utilizarlo de manera realmente efectiva es necesario que el usuario tenga grandes conocimientos sobre el problema y sobre el propio programa.

Uno de los servicios que ofrece la empresa productora consiste en formar al personal de las empresas clientes para que sean expertos en la resolución de ese tipo de problemas usando ese programa.

Otra opción posible consiste en ofrecer servicios de *outsourcing* a las empresas clientes, de manera que les resulte económicamente más interesante.

#### Escenario 2

Una empresa desarrolla un programa libre para resolver un determinado tipo de problemas. El resultado producido por el programa tiene una alta complejidad debido a la naturaleza del problema a resolver.

La empresa productora puede ofrecer como servicio la resolución del problema, el análisis de los resultados y la presentación de un informe o asesoría al respecto.

De nuevo las empresas clientes pueden optar por contratar los servicios de la empresa productora si económicamente es más rentable que formar a su propio personal en el uso del programa y en el análisis de los resultados.

En ambos escenarios aparece el software libre como herramienta útil en el servicio prestado pero no como objeto en sí del servicio.

### Documentación libre

El concepto de *documentación libre* es muy similar al de *software libre*. Básicamente hace referencia a aquella documentación que uno puede consultar, distribuir y modificar de acuerdo con alguna licencia de documentación libre.

Hasta hace unos años, la mayor parte de la documentación *libre* estaba sujeta, bien a una licencia de tipo *copyleft*, cuyos términos exactos eran definidos por cada productor, bien a la licencia del software libre a la que acompañaba. Actualmente, la FSF ofrece la licencia *GNU Free Documentation License* (GFDL), aplicable por ejemplo a los manuales y documentación técnica que acompañan al software.

La empresa *Creative Commons* ofrece varias licencias aplicables a trabajos de carácter artístico y creativo (tanto material escrito como audiovisual, educativo, etc. en cualquier formato). En general, recomienda la GFDL para documentación de tipo técnico.

#### Creative Commons:

<http://creativecommons.org>

### Ventajas del software libre

Las ventajas del software libre son muchas, algunas de carácter principalmente práctico y otras de carácter ético. En primer lugar, el hecho de que el código fuente sea accesible permite un crecimiento del programa mucho más rápido que el que sufre el software propietario. Además, los errores y vulnerabilidades existentes se encuentran y arreglan mucho antes.

La explicación a este fenómeno es sencilla: si el código fuente es accesible, mucha gente podrá estudiarlo, adaptarlo a sus necesidades y redistribuir sus modificaciones, lo que redundará directamente en la calidad (y cantidad) de software libre disponible. De forma análoga, cuanta más gente tenga acceso al código fuente, más gente podrá detectar errores y vulnerabilidades de seguridad presentes en el software, y antes se resolverán.

Por otra parte, al disponer del código fuente de un programa se evita la dependencia de los usuarios hacia un fabricante de software concreto. En el caso de que la empresa deje de dar soporte a un determinado programa, éste se queda rápidamente obsoleto. Si los usuarios disponen del código fuente, pueden mantener el programa y corregir errores y vulnerabilidades de seguridad, etc.

En ocasiones algunos usuarios admiten no hacer uso alguno del código fuente de los programas libres. Es posible que un determinado usuario o empresa no pueda continuar un desarrollo determinado, pero otras personas o empresas sí pueden, siempre y cuando dispongan del código fuente y de las libertades antes mencionadas. Es decir, siempre y cuando el código sea libre.

En cualquier caso, un usuario o grupo de usuarios siempre puede pagar para que cualquier programador o empresa de programación de su elección modifique, mejore, adapte o solucione los problemas de un producto de software libre. Como hemos enfatizado anteriormente,

## Software Libre

no se trata de disponer de software de forma gratuita, sino de software libre y abierto, atributos en general mucho más importantes para los usuarios profesionales y las empresas que el mero coste. El acceso libre al código fuente puede verse como un potente argumento de calidad del software.

Existe otra ventaja relacionada con la seguridad de los programas y, sobre todo, de los datos que estos manejan. Esto es muy importante cuando estos datos son especialmente sensibles (datos identificativos, económicos, médicos, etc.).

Cuando un programa es libre y el código fuente está a disposición de cualquier usuario, es realmente difícil incluir en el programa (y que pase inadvertida) funcionalidad maliciosa que atente contra la seguridad de los datos del usuario, por ejemplo para recabar información del sistema sin que el usuario se dé cuenta y enviarla a sitios desconocidos por este.

Esta información puede ser de diverso tipo: qué programas tiene el usuario instalados en su sistema, con qué frecuencia los utiliza, qué medidas de seguridad toma, qué hábitos de uso tiene, qué contactos tiene en su libreta de direcciones de correo electrónico, qué direcciones web visita y con qué frecuencia, qué datos envía (por ejemplo, datos identificativos, números de tarjeta de crédito, etc.) y, en general, cualquier tipo de información que sea accesible.

A otro nivel, encontramos otro tipo de ventajas. En primer lugar, los beneficios *sociales* que produce el software libre son globales. Es cierto que usando software libre, el usuario se beneficia de la comunidad de desarrolladores de software libre, pero también es cierto que el usuario tiene la oportunidad de ayudar a la comunidad, principalmente aportando aquellas modificaciones propias que ha hecho en programas libres.

Por otra parte, es habitual que las licencias de software propietario impongan una serie de restricciones al usuario. Entre ellas suele encontrarse la prohibición de distribuir copias del programa.

En cambio con el software libre no surgen problemas de este tipo, ya que la propia definición de software libre no sólo permite sino que además alienta la redistribución del software.

### Desventajas del software libre

Una de las críticas que se le suele hacer al software libre se refiere al soporte y a la documentación ofrecidos. Las aplicaciones de una cierta envergadura suelen contar con alguna documentación y ciertos recursos, principalmente humanos, dedicados al soporte técnico. En ocasiones, estos servicios son ofrecidos por una empresa creada ad-hoc. Éste es precisamente uno de los modelos de negocio que últimamente algunas empresas están aplicando con éxito.

Sin embargo, en la actualidad la gran mayoría de software libre es desarrollado y mantenido por comunidades y grupos de usuarios organizados, e incluso por desarrolladores individuales. En estos casos, la documentación es limitada y en ocasiones está parcial o totalmente desactualizada, y el soporte técnico está sujeto en muchas ocasiones a

grandes limitaciones (relativas al tiempo y a los recursos de que disponen los desarrolladores).

Esta situación debe ser cuidadosamente valorada por aquellos usuarios o empresas que necesiten una documentación completa y actualizada de la aplicación, un buen soporte técnico rápido y fiable, etc. En su caso, puede considerarse contratar el soporte técnico, bien a la empresa relacionada con la aplicación, si la hay, bien a una tercera empresa que ofrezca tal servicio.

Desde el punto de vista del desarrollador de software libre, existen inconvenientes que conviene conocer. Uno de los problemas más importantes consiste en el volumen de negocio que se puede conseguir con el software libre, reducido en la mayoría de casos. A esto hay que añadir el coste que supone la propia gestión, mantenimiento y desarrollo del producto.

Por otra parte, existe un riesgo que los creadores de una aplicación libre deben conocer. Dado el carácter abierto del código y la posibilidad de que cualquier usuario pueda ofrecer una versión del producto, es posible que el desarrollo de un producto se ramifique en varias versiones, posiblemente mejores o más completas que la original. Una gestión abierta y flexible del proyecto debería bastar para aprovechar las mejoras introducidas en dichas *ramas* e integrarlas en el producto original, de manera que el mayor número posible de usuarios puedan verse beneficiados de las mismas.

### Conclusiones

El software libre es una interesante alternativa al software propietario, que cabe considerar a la hora de instalar un nuevo sistema de información, renovar o mantener uno existente o instalar un sistema doméstico.

A pesar de que el desarrollo y uso de software libre no es nada nuevo, hasta los últimos años no se ha popularizado más en que en determinados entornos. Con los pasos que han dado y están dando gobiernos, comunidades autónomas, ayuntamientos, etc. y la difusión de noticias y espacios correspondientes por parte de los medios de comunicación los usuarios empiezan a interesarse de forma activa por el software libre.

El software libre proporciona innumerables ventajas y *libertades* al usuario y este debe, al menos, conocerlas. Solo así será capaz de decidir de manera objetiva si le interesa o no, y en qué casos, utilizar software libre o software propietario.

Finalmente, las empresas de desarrollo de software han de conocer y comprender también las características y posibilidades del software libre como base de un naciente modelo de negocio basado en la prestación de servicios y en proporcionar soluciones de calidad en un entorno de competencia abierta, más que en la venta de licencias de programas ejecutables.

Autor: Emili Miedes

Más información: [actualidadtic@iti.upv.es](mailto:actualidadtic@iti.upv.es)

## Integridad de datos relacionales

Los datos erróneos o inconsistentes almacenados en bases de datos son causa de errores y fallos, a veces con graves consecuencias económicas para las empresas que los gestionan. A pesar de ello, la comprobación convencional de las restricciones de integridad en bases de datos resulta demasiado costosa. En este artículo presentamos SSQL, una metodología diseñada en el ITI, y actualmente en proceso de desarrollo, para dar soporte a la especificación y comprobación eficiente de restricciones arbitrariamente complejas.

### Motivación

En los últimos números de ActualidadTIC se presentaron los sistemas Hydra y COPLA, del ITI. Ambos permiten salvaguardar datos almacenados electrónicamente frente a posibles fallos de software o hardware. Sin embargo, la mayoría de errores y fallos en la información contenida por las bases de datos no se debe a tales fallos, sino a un desacuerdo entre la semántica pretendida de los datos y los valores que de hecho se almacenan. En otras palabras, los datos almacenados guardan a menudo valores incorrectos de sus atributos. En contexto de bases de datos se habla de *violación de la integridad*, es decir, violación de la consistencia semántica de los datos, cuando uno o más valores son obsoletos, engañosos, o incorrectos de cualquier forma.

En la vida cotidiana, los pequeños errores suelen ser inofensivos y a menudo pueden ser corregidos o ajustados sobre la marcha. Pero cuando hablamos de computadoras incluso los errores más nimios pueden tener consecuencias desastrosas. Una coma ausente, por no hablar de errores tipográficos o relaciones incorrectas entre los datos, puede impedir el funcionamiento correcto de un sistema. En los sistemas de bases de datos, los fallos en la consistencia semántica y la integridad pueden tener consecuencias dramáticas sobre los resultados económicos de empresas que usan tales bases de datos para almacenar información sobre productos y clientes, y para administrar su negocio. Por ejemplo, si los datos de contabilidad de una compañía violan las convenciones para las fórmulas de costos, tales errores pueden resultar en una grave pérdida de beneficios, aunque hayan ocurrido inintencionadamente.



“¡¡Te dije que no debíamos confiar en estos datos!!”

Otros ejemplos de datos fallidos que violan reglas de integridad son, por ejemplo, datos erróneos sobre clientes, direcciones o nombres inexactos en las facturas, precios incorrectos, etc. Tales errores a menudo conllevan tediosas interacciones adicionales con los clientes, costosos retrasos en los pagos, y una menor satisfacción de los clientes.

Los fallos en la integridad de los datos pueden tener graves consecuencias económicas.

En general, la integridad de la información en una base de datos puede ser violada con cada modificación de su contenido, resultando en la necesidad de comprobar todas y cada una de las restricciones de integridad. Tal sobrecoste no es tolerable en una base de datos operacional, que necesita sus recursos y potencia computacional sobre todo para contestar a las preguntas y solucionar cuestiones de los usuarios.

Al problema de los costes prohibitivos que impiden un mantenimiento minucioso de la integridad, se añade la gran dificultad de expresar las restricciones de manera precisa, correcta y completa en el marco formal de un sistema de bases de datos. El grupo de trabajo de bases de datos del ITI ha desarrollado una metodología llamada SSQL para expresar y evaluar restricciones de integridad arbitrariamente complejas para cualquier tipo de aplicación de bases de datos, sin incurrir en costes excesivos sobre el rendimiento.

Este artículo revisa los principios fundamentales de SSQL. Mediante este proyecto el Instituto pretende colaborar con los administradores de bases de datos de PYMEs en la resolución de los problemas relacionados con el mantenimiento de la integridad y consistencia de los datos de que tales administradores son responsables.

### Objetivos de SSQL

El acrónimo SSQL está compuesto por un prefijo S seguido del sufijo SQL. Este significa *Structured Query Language*, es decir, “lenguaje de preguntas estructuradas”, que es el lenguaje estándar de programación declarativa, muy bien establecido en el campo de las bases de datos (BBDD), cubriendo más del 95% del mercado. El prefijo S significa *sound*, adjetivo inglés con connotación lógica estricta, que los diccionarios traducen como “sano”, “sólido”, “seguro”, “severo”, todo ello en el sentido de datos almacenados sin contradicciones, es decir, datos consistentes, correctos y en correspondencia con el sentido buscado por el programador. En resumen: datos con integridad no-violada. Los datos relacionales almacenados en BBDD, siendo *sound*, no llevan a ningún problema de integridad, ninguna conclusión falsa, ni una mala interpretación de su contenido.

## Integridad de datos relacionales

El principal objetivo tecnológico es extender los sistemas gestores de bases de datos relacionales (SGBD) con restricciones de integridad más generales y poderosas que las condiciones de integridad soportadas por BBDD comerciales. Hemos estudiado la viabilidad de las ideas básicas. Estamos desarrollando un prototipo de sistema automático y eficaz para especificar y hacer cumplir tales restricciones a lo largo de la evolución de la BBDD. En general, las restricciones de integridad se definen como la clase de condiciones especificadas en la sintaxis declarativa más general de SQL. Actualmente, ningún SGBD comercial provee el soporte para restricciones de integridad tan generales. La clase de condiciones declarativas de integridad realizada en SSQL es mucho más expresiva y poderosa que la que ofrecen los sistemas en el mercado, permitiendo capturar el sentido del mundo real asociado con los datos almacenados.

Habitualmente, la especificación del esquema de una BD incluye tipos sencillos y rudimentarios de restricciones de integridad. Tales restricciones son condiciones declaradas en un lenguaje formal para capturar el sentido correcto de los datos y para evitar que alguna modificación de estos pudiera violar su corrección y exactitud. La finalidad de la metodología SSQL es ampliar la expresividad y el alcance de la especificación de restricciones de integridad en BBDD relacionales, permitiendo una complejidad arbitraria al expresar cualquier condición, sin comprometer la eficacia de hacerla cumplir automáticamente.

Las restricciones de integridad permiten la declaración y comprobación de condiciones para expresar la consistencia, corrección y exactitud de datos almacenados. Todos los SGBD comerciales ofrecen la posibilidad de especificar condiciones sencillas de integridad. Pero existe muy poco soporte declarativo para las restricciones que se relacionan con el significado general de las aplicaciones y con las partes del mundo codificadas en las BBDD. El objetivo de SSQL es realizar un soporte eficaz de restricciones de integridad que sean de una generalidad y universalidad sin precedente en el mercado, sobrepasando todo lo que se ofrece actualmente por los vendedores más importantes de SGBD. Y asegurando que toda esta funcionalidad adicional se cumpla sin costes exorbitantes.

### Beneficiarios de SSQL

Hay tres clases diferentes de beneficiarios inmediatos de SSQL. En primer lugar los diseñadores de BBDD pueden beneficiarse de la mayor facilidad y claridad de la especificación declarativa de restricciones de integridad arbitrariamente complejas. En segundo lugar, los usuarios de BBDD (es decir, gerentes, administradores o colaboradores en proyectos, o bien empleados o clientes con sus aplicaciones) pueden disfrutar las ventajas de un nivel superior de consistencia y corrección de sus datos. En tercer lugar, los vendedores de SGBD o de sistemas que se apoyan en ellos, puesto que esperamos les será fácil incorporar la tecnología SSQL en sus productos, con la finalidad de mejorar su funcionalidad.

En general es un fenómeno bien conocido que una colección de datos almacenados puede perder su integridad a lo largo de su ciclo de vida. Típicamente, la

información que ha sido establecida sólidamente una vez en una BD, tiende a ser sobrescrita, incomprensiblemente corregida o modificada o a quedar obsoleta con el curso del tiempo. Las restricciones de integridad sirven para prevenir tal deterioro de la consistencia y corrección de las informaciones almacenadas.

Normalmente, los desarrolladores de aplicaciones de BBDD están forzados a escribir código adicional para hacer cumplir la integridad de los datos almacenados. Tal código propietario es propenso a errores, tiende a producir efectos secundarios indeseados, y a medida que se acumulan las modificaciones de datos, fácilmente puede comportarse de forma incontrolada. Además, el uso ad-hoc de disparadores procesales (desgraciadamente recomendado por los vendedores de BBDD como medio de apoyar la integridad) tiene esencialmente las mismas desventajas, y puede producir reacciones en cadena incontrolables. En comparación, las restricciones de integridad son declarativas, tienen características demostrables de comportamiento correcto y pueden ser mantenidas muy fácilmente, debido a su independencia de los programas de uso. Así, puede esperarse que la utilización de la tecnología SSQL dé lugar al aumento de la calidad de los datos y de la productividad de los usos de bases de datos.

Un aspecto importante de SSQL es su usabilidad con los productos de todos los vendedores importantes de BBDD conformes al estándar establecido de SQL. Por lo tanto, SSQL resultará interesante y útil tanto para los vendedores de bases de datos como para sus clientes. Además, intentamos desarrollar nuestro sistema para ser utilizado no solo sobre productos comerciales, sino también, en una versión futura, para ser integrado en el código del principal producto *open source* de BD, MySQL. Puesto que MySQL es un producto de código libre, su ampliación con la integridad semántica reforzará su posición primordial en el movimiento de código abierto.

### Estado actual

La integridad de datos relacionales puede ser expresada de manera declarativa o procesal. Las soluciones que existen en el campo científico son de tipo declarativo y muy avanzado, pero nunca han salido del campo teórico. Por otra parte, lo que está realizado en los SGBD comerciales con respecto a la integridad es bastante pobre en términos de declaratividad. Por ello las restricciones de integridad avanzadas han de ser especificadas de manera procesal. Esto las hace más propensas a errores y más difíciles de mantener, pues a menudo están completamente oscurecidas dentro del código de la aplicación. Una contribución esencial de SSQL es la de adaptar y trasladar lo que ha sido concebido de forma teórica y científica al mundo práctico y comercial.

Actualmente, la mayoría de los SGBD comerciales solamente ofrecen soporte para los siguientes tipos declarativos de restricciones de integridad:

- Restricciones de dominio (por ejemplo tipos de datos básicos, opcionalmente de rango restringido, como números, cadenas de caracteres, etc., incluyendo opciones que permitan valores nulos).

## Integridad de datos relacionales

- Restricciones de unicidad (por ejemplo las claves primarias, que garantizan la inconfundibilidad de los objetos almacenados, o los índices únicos, que facilitan la búsqueda sistemática de datos).

- Restricciones de clave ajena (para asegurar la correlación correcta entre diferentes tablas, p.e., que para cada asignación de una persona a un proyecto en una tabla que almacene los equipos de personal, haga falta que cada una de estas personas esté también registrada en la tabla de empleados).

Las restricciones de dominio y unicidad son muy simples y básicas. Las primeras están ligadas a una sola columna de una tabla, mientras que las segundas pueden aplicarse a combinaciones de columnas, pero dentro de una misma tabla. Las restricciones de clave ajena relacionan columnas que pueden estar en dos tablas distintas. Adicionalmente tales columnas deben satisfacer una restricción de unicidad. La potencia expresiva de las claves ajenas es bastante limitada. Por ejemplo no es posible expresar, con ninguna combinación de las tres construcciones citadas, que en una tabla de registro de ciudadanos solteros y matrimonios en una comunidad, cada persona esté casada como máximo con otra persona y que la edad de cada persona en el momento de casarse sea al menos 18, a menos que haya un consentimiento explícito por parte de los padres legales. En cambio la potencia expresiva de las restricciones de integridad soportadas por SSQL no está limitada en el número de tablas involucradas ni por ningún otro requisito estructural.

En los últimos años, también la expresividad declarativa de SQL se acercó a la completitud de expresar y evaluar restricciones de integridad arbitrariamente generales, en virtud de la introducción de las subconsultas, unas construcciones de calificación (EXISTS) y negación (NOT), y tablas especiales llamadas vistas (cuyas definiciones se apoyan en otras tablas más básicas). Sin embargo, todos los manuales de SGBD comerciales todavía recomiendan la implantación de restricciones de integridad complejas mediante procedimientos, porque el uso de medidas declarativas resulta a menudo demasiado costoso, hasta al punto de su intolerabilidad. Es precisamente este problema de falta de eficiencia lo que SSQL trata de resolver.

Las restricciones de integridad que van más allá del nivel introductorio de SQL, normalmente implican *joins* (productos de combinaciones) enormes de varias tablas, consultas de tablas enteras, subconsultas y negaciones anidadas, y similares. Así, los costes de su evaluación llegan a ser fatalmente caros. Las aplicaciones típicas de SGBD (p.e., sistemas de la reserva de vuelos) y también los procesos de tiempo crítico (p.e., el *data warehousing* para extraer, borrar, transformar, homogeneizar y cargar datos empresariales), no pueden soportar caras comprobaciones de integridad. Por eso, los vendedores de SGBD desaconsejan la utilización de condiciones declarativas de consulta. En lugar de eso, se debe recurrir a las construcciones más opacas basadas en procedimientos.

Hay tres maneras de expresar restricciones de integridad mediante procedimientos: como *triggers* SQL disparados ante actualizaciones predefinidas, como procedimientos almacenados asociados a determinadas transacciones, o incluyéndolas directamente en el código de la aplicación.

Cada una de estas tres opciones compromete gravemente el ideal de declaratividad. La utilización de procedimientos implica los peligros ya mencionados. Además, cada implantación mediante procedimientos de las restricciones de integridad tiene el inconveniente adicional de frustrar su posible simplificación, que aceleraría su evaluación enormemente. Por el contrario, SSQL aprovecha esta posibilidad de simplificar la comprobación.

### Ejemplos de SSQL

Fijemos una BD conteniendo entradas de trabajadores o gerentes en la jerarquía de empleados de una gran empresa. Una condición típica de integridad semántica que no puede ser expresada y mantenida eficientemente con las medidas de integridad proporcionadas por los SGBD comerciales sería la de que ningún obrero sea gerente. Otra condición aún más complicada podría exigir para cualquier departamento la existencia de un individuo que fuera el superior de cada empleado del propio departamento, excepto este mismo individuo.

Dado que cada actualización puede resultar en una violación de la integridad de los datos, se sigue la necesidad de comprobar las restricciones de integridad casi continuamente en BBDD con elevada frecuencia de transacciones. Tal sobrecoste en general no es tolerable. Por ejemplo, la comprobación de una restricción de integridad que postule que la altura de la jerarquía de empleados no sobrepase tres niveles, necesitaría, en una BD con entradas para cada empleado y cada uno de sus superiores directos, la travesía de un espacio de búsqueda enorme (1.000.000.000.000 de pares de datos relacionales cuando el número de empleados sea 10.000). En una BD muy dinámica con miles de datos la comprobación de restricciones complejas de integridad sería pues prácticamente imposible.

Consideremos ahora como ejemplo concreto una BD para la administración del personal en un departamento de una empresa. Supongamos una tabla con el nombre **superior**, capturando qué empleado es el superior directo de otro empleado y asumamos por simplicidad que la tabla **superior** tiene solamente dos columnas, cada una con una gama de los valores que permiten la identificación única de empleados. Una entrada **superior(A,B)** significaría que el individuo **A** es un superior directo de **B**. Supongamos que en una ronda de reestructuración ha sido impuesta una regla de negocio según la cual la jerarquía directiva deba ser aplanada, de forma que solo tres niveles jerárquicos de gerencia están permitidos. Esto significa que no debe existir ninguna cadena de entradas en la tabla **superior** de la forma:

**superior(X1,X2)–superior(X2,X3)–superior(X3,X4)**,

donde X1, X2, X3, X4 simbolizan individuos de manera formal. O, de forma más exacta:

No existen X1, X2, X3, X4 tal que **superior(X1,X2)** y **superior(X2,X3)** y **superior(X3,X4)**.

La traducción de especificaciones de restricción de integridad en lenguaje natural, aún más conveniente para el usuario, en una forma equivalente de SQL está fuera del alcance de SSQL. Sin embargo, la continuación del proyecto SSQL incluirá tal objetivo, teniendo en mente

## Integridad de datos relacionales

que un grupo muy fuerte del ITI se ocupa de lenguajes naturales.

En los SGBD del mercado, una regla como la propuesta ha de ser llevada a cabo por un procedimiento uso-dependiente, que comprobaría si hay alguna cadena que viole el requisito especificado en la regla. De ser así, publicaría un mensaje de advertencia. Tales procedimientos son muy propensos a errores. Por el contrario, SSQL permite una representación declarativa de la regla de negocio en forma de restricción de integridad, como la siguiente condición:

```
NOT EXISTS (SELECT * FROM superior s1,
            superior s2, superior s3 WHERE superior1.#2 =
            superior2.#1 AND superior2.#2 = superior3.#1)
```

Según la sintaxis estándar SQL-2, los alias superior1, superior2, superior3 sirven para distinguir los tres diferentes usos de la tabla **superior**, y los dos atributos de la tabla están referidos por .#1 y .#2, respectivamente. La evaluación de condiciones como la anterior puede ser extremadamente costosa. Si hay 10.000 empleados y 30.000 superiores directos (para cada empleado puede fácilmente haber tres superiores diferentes, p.e., un técnico, un administrativo y un comercial), nuestra condición SQL proporciona un espacio de búsqueda de 27 billones de tripletes de pares de empleados, cada uno de los cuales podría en principio ser un candidato a violar la integridad. Este enorme espacio tendría que ser recorrido para cada cambio en la BD. Con SSQL, el espacio se reduce en 4 potencias de 10 a 2.700 millones. Otra reducción del mismo orden se logra con un índice apropiado impuesto sobre la tabla **superior**.

### Loa seis pasos de SSQL

Vamos a ilustrar la funcionalidad innovadora de SSQL, que logra una reducción dramática de costes, mediante un ejemplo concreto y una sucesión de seis pasos de la metodología.

Consideremos una BD con tablas con atributos de nombre (**name**) para trabajadores (**worker**) y gerentes (**manager**). Supongamos la restricción de que ningún trabajador sea gerente. Eso se expresa en SQL como

```
NOT EXISTS (SELECT * FROM worker, manager
            WHERE worker.name = manager.name)
```

Un programador de SQL podría decir que dicha restricción será mucho más fácilmente comprobable con un disparador relacionado a las inserciones en la tabla **worker**. Sin embargo, es fácil pasar por alto que la restricción de integridad arriba indicada ya exige otro disparador, para las inserciones en **manager**. Y, aparte de que los disparadores codificados a mano sean susceptibles de fallos, pueden traer también efectos imprevisibles de interacciones mutuas.

Supongamos que **INSERT worker(Fred, sales)** sea una actualización a llevar a cabo. La metodología implica la ejecución de los siguientes pasos:

#### 1. Generar diferencia entre estado previo y posterior

La actualización explícita **INSERT worker(Fred, sales)** puede tener actualizaciones derivadas implícitas sobre

vistas de la BD cuyas definiciones involucren a la tabla **worker**. El conjunto de actualizaciones implícitas y explícitas constituye la diferencia entre el estado previo de la base de datos y el nuevo. Cada acción en esta diferencia puede violar la integridad.

#### 2. Evitar actualizaciones vanas

Si Fred era ya un trabajador (p.e., en otro departamento) antes de la inserción, no es necesario comprobar de nuevo que no es un gerente.

#### 3. Identificar las restricciones relevantes

A menos que se haya aplicado 2, la restricción de que ningún trabajador pueda ser gerente es claramente relevante por las inserciones en la tabla **worker** (pero nunca por un borrado), y debe ser comprobada.

#### 4. Especializar las restricciones relevantes

Para la inserción dada, la condición SQL

```
EXISTS (SELECT * FROM worker, manager WHERE
        worker.name = manager.name)
```

podría especializarse a una forma mucho menos cara:

```
EXISTS (SELECT * FROM worker, manager
        WHERE worker.name = 'Fred' AND worker.name =
        manager.name)
```

#### 5. Optimizar las restricciones especializadas

Claramente, la condición especializada en 4 puede ser optimizada por la siguiente sentencia:

```
EXISTS (SELECT * FROM manager WHERE name =
        'Fred')
```

#### 6. Evaluar restricciones

Finalmente, la evaluación de la consulta resultante sobre si Fred es un gerente será sencilla. La búsqueda de una sola fila en una sola tabla es, por supuesto, mucho menos costosa que tener que evaluar por completo todas las restricciones de integridad.

### Conclusión

El ejemplo comentado es muy sencillo. Sin embargo se pueden obtener las mismas simplificaciones para restricciones arbitrariamente complejas.

Actualmente en el ITI se está desarrollando una metodología sistemática que traduce automáticamente los resultados de esta secuencia de pasos en *triggers* seguros que resultan aún más eficientes.

Autor: Hendrik Decker  
Más información: [sidi@iti.upv.es](mailto:sidi@iti.upv.es)

## Primera Jornada de Testeo (JTS2004)

El pasado mes de Marzo, expertos internacionales se reunieron en Valencia para fomentar la calidad y el testeo del software en la primera Jornada de Testeo de Software. Alrededor de 80 participantes de toda España participaron en el evento. Los participantes pertenecían a empresas relevantes de diferentes sectores, no solo los relacionados con el sector informático, como proveedores de soluciones informáticas, sino también de los sectores de banca y grandes almacenes.

En estas páginas resumimos los contenidos tratados durante este encuentro, que se celebró en el campus de la Universidad Politécnica de Valencia (UPV).

### Introducción

La Primera Jornada de Testeo de Software, organizada por el grupo SQuaC (Software Quality and Correctness) del Instituto Tecnológico de Informática y patrocinado por las empresas SOGETI (<http://www.sogeti.com>) y CompuWare (<http://www.compuware.es>) pretende concienciar a las empresas españolas del sector de las TICs acerca de la importancia de testear software en el mercado actual mediante la presentación de los procesos básicos que las empresas deben realizar para poder garantizar la calidad de sus productos. Entre otros, el encuentro abordó los fundamentos y estándares del testeo, las principales herramientas de testeo de aceptación automatizado que actualmente existen en el mercado y las diferentes técnicas, modelos y metodologías que se pueden implantar. Los asuntos a ser tratados durante la Jornada fueron preparados por el ITI con antelación para asegurar que se ajustaran a las necesidades de las empresas españolas desarrolladoras de software y cubrieran los más novedosos aspectos sobre testeo de software.

### Desarrollo de las JTS2004

La Jornada empezó con una introducción al testeo de software que explicaba varias definiciones del testeo, la historia del mismo desde 1956 hasta hoy, la relación de los procesos del testeo con el ciclo de vida del desarrollo del software, y las causas de la mayoría de errores que se encuentran en el software.

Después de la introducción, Tim Koomen, ganador del *European Testing Excellence Award* de 2003 y testeador profesional por Sogeti, pronunciaba su ponencia sobre el modelo TPI® para la mejora de procesos de testeo (Test Process Improvement) y el enfoque TMap® para la gestión del testeo estructurado. Koomen indicaba cómo TPI® está basado en las mejores prácticas de la industria relativas a la mejora del proceso del testeo, y explicaba

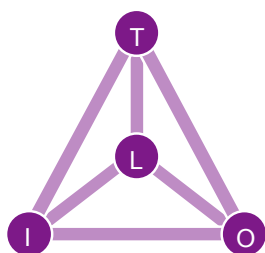


Figura 1: Los pilares de TMAP

(Software Testing - A Guide to the TMap Approach. E. van Veenendaal, M. Pol, R. Teunissen, Addison-Wesley, 2002)

**La JTS2004 demostró el creciente interés del testeo de software para las empresas.**

sus guías prácticas para evaluar el nivel de madurez del testeo de una organización así como los pasos para mejorar el proceso. Su ponencia sobre TMap® proporcionaba respuestas a qué, cuándo, cómo, dónde y quién en cuestión del testeo. Para estructurar la empresa y ejecutar los procesos, TMap® se basa en cuatro pilares: (L) un modelo del ciclo de vida (Life cycle) para las actividades del testeo relacionado con los procesos del desarrollo de software; (O) una Organización sólida; (I) una Infraestructura adecuada; y (T) Técnicas para las diversas actividades de prueba. Koomen indicaba que durante los últimos años TMap® se ha transformado en un estándar en los Países Bajos, Bélgica, Luxemburgo y Alemania y está siendo utilizado por más de trescientas empresas.



Algunos momentos de las Jornadas

Manuel Caballero de la Torre, Jefe de Área de Informática y responsable de comunicaciones de la Subdirección de Nuevas Tecnologías del Ministerio de Justicia, explicó en detalle las características de la metodología Métrica. METRICA es una Metodología de Planificación, Desarrollo y Mantenimiento de sistemas de información del Consejo Superior de Informática del Ministerio de Administraciones Públicas. Su objetivo es ofrecer a las organizaciones un instrumento útil para la sistematización de las actividades

## Primera Jornada de Testeo (JTS2004)

que dan soporte al ciclo de vida del software. Básicamente divide el Proceso de Desarrollo de un Sistema de Información en 5 subprocesos: Estudio de viabilidad del sistema (EVS), Análisis del sistema de información (ASI), Diseño del sistema de información (DSI), Construcción del sistema de información (CSI) e Implantación y Aceptación del sistema (IAS). Dentro del DSI se encuentra la Especificación Técnica del Plan de Pruebas, donde debe indicarse qué pruebas realizar y en qué momento, considerando los siguientes tipos de pruebas: Unitarias, de Integración, del Sistema, Implantación, de Aceptación y de Regresión. Por tanto, nos proporciona un esquema normalizado para la realización de las actividades de testeo de software y gestión de calidad del software.

Stuart Reid, profesor de ingeniería de software en la Universidad de Cranfield en el Reino Unido, presentó el marco actual de estándares relacionados con el testeo de software (Fig. 2). Stuart Reid mostró como las actividades de testeo de software y la documentación de un proyecto típico están relacionadas con los estándares. Finalmente, analizó el marco actual en mayor profundidad de modo que fallos serios quedaron en evidencia. Grandes lagunas, áreas de duplicidad y diversidad respecto a terminología y requisitos fueron identificados. El objetivo de esta presentación fue animar a los testeadores a usar y beneficiarse de los mejores estándares de testeo de software que existen en la actualidad.

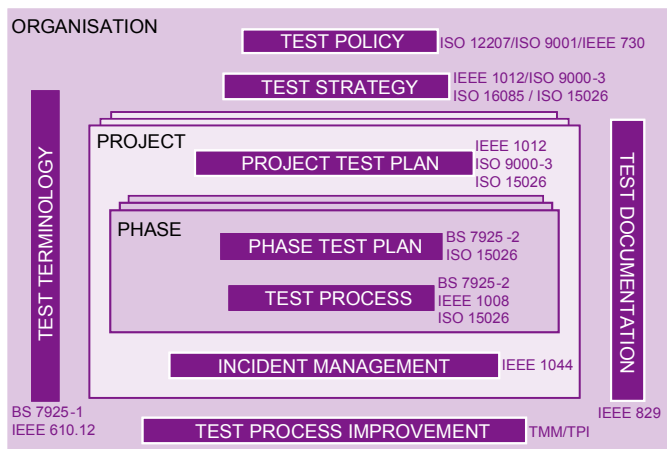


Figura 2: Marco actual de los estándares más importantes existentes sobre testeo de software

Roberta Biffi, directora de Soporte a soluciones de Compuware, expuso CARS, la solución de Compuware para la fiabilidad de aplicaciones software. La mayor parte de las organizaciones solo prueban sus aplicaciones inmediatamente antes de su implantación, cuando a menudo resulta ya demasiado tarde y costoso resolver problemas. Los métodos tradicionales de entrega de aplicaciones suelen resultar ineficaces, ya que carecen de procesos consistentes y de indicadores de calidad sobre los que basar las decisiones de implantación. Cuanto mayor es el grado de influencia de una aplicación, mayor importancia adquiere la calidad. R. Biffi introdujo CARS como la solución que comprende la metodología, tecnología y experiencia necesarias para garantizar la calidad y fiabilidad de las aplicaciones, permitiendo predecir fallos y gestionar el buen funcionamiento de estas

una vez están en producción. En su presentación daba respuestas a las siguientes preguntas: ¿Cómo “pilotar” la calidad y fiabilidad de las aplicaciones utilizando CARS? ¿Cómo puede CARS ayudar a decidir si una aplicación está lista para el despliegue? ¿Qué indicadores son necesarios para tomar la decisión final? ¿Cómo se puede garantizar el cumplimiento de los requisitos críticos?

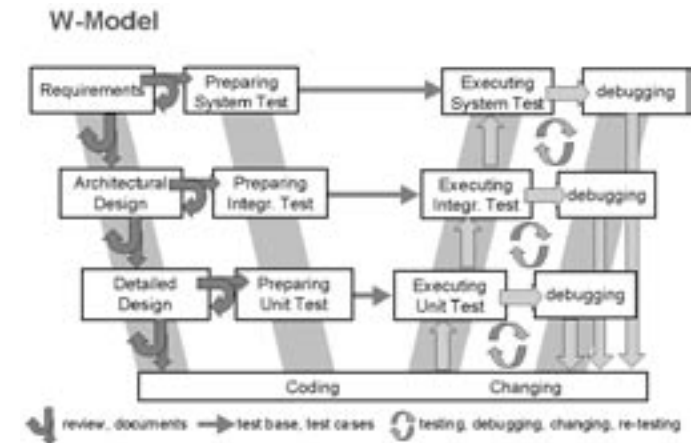


Figura 3: El Modelo W de Andreas Spillner  
(The W Model Strengthening the Bond Between Development and Test. A. Spillner, SQE STAR Conference 2002)

La última ponencia fue de Andreas Spillner, profesor en el Hochschule Bremen, donde es responsable de las áreas de ingeniería de software y garantía de calidad. En esta exposición trató su modelo W (Fig. 3), un modelo para ejecutar los procesos de testeo paralelo a los procesos de desarrollo de software. En desarrollo de software, el 30–40% de la actividad está relacionada con el testeo. Por este motivo, es esencial lanzar las actividades de testeo al principio del proyecto en lugar de hacerlo una vez terminada la codificación. El modelo W clarifica la prioridad de las tareas y la dependencia entre las actividades de desarrollo y las de testeo. Aun siendo simple, el modelo hace patente la importancia de ordenar las actividades individuales de testeo. Asimismo, evidencia que testeo no es equivalente a eliminación de errores.

### Conclusión

Con esta intervención se dieron por concluidas las Jornadas de Testeo de Software 2004. Queremos desde aquí agradecer la participación y el interés de todos los asistentes a las mismas.

Gracias a la gran aceptación de las jornadas y al espíritu del ITI de innovación y acercamiento de las últimas tendencias tecnológicas a las empresas esperamos la celebración, el próximo año, de unas Segundas Jornadas de Testeo (JTS2005) que continúen la labor de acercar el conocimiento sobre testeo de software a las empresas.

Autoras: Tanja Vos, Marga Nácher y Ester Palacios

Más información: [squac@iti.upv.es](mailto:squac@iti.upv.es)

## Noticias Breves

### Participación del ITI en DOMOGAR



Del 5 al 8 de Mayo de 2004 se celebró **DOMOGAR**, la primera Feria Europea de la Domótica y del Hogar Digital en las instalaciones de Feria Valencia. En ella concurren las principales empresas valencianas y españolas relacionadas

con las instalaciones domóticas y las principales asociaciones del sector, para mostrar los últimos avances tecnológicos en sistemas de control, seguridad, comunicaciones y confort.

En un stand conjunto con otros Institutos Tecnológicos, el ITI preparó una demostración de su sistema de reconocimiento facial que causó gran interés entre los asistentes a la feria.

### Innovación en las PYMES

El pasado miércoles 2 de junio, se presentaron a las empresas y a los representantes de las diversas universidades y entidades empresariales de la Comunidad Valenciana las distintas oportunidades desarrolladas en materia de innovación tecnológica en el ámbito internacional.

La jornada estaba orientada a presentar una serie de instrumentos y medidas de apoyo al desarrollo tecnológico y la innovación puestos a disposición de las PYME en la Comunidad Valenciana, a nivel regional, nacional e internacional. Se analizaron las novedades en materia de desgravaciones fiscales por actividades de I+D+i en el impuesto de sociedades y los programas regionales de apoyo al desarrollo tecnológico y la innovación. Asimismo se presentaron las oportunidades que a través del IRC CENEMES se ofrece a las empresas en materia de transferencia transnacional de tecnología.

Cabe destacar la participación de IDI EIKON (asociada al ITI) y el Grupo ETRA (que participa con el ITI en un proyecto Europeo) como "casos de éxito" en el marco de los Programas Europeos de Cooperación Internacional.

### Red CENEMES

Desde el mes de abril del presente año, el ITI forma parte de la **RED CENEMES** (Centro de Enlace del Mediterráneo Español) que es uno de los miembros de la Red Europea de Centros de Enlace para la Innovación. Este último es un proyecto financiado por el Programa INNOVATION de la **Unión Europea**, que tiene el objetivo de promover y facilitar la transferencia de tecnología entre las empresas, universidades y centros de investigación europeos.

Aquellas empresas asociadas que tengan deseos de introducir una oferta o demanda tecnológica en la Red CENEMES, o iniciar un proceso de cooperación con alguna empresa cuya oferta o demanda haya sido insertada en la Red, solo tienen que ponerse en contacto con el ITI (Carolina Quintá).

### Participación del ITI en LLIUREX

Durante el pasado mes de Mayo se celebró el **I congreso de Software Libre, Comunidad Valenciana, LLIUREX**, en el Museo Principe Felipe. El evento congregó a algunos de los máximos representantes del movimiento del software libre mundial como Richard Stallman (fundador del GNU Project), o Jon *Maddog* Hall (presidente de Linux Internacional), y sirvió como punto de discusión sobre el panorama actual y futuro del software libre tanto a nivel nacional como europeo y mundial.



Diversas empresas asociadas del ITI, así como el propio Instituto, participaron en el congreso. El ITI presentó talleres divulgativos (traducción asistida, reconocimiento biométrico facial, o difusión digital de audio y video), así como una conferencia sobre las experiencias del ITI en la implantación de aplicaciones basadas en software libre.

## CURSOS DE FORMACIÓN DEL ITI

Uno de los objetivos fundamentales del ITI consiste en organizar un buen servicio de formación que permita preparar a las empresas en el uso de las Tecnologías de la Información y que tenga localizados a los mejores profesores de las distintas materias, provenientes en su mayoría de las Universidades Valencianas.

En virtud de ello, el Instituto realiza cursos estándar y a medida, los cuales se pueden celebrar en cualquier punto de la geografía española, dentro de la propia empresa o, si es más conveniente, en las instalaciones del ITI.

Los cursos a medida son acciones de formación que, partiendo de módulos disponibles, se pueden desarrollar adaptándolos a las condiciones particulares de cada empresa, con el objeto de mejorar la

tecnología informática de la empresa.

Además, se realizan cursos subvencionados al 85% por el Fondo Social Europeo y el IMPIVA, para empresas industriales con sede social o establecimiento de producción industrial en la Comunidad Valenciana. Durante los próximos meses de julio y septiembre se dictarán cuatro cursos de formación continua: dos basados en Lenguaje de Programación Java, uno sobre Seguridad en los Sistemas de Información Empresariales y finalmente uno sobre Administración de Sistemas Linux.

Se invita a participar a todos aquellos que tengan experiencia demostrable en este campo y que sean autónomos o trabajadores en activo. Cada curso tiene una duración estimada de 30 horas.

### Más información:

<http://www.iti.upv.es/>  
[otri@iti.upv.es](mailto:otri@iti.upv.es)

## Oferta y Demanda Tecnológica

*Las siguientes Demandas y Ofertas Tecnológicas proceden de empresas innovadoras, Institutos Tecnológicos y Universidades de toda Europa y son promocionadas por la Red de Centros de Enlace para la Innovación (Red IRC).*

*Interesados, contactar con Carolina Quintá: [otri@iti.upv.es](mailto:otri@iti.upv.es)*



### OFERTA DE TECNOLOGÍA

**Ref. 01060402 - Sistema de apoyo en la toma de decisiones basado en modelos de procesos dinámicos y optimización dinámica.** Un consorcio de centros de investigación y empresas europeas ha desarrollado y probado un sistema de apoyo en la toma de decisiones basado en modelos de procesos dinámicos y en optimización dinámica. La principal novedad de este sistema es que se puede iniciar la optimización dinámica como una herramienta diaria en las fábricas. De esta forma los operarios pueden tomar decisiones complejas de forma eficaz y ser más flexibles a las necesidades de los clientes. El consorcio busca socios para comercializar el sistema bajo acuerdos de licencia.

**Ref. 01060403 - Sistema corporativo de gestión de procesos.** Una empresa asturiana ha desarrollado un software de gestión de procesos para empresas. Este software es de código abierto y ofrece a través de la web un conjunto de herramientas que se adaptan fácilmente a las necesidades de cualquier organismo: gestión de documentos, incidencias, contactos, eventos, tareas, reuniones, emails, etc. La información se guarda en una sitio web común al que puede acceder toda la plantilla. Este software agiliza el flujo de información, elimina las barreras entre los departamentos de una empresa y mejora la toma de decisiones y el servicio a los clientes. La empresa desea alcanzar acuerdos de "joint venture" o comerciales con asistencia técnica.

**Ref. 01060406 - Software de gestión de fondos y control de proyectos subvencionados.** Una PYME británica ha desarrollado un software de control de proyectos y gestión de fondos para hacer un seguimiento de proyectos subvencionados. El software, basado en web, puede utilizarse de forma independiente o integrarse con otras aplicaciones y está especialmente indicado para organismos involucrados en la financiación o administración de proyectos: gobiernos locales o centrales, colegios, universidades, agencias de desarrollo u organismos de investigación. La empresa busca socios para alcanzar acuerdos de licencia o cooperación con asistencia técnica.

**Ref. 31050401 - Instalador de software multiplataforma para Linux, Windows y Unix.** Una empresa andaluza desarrolla software multiplataforma y herramientas de gestión para sistemas operativos como Linux, Windows, Solaris, Unix, etc. La empresa ha desarrollado un software que permite reducir los costes de los servicios de apoyo de las empresas. Este software permite realizar de forma automática los complejos pasos de instalación que se realizan de forma manual y que a menudo

producen errores. La empresa busca socios para alcanzar acuerdos comerciales con asistencia técnica.

**Ref. 31050406 - Análisis de patrones de tejidos.** Una PYME polaca ha desarrollado un aplicación para Windows de 32 bits especialmente diseñada para realizar análisis exhaustivos de patrones de tejidos. Estos patrones se obtienen mediante diferentes técnicas ópticas. La aplicación incluye rutinas especializadas para la medición automática del alargamiento/desplazamiento, formas en 3D y rugosidad. La empresa busca socios industriales con perfiles complementarios (dedicados a CAE, multimedia, medicina) y socios comerciales.

**Ref. 28050408 - Software para agentes de seguros de vehículos.** Una PYME polaca ha desarrollado un software para agentes de seguros de vehículos. Este software permite gestionar diferentes seguros y está especialmente indicado para los clientes que aseguran sus vehículos con diferentes empresas, en las que los algoritmos para calcular las primas son diferentes. El software permite realizar modificaciones de forma más rápida que con los sistemas convencionales. La empresa busca socios para continuar con el desarrollo y comercializar el software.

**Ref. 28050402 - Software de visualización.** Una empresa griega con una larga tradición en arquitectura e ingeniería de construcción ha desarrollado un software de visualización estereoscópica, fotorrealística, en 4D y en tiempo real de cualquier modelo sólido producido por los programas de diseño más conocidos (AutoCAD, 3d-Studio, etc.). Los resultados pueden visualizarse en pantallas de ordenador, pantallas estereoscópicas o proyectores gemelos estereoscópicos. El sistema está desarrollado en C++ y OpenGL y utiliza las últimas tecnologías de modelado sólido, renderizado, animación y estereoscopia. La empresa ofrece asistencia técnica y desea alcanzar acuerdos de comercialización.

**Ref. 28050403 - Servidor para comunicaciones con teléfonos móviles, PDAs y agendas electrónicas.** Una PYME francesa especializada en software de comunicaciones ha desarrollado un servidor que facilita el acceso desde las PDAs o PCs a todas las aplicaciones corporativas de una empresa. Este servidor es totalmente compatible con la infraestructura existente de la empresa. La arquitectura del servidor permite realizar conexiones directas utilizando los diferentes tipos de tecnologías de telecomunicaciones: GSM, Wi-Fi, ADSL, etc. Todas las comunicaciones están codificadas para garantizar la integridad y no vulnerabilidad de los datos. La compañía francesa busca un distribuidor de tecnologías de comunicaciones o integradores de software para promocionar y distribuir el servidor.

### DEMANDA DE TECNOLOGÍA

**Ref. 02060401 - Módulos complementarios para software de gestión de la producción.** Una empresa belga ha desarrollado un software de gestión de la producción para diseñar, gestionar, planificar y controlar el desarrollo completo de un producto, su ciclo de vida y el proceso de realización. Este software permite a los gestores de proyectos y a los ingenieros de producción controlar las etapas de un proyecto, sus planificación, costes y recursos disponibles. El software está basado en Java y ofrece una gran capacidad de comunicación y compatibilidad con otras plataformas. La empresa busca ahora módulos complementarios (ERP, financiación, ventas, cadena de suministro, etc.) para mejorar su producto mediante el establecimiento de acuerdos de licencia y cooperación técnica.

**Ref. 02060402 - Socios integradores para planificación de recursos materiales.** Una empresa belga dedicada al desarrollo de software de gestión de la producción desea aumentar su cartera de productos y busca socios integradores con experiencia en la implementación de sistemas para entornos industriales. La estrategia de la empresa es ofrecer un concepto innovador de definición de datos y control de la producción que sea intuitivo y sencillo de utilizar. Los socios buscados deben tener un amplio conocimiento en Java y en tecnologías de bases de datos. La empresa desea alcanzar acuerdos de licencia y cooperación técnica.

**Ref. 01060401 - Sistema web basado en herramientas de código abierto para el intercambio de documentos.** Una empresa austríaca con experiencia en el desarrollo de software de gestión de procesos, desarrollo de sistemas portátiles y eCRM busca un sistema web basado en herramientas de código abierto. Los usuarios internos y externos de este sistema deben tener acceso controlado mediante derechos específicos. Este sistema se empleará como herramienta de apoyo a los equipos de gestión de proyectos y para intercambiar documentos y experiencias. La empresa busca socios que dispongan de la tecnología o que deseen colaborar en su desarrollo.

**Ref. 24050401 - Sistema de gestión de flotas de vehículos.** PYME israelí busca socio que disponga de tecnología para generar rutas de flotas de vehículos. Estos vehículos se encargan de la recogida de residuos de contenedores. El sistema incorporará un sensor y comunicación inalámbrica para determinar si los contenedores están llenos. La información se transmitirá a un centro de control en que el software determinará la ruta óptima. La empresa, que ya tiene desarrollados el sensor y las comunicaciones, desea alcanzar acuerdos de "joint venture" o cooperación técnica.

## Ayudas y Subvenciones

### Ámbito Autonómico

#### PROGRAMA DE APOYO A EMPRESAS INDUSTRIALES 2004

##### PROGRAMA DE DESARROLLO TECNOLÓGICO

Organismo Gestor: IMPIVA

**Beneficiarios:** *Pymes industriales* con sede social o establecimiento productivo en la Comunidad Valenciana.

**Actuaciones apoyables:** Actividades tendientes a desarrollar conocimientos tecnológicos para lograr productos o procesos productivos con especificaciones con propiedades técnicas diferenciadas de las existentes en el mercado. Los proyectos podrán abarcar: la creación de nuevos productos o procesos, pruebas experimentales y ensayos; y la elaboración de prototipos previos al inicio de la producción comercial.

**Tipo de ayuda:** Subvención a fondo perdido de hasta 35% del coste subvencionable del proyecto con carácter general. En caso de proyectos de cooperación con otra empresa o centro tecnológico, o en caso de que el solicitante sea un micro-empresa la ayuda podrá incrementarse en un 10%.

Los proyectos que puedan calificarse como investigación industrial podrán acceder a una intensidad de ayuda de hasta un 70%. Se entiende por investigación industrial la investigación planificada o estudios críticos que tienen por objeto obtener nuevos conocimientos que puedan resultar de utilidad en el desarrollo de productos o procesos nuevos, o para suscitar mejoras importantes de productos o procesos existentes.

**Plazo:** Durante todo el ejercicio 2004.

#### PROGRAMA DE CREACIÓN DE EMPRESAS DE BASE TECNOLÓGICA

Organismo Gestor: IMPIVA

**Beneficiarios:** *Pymes industriales* con sede social o establecimiento productivo en la Comunidad Valenciana que tengan la consideración de empresas de base tecnológica.

**Actuaciones apoyables:** Creación de nuevas empresas que exploten los resultados de proyectos de I+D realizados en la propia empresa o a consecuencia de acuerdos de transferencia de tecnología, de forma *exclusiva*, con Centros de Investigación públicos o privados.

**Tipo de ayuda:** Subvención a fondo perdido de hasta un 50% del coste elegible

del proyecto con un máximo de 60.000 Euros.

**Plazo:** Durante todo el ejercicio 2004.

#### PROGRAMA DE FORMACIÓN A MEDIDA PARA EMPRESAS

Organismo Gestor: IMPIVA

**Beneficiarios:** *Pymes industriales* con sede social o establecimiento productivo en la Comunidad Valenciana.

**Actuaciones apoyables:** Cursos de formación de carácter tecnológico presentados por pymes diseñados e impartidos por Centros Tecnológicos y otras Entidades sin ánimo de lucro. Tendrán una duración de entre 10 y 30 horas, y podrán prepararse para un grupo de entre 1 y 10 trabajadores. Podrán desarrollarse tanto en las instalaciones del Instituto Tecnológico como en las de la propia empresa solicitante. Todos los alumnos deben ser residentes en la Comunidad Valenciana y serán empresarios o trabajadores en activo.

**Tipo de ayuda:** El importe máximo de la ayuda por proyecto podrá llegar hasta el 45% de los costes elegibles, con un máximo de 90 Euros por hora de formación.

**Plazo:** Presentación de solicitudes hasta el 28 de octubre de 2004. Envío provisional por internet hasta el 27 de octubre y presentación de solicitud para convalidar el envío por internet hasta el 9 de noviembre.

### Ámbito Nacional

#### FINANCIACIÓN DE PROYECTOS DE I+D+I EMPRESARIALES

Organismo Gestor: Centro para el Desarrollo Tecnológico Industrial (CDTI)

**Beneficiarios:** Sociedades Mercantiles con capacidad técnica para desarrollar un proyecto de investigación, desarrollo o innovación tecnológica y capacidad financiera para cubrir con recursos propios un mínimo del 30% del presupuesto total del proyecto.

**Actuaciones apoyables:**

1. Proyectos de Desarrollo Tecnológico
2. Proyectos de Innovación Tecnológica
3. Proyectos de Investigación Industrial Concertada

**Tipo de ayuda:** Créditos a tipo de interés "cero" y con largo plazo de amortización que cubren hasta el 60% del presupuesto total del proyecto. El Centro solo apoya proyectos viables técnica y económicamente, pero no exige garantías reales a la empresa promotora para la concesión de

sus créditos. La financiación que presta el CDTI proviene básicamente de los recursos propios del Centro y del Fondo Europeo de Desarrollo Regional (FEDER).

Herramientas financieras:	Tipo interés	Plazo (años)	Crédito (% presup.)
Proy. de Desarrollo Tecnológico	0%	7-9	60%
Proy. de Innovación Tecnológica	0%	7-9	40%
Proy. vinculados a prog. internacionales	0%	9	60%
Proy. de Investigación Industrial Concertada	0%	7-9	Hasta 60%
Línea de financiación bancaria CDTI-ICO	Euribor-0,50%	5-7	70%

Estos créditos se caracterizan por incluir una cláusula de riesgo técnico según la cual, en el caso de que el proyecto no alcance sus objetivos técnicos, la empresa queda exenta de reintegrar la totalidad del préstamo.

#### PROGRAMA ARTE/PYME II

Organismo Gestor: MICYT - FONDO SOCIAL EUROPEO

**Objetivo:** Ayudar a las Pequeñas y Medianas Empresas (PYME) a su integración en la Sociedad de la Información, cofinanciando proyectos basados en el comercio electrónico que involucren la utilización de Servicios Avanzados de Telecomunicaciones (SAT) para satisfacer necesidades comunes de colectivos de PYME.

**Beneficiarios:**

1. Las *organizaciones públicas o privadas* que, sin ánimo de lucro, tengan la finalidad de prestar servicios de apoyo a las PYME, mediante la realización de proyectos comunes de asistencia o la promoción de servicios que contribuyan a la mejora de la competitividad de la PYME.

2. *Agrupaciones de interés económico de empresas* que cumplan la finalidad del párrafo anterior.

El Programa ARTE/PYME II va dirigido a las PYMES como destinatarios finales de los proyectos.

**Actuaciones apoyables:** Proyectos basados en el comercio electrónico cuyos objetivos puedan encuadrarse dentro de alguna de las siguientes líneas de actuación:

- a. Estudios de necesidades y viabilidad
- b. Proyectos piloto
- c. Implantación de Centros de Servicios Avanzados de Telecomunicación.
- d. Promoción del uso de Servicios Avanzados de Telecomunicación

**Tipo de ayuda:** El importe de las subvenciones podrá, aisladamente o en

## Ayudas y Subvenciones

concurrancia con otras subvenciones o ayudas, superar el 60% del coste de la actividad a desarrollar por el beneficiario. Solamente serán subvencionables las actividades cuyo gasto se haya comprometido con fecha posterior a la presentación de la solicitud y anterior a la fijada para la finalización del proyecto. Los bienes subvencionados deberán destinarse a los objetivos que justifican la concesión de la subvención durante un período mínimo de tres años.

**Plazo:** Hasta el 30 de junio de 2006

### PROGRAMA OPERATIVO DE INICIATIVA EMPRESARIAL Y FORMACIÓN CONTINUA DEL FONDO SOCIAL EUROPEO

### PROGRAMA DE FORMACIÓN EN TELECOMUNICACIONES (PROGRAMA FORINTEL)

**Organismo Gestor:** MCYT - FONDO SOCIAL EUROPEO

**Objetivo:** Organización e impartición de acciones formativas presenciales, a distancia (teleformación) o mixtas sobre materias relacionadas con los servicios avanzados de telecomunicaciones y las tecnologías que les proporcionan soporte.

**Beneficiarios:** *Empresas y Organismos Intermedios.*

**Actuaciones apoyables:**

1. Actuaciones de formación general.
2. Actuaciones de formación de usuarios de telecomunicaciones y nuevas tecnologías de la información.
3. Actuaciones dirigidas a la formación de profesionales que desempeñen puestos de trabajo relacionados con las telecomunicaciones y las TIC.

**Tipo de ayuda:** Subvención del 70 % del coste de la actuación, cualquiera sea el tipo de beneficiario, excepto si se trata de grandes empresas, en cuyo caso la subvención sería del 50 %. En las zonas del objetivo 1 (Comunidad Valenciana) el porcentaje de ayuda se incrementa un 10%.

**Plazo:** Hasta el 30 de junio de 2006

### Ámbito Internacional

### PROGRAMAS EUREKA E IBEROEKA

**Organismo Gestor:** CDTI - MCYT (PROFIT)

**Beneficiarios:** *Empresas y Centros Tecnológicos* capaces de realizar proyectos de I+D de carácter aplicado en colaboración con otras empresas y/o Centros Tecnológicos de otros países de Eureka e Iberoeika.

**Actuaciones apoyables:** Realización de proyectos tecnológicos internacionales, orientados hacia el desarrollo de productos, procesos o servicios con claro interés comercial en el mercado internacional y basados en tecnologías innovadoras.

**Tipo de ayuda:** Cada país asume la financiación de sus empresas e institutos tecnológicos

- FASE DE DEFINICIÓN: Un 75% de subvención a través del Programa de Fomento de la Investigación Técnica (PROFIT) del Ministerio de Ciencia y Tecnología.

- FASE DE DESARROLLO: Un 60% con créditos CDTI sin intereses a pagar en un plazo máximo de 8 años. Hasta un 35% en subvenciones con fondos PROFIT compatibles con otras subvenciones autonómicas o regionales.

**Plazo:** Durante todo el ejercicio 2004

### VI PROGRAMA MARCO DE LA UNIÓN EUROPEA

**Organismo Gestor:** COMISIÓN DE LA UNIÓN EUROPEA

**Objetivo:** Conseguir una investigación más centrada e integrada a escala comunitaria y articular y fortalecer las bases del Espacio Europeo de Investigación. Fomentar la participación de las PYME.

**Beneficiarios:** *Empresas y Centros Tecnológicos* capaces de realizar proyectos de I+D+i en colaboración con otras empresas y/o Centros Tecnológicos de países miembros de la UE o países Asociados.

**Actuaciones apoyables:** Realización de proyectos de carácter innovador.

**Tipo de ayuda:** Subvención a fondo perdido para empresas y Centros Tecnológicos. La participación de las PYME está abierta en las distintas prioridades temáticas y a través de los diferentes instrumentos (proyectos integrados, redes de excelencia, proyectos específicos de investigación focalizada). Existen espacios específicos de apoyo a las PYME, en forma de acciones de investigación cooperativa (Proyectos Craft) y colectiva (Proyectos Colectivos).

**Plazos:**

• **17/12/03 - 21/10/04 - Referencia: FP6-2003-SME-1-Proyectos de investigación cooperativa (CRAFT).** Actividades horizontales de investigación con participación de las PYME. Cualquier tema de todo el campo científico.

• **10/05/03 - 27/04/2006.** Convocatoria de manifestaciones de interés para la prestación de asistencia en el ámbito de diversas actividades técnicas, administrativas y organizativas propias de las Direc-

ciones que participan en el programa de TSI, a partir de los objetivos del programa de trabajo 2003-2004 (lista disponible en <http://www.cordis.lu/calls/>). Las candidaturas podrán presentarse hasta 3 meses antes de la conclusión del período de validez de la lista.

• **17/12/03-31/12/2004. Tecnologías futuras y emergentes** - Sistema abierto (presentación continua). Diario Oficial de las Comunidades Europeas (DOCE) 17.12.2002 ES C 315/12.

• **27/05/04-29/09/04. 3ª convocatoria.-FP6-2004-IST** - Estado: en proyecto.

• **Dic/04-Abr/05. 4ª convocatoria.-FP6-2004-IST-1. 5ª convocatoria.-FP6-2005-IST-2.**

**Ver Oficina Española de Ciencia y Tecnología (SO S+T): calendario de convocatorias:**

[http://sost.cdti.es/vipm\\_era.htm](http://sost.cdti.es/vipm_era.htm)

**Sociedad de la Información (ISTI):** Es la prioridad temática con mayor presupuesto en el VI Programam Marco (3.625 M€). Su contenido está estructurado como ilustra el siguiente esquema:



El propósito es asegurar la competitividad en Europa mediante actividades de investigación, desarrollo e innovación por medio de tecnologías genéricas y aplicadas. El objetivo es avanzar en la futura generación de tecnologías que permitan integrar el mundo digital en un entorno cotidiano permitiendo el acceso a una gran cantidad de aplicaciones y servicios de fácil utilización.

Cuatro líneas generales definen las prioridades de investigación: Investigación aplicada para hacer frente a grandes retos económicos y sociales, Tecnologías de comunicaciones y tratamiento de la información, Componentes y microsistemas, Tecnologías de las interfaces y del conocimiento. Además se cubren otras tres áreas: Tecnologías Futuras y Emergentes (FET), Bancos de prueba para las redes de investigación y Acciones de apoyo generales.

**Más información:**

[otri@iti.upv.es](mailto:otri@iti.upv.es)