

# Fisterra 2: software libre para gestión empresarial

Alejandro García Castro, José María Casanova Crespo,  
Juan José Sánchez Penas, José Dapena Paz  
Igalia: Ingeniería en Informática y Software Libre  
Gutenberg 34B 2º, 15008, A Coruña, Galicia, España

10 de enero de 2004

©2004, Igalia S.L. <sup>1</sup>

## Resumen

**Fisterra 2** constituye una evolución en las aplicaciones de gestión empresarial en el ámbito del software libre. Partiendo del trabajo realizado por **Igalia** en la primera versión, publicada en 2003, se ha rediseñado una nueva arquitectura multicapa, más genérica y ambiciosa, que verá la luz en su versión estable durante el año 2004.

En el artículo se realiza una evaluación sobre los distintos proyectos libres orientados al mundo de la gestión y, partiendo de éste, se justifica y explica la aparición de la nueva versión de **Fisterra**.

## 1. Introducción: necesidad de aplicaciones empresariales

En los últimos años, las plataformas basadas en software libre y su nuevo modelo de desarrollo cooperativo, y en particular el sistema **GNU/Linux** y sus aplicaciones asociadas, se han convertido en una alternativa sólida a cualquier otra solución tradicional, basada en software propietario cerrado. El nacimiento de decenas de nuevos proyectos cada mes, la adopción cada vez mayor del software libre por parte de administraciones públicas de todo el mundo, e incluso el apoyo explícito de grandes corporaciones como **IBM**, **Sun** o **HP**, son pruebas contundentes de la evolución de este movimiento que tiene apenas dos décadas y cuya explosión se está viviendo en los últimos cinco años.

El avance del software libre tiene relación con distintos aspectos, que se pueden ver en la mayoría de los

casos como complementarios: en ocasiones es decisiva la libertad que otorga al usuario el hecho de disponer del código fuente y de la posibilidad de analizarlo, modificarlo y redistribuirlo; en otras, cobra especial interés la reducción del TCO (coste total de propiedad) o la muchas veces mayor calidad técnica de las soluciones libres. En cualquier caso, miles de usuarios (empresas, administraciones o usuarios finales) están optando a cada vez mayor velocidad por soluciones innovadoras basadas en una nueva forma de entender el desarrollo de software.

Esta evolución del sistema va pareja a la aparición de soluciones (aplicaciones, distribuciones, servicios) para los distintos ámbitos y sectores de utilización de software libre. Cada vez son menos los huecos en los que hay que recurrir a sistemas cerrados porque el conjunto de sistemas operativos y aplicaciones libres no son capaces todavía de satisfacer las demandas de los usuarios. Si hace unos años las limitaciones de las soluciones de escritorio era una crítica habitual a los sistemas libres, en la actualidad esa carencia ya no existe, gracias a proyectos como **KDE** o **Gnome**, que han equiparado la calidad del escritorio **GNU/Linux** a la de cualquier sistema propietario. No obstante, todavía existen algunos ámbitos en los que faltarán unos años para que esta situación llegue; entre ellos están el del diseño gráfico, el de las herramientas de **CAD**, o, atendiendo a lo que más interesa para el presente artículo, el sector de las aplicaciones de gestión empresarial (**ERP**, **CRM**, contabilidad, etc.).

Existe un hueco por llenar, que se corresponde con las tecnologías, herramientas y aplicaciones adecuadas para construir sistemas de gestión eficientes, eficaces, flexibles, modernos y de calidad. **Fisterra** es un proyecto nacido con la vocación de colaborar en este trabajo, intentando que los sistemas libres, además de todos los sectores ya “conquistados”, puedan acceder a nuevos ámbitos en los que su aplicación traería mayor libertad para todos los participantes.

<sup>1</sup>Copyright Igalia, S.L. Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled “GNU Free Documentation License”.

## 2. Aplicaciones de gestión en GNU/Linux

La necesidad de aplicaciones y tecnologías de desarrollo especializadas en la gestión empresarial es algo difícilmente cuestionable. GNU/Linux, por la herencia de UNIX, nace como un sistema con especial fuerza en la parte servidor, lo que conlleva que su mayor cuota de mercado durante muchos años haya sido los servidores de intranet y de Internet (el ejemplo más utilizado es el del servidor web Apache). Progresivamente, esta mayor capacidad para el “lado servidor” se ha ido desplazando al *desktop* (escritorio del usuario final), y esto abre nuevas posibilidades entre las que, en el ámbito empresarial y de gestión, es cada vez más evidente el de la aplicación de software libre para solucionar las necesidades que tradicionalmente cubrían (y todavía cubren en la mayoría de los casos) los ERP/CRM propietarios de las grandes empresas de desarrollo de software (SAP, Navision, Libra, etc.).

En los últimos años, han nacido iniciativas derivadas de la coincidencia por parte de distintos grupos de desarrolladores o empresas en la necesidad de software libre de gestión. Algunas de estas iniciativas están en sus primeras fases todavía y cuentan con pocos apoyos (grupo de desarrolladores limitado y problemas de financiación), pero otras comienzan a vislumbrarse como proyectos de gran interés. Comentamos a continuación algunas de las iniciativas existentes:

- GNUe [1] (GNU Enterprise): metaproyecto que forma parte del proyecto GNU, y que tiene tres objetivos claros: un conjunto de herramientas, incluyendo interfaces de usuario, generadores de informes, módulos para la creación de aplicaciones cliente/servidor de gestión, etc.; un conjunto de paquetes que implementen, utilizando las herramientas anteriores, un ERP completo; y finalmente, la creación de una comunidad de usuarios alrededor de este proyecto. GNUe nace en marzo de 2000, aunque toma las riendas de varios proyectos anteriores que ya se habían iniciado en 1996. En estos momentos las herramientas (servidor de aplicaciones, generador de informes, generador de formularios, navegador, etc.) comienzan a ser utilizables, aunque sus versiones no son todavía muy estables. Los paquetes están todavía en las primeras fases de desarrollo, o incluso sin comenzar. No obstante, ya existen tres proyectos que utilizan las herramientas de GNUe para implementar directamente sus programas de gestión: Luca [10], GNUe Small Business [11] y Rent-Free [12]. El lenguaje escogido para el desarrollo es Python.
- GNUCash [2]: aunque con frecuencia no es citado como una herramienta de gestión empresarial,

por estar claramente enfocado a la administración contable más doméstica, el proyecto tiene características interesantes que podrían ser utilizadas para una contabilidad más empresarial. Es un proyecto muy estable, con licencia GPL y desarrollado con tecnología GNOME.

- Compiere [3]: iniciado en 1999 por la empresa del mismo nombre, se trata de un ERP desarrollado en Java y con interfaz web. Es multiplataforma y está publicado con licencia Mozilla (MPL), en la actualidad está ya en una versión completamente estable. El servidor está basado en JBoss [13], con el desarrollo basado en servlets y EJBs. El proyecto tiene en la actualidad una gran actividad.
- G-CTB [5] (GNU ConTaBilidad): implementación de una aplicación de contabilidad clásica, con licencia GPL, que utiliza tecnología GNOME para la interfaz gráfica y el acceso a datos, está desarrollado en C, y se encuentra todavía en versiones iniciales, además de encontrarse el proyecto completamente parado en el último año.
- ASPL-Fact [4]: proyecto iniciado en el 2000 e impulsado por la empresa ASPL (Advanced Software Production Line) que pretende crear una arquitectura modular para el desarrollo de aplicaciones de gestión en GNU/Linux. Por el momento el proyecto se ha centrado en la definición de una arquitectura flexible basada en tecnología GNOME y en el protocolo BEEP para la comunicación cliente/servidor; una primera versión de esta arquitectura está prevista para agosto de 2003. Todo el desarrollo es bajo licencia GPL y con C como el lenguaje principal de programación.
- Bulmages [6]: aplicación de contabilidad para GNU/Linux, con licencia GPL, que utiliza tecnología KDE, desarrollado en C++, y se encuentra en la actualidad en una versión utilizable, con las funcionalidades básicas y implementadas. Su desarrollo está siendo coordinado por miembros del grupo de usuarios de GNU/Linux de Mallorca. El proyecto inició su actividad en noviembre de 2002.
- Facturalux [7]: aplicación de gestión tipo ERP para GNU/Linux, con licencia GPL y tecnología KDE (QT, C++, Kugar), utiliza una arquitectura denominada A3D (Arquitectura Abierta de Aplicaciones Dinámicas), que almacena código y datos en la BD utilizando el estándar XML y ECMAScript; BEEP [14] es el protocolo de comunicación escogido. El proyecto está impulsado por InfoSiAL, desde principios del 2001, y mantiene una actividad alta en la versión actual del código, que todavía es un prototipo.

- GestíONG [8]: implementación de una aplicación de gestión de ONGs y sociedades sin ánimo de lucro, haciendo especial hincapié en la creación de un programa estándar de contabilidad. El proyecto dio comienzo mayo de 2003 y está disponible ya un prototipo con funcionalidad básica. El programa, con licencia GPL, utiliza también tecnología KDE para la implementación (QT, C++, Kugar) y presenta una actividad considerable actualmente.
- Gestión Libre [9] en Hispalinux: proyecto que pretende coordinar acciones (documentación, desarrollo, especificaciones) relacionadas con la introducción de software libre de gestión en empresas y administraciones públicas. El proyecto existe desde el año 1999.

En este contexto, en el que se ve claramente que están surgiendo iniciativas en los últimos 3 años para la creación de software de gestión empresarial, y también se puede observar que la mayor parte de los proyectos están todavía lejos de proporcionar una solución sólida al usuario, en el año 2003, nace el proyecto **Fisterra**. El proyecto es impulsado y desarrollado por **Igalia** y basado en tecnologías **GNOME**. En los siguientes apartados se detallan los aspectos más importantes del proyecto, describiendo la arquitectura propuesta para la nueva versión, que constituye un paso mucho más ambicioso para colocar a **Fisterra** definitivamente entre las alternativas más interesantes para la construcción de soluciones ERP/CRM en **GNU/Linux**.

### 3. Fisterra 2

#### 3.1. Presente del proyecto: **fisterra-base** y **fisterra sectorial**

Actualmente **Fisterra** es un proyecto activo, desde el momento de su publicación hasta la fecha de hoy se ha trabajado mucho en el diseño y arquitectura de la nueva versión. Se está invirtiendo un gran esfuerzo en definir las necesidades de las aplicaciones de gestión empresarial tanto desde el punto de vista tecnológico como conceptual. La versión original, más orientada a un sector, del proyecto, ha sido generalizada hacia la creación de un *framework* para el desarrollo de aplicaciones de gestión de diversos sectores empresariales. Además, se ha potenciado la creación de herramientas que faciliten el trabajo en comunidad, por lo que se espera que el año 2004 sea clave para el crecimiento en el número de usuarios y desarrolladores de **Fisterra**. **Igalia** está en conversaciones con empresas especializadas en la gestión de negocio para la colaboración en el enriquecimiento de la solución y para la distribución a mayor escala de sistemas basados en esta tecnología.

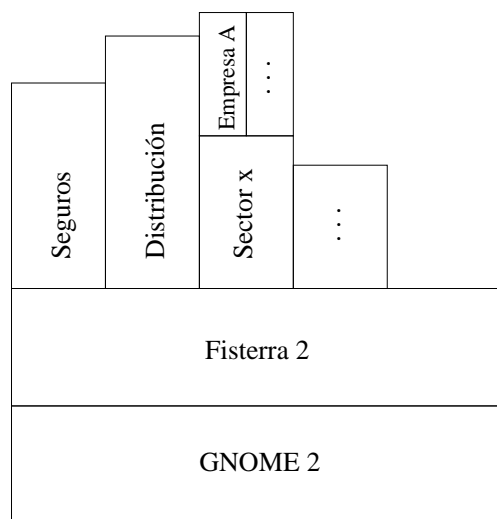


Figura 1: Estructura de **Fisterra 2**

**Fisterra 2** consta hoy por hoy de un núcleo sobre el que se apoyan desarrollos de **Igalia** orientados a desarrollos verticales para sectores específicos.

Durante el primer mes de 2004 se está dando un importante giro en cuanto a las herramientas para el trabajo en comunidad dentro del proyecto. Una nueva web, más útil y dinámica, será el eje central de coordinación del proyecto. El trabajo del equipo de desarrollo será realizado en un servidor de CVS público y accesible para cualquier persona interesada. Una larga lista de herramientas de ayuda como **bonsai**, **tinderbox**, **bugzilla** o **natzilla** estarán a la disposición de la comunidad para coordinar el desarrollo de **Fisterra**.

Uno de los intereses principales es la creación de una comunidad de desarrolladores fuerte que trabaje en el desarrollo aplicaciones de gestión basadas en software libre; por ello estamos intentando la colaboración con otros proyectos, acordando arquitecturas comunes que nos puedan llevar en un futuro cercano a la creación de estándares de facto para la creación de este tipo de aplicaciones.

**Fisterra 2** se divide en un framework genérico de trabajo, que es conocido como *fisterra-base*, y el conjunto de aplicaciones sectoriales genéricas desarrolladas utilizando las bibliotecas y facilidades aportadas por dicho framework. Las aplicaciones sectoriales implementan un conjunto de procesos de negocio que son interesantes para la lógica de trabajo de ese sector.

Intentando la generalización de los procesos de negocio de una empresa **Fisterra** se ha dividido en las siguientes partes principales, que están siendo implementadas de forma progresiva para **Fisterra 2**: gestión de contacto con clientes (CRM); gestión de ventas (del presupuesto-venta a la contabilidad); gestión de compras (del presupuesto-compra a la contabilidad);

Figura 2: Captura de pantalla de **Fisterra 2**

gestión de la logística (almacenes); gestión de fabricación (SCM); gestión de recursos humanos; gestión de tesorería; gestión de la contabilidad y finanzas; gestión de negocio, táctica y estrategia (datawarehouse, OLAP); gestión de la integración (legacy); gestión de la aplicación (administración); gestión de proyectos; y gestión de calidad.

### 3.2. Objetivos de **Fisterra 2**

**Fisterra 2** nace como la evolución tecnológica en el desarrollo de aplicaciones de gestión empresarial de **Igalia**. Tras la experiencia alcanzada con el desarrollo de la primera versión, se ha realizado una refactorización completa de la arquitectura para permitir desarrollos más escalables de sistemas de información para empresas.

Los objetivos de **Fisterra 2** son construir una solución más general y flexible que sea adaptable de forma sencilla a nuevos negocios.

Entre los objetivos a alcanzar con la nueva versión se encuentran: incorporación de la tecnología GNOME2; arquitectura multicapa; y proporcionar un framework genérico y un conjunto de ejemplos de aplicaciones para sectores de negocio para facilitar el desarrollo y adaptación de nuevos módulos sectoriales sin partir de cero.

### 3.3. Diseño del sistema

**Fisterra 2** establece una separación entre cliente y servidor. El cliente tiene como funciones la interacción con el usuario final mediante la interfaz gráfica. Es necesario poder ejecutar instancias del cliente en múltiples equipos. Su implementación es totalmente independiente de la del servidor siempre que cumpla las interfaces de comunicación.

El servidor se encarga de gestionar el proceso de negocio de la aplicación. Realizar las operaciones que enviadas por los clientes, a la vez que gestiona el acceso a la base de datos, el control de accesos y las autorizaciones.

#### 3.3.1. Cliente **Fisterra 2**

El cliente en **Fisterra** es la parte visible al usuario, la cual está controlada por un autómata de estados que determina el flujo de la aplicación. La estructura interna del cliente está definida siguiendo el patrón de diseño MVC (*Model-View-Controller*).

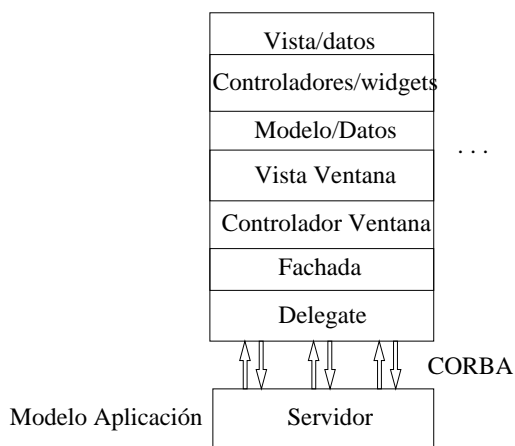


Figura 3: Diseño del Cliente de **Fisterra2**

Cada ventana está formada por tres tipos de componentes: *vista* descrita mediante ficheros XML de glade; *controlador*, que rellena la ventana en base a su modelo, gestiona los eventos y callbacks; y *widgets* para la representación de los datos del dominio del negocio utilizaremos widgets creados según el patrón model-view-controller, de forma similar a la solución aplicada para los *GtkTrees*. De esta forma tendremos modelos que representarán estructuras de datos de dominio complejas orientadas a al trabajo visual y controladores/vista encargados del mostrado de estos objetos. Esta solución nos permitirá la creación múltiples vistas sin tener que cambiar la representación de datos del cliente.

Para la gestión de ventanas y su creación usamos un sistema de controlador principal en el que se registran las ventanas para poder ser construidas en el sistema. Para la construcción de una ventana se hace una petición a este controlador que utilizará *builders* para la creación de las mismas.

Por abajo de los controladores de ventana encontramos las fachadas encargadas de implementar las llamadas a los servicios utilizando los modelos que representan los datos de las ventanas. La abstracción de la tecnología de comunicaciones, en este caso CORBA, se consigue mediante una capa de delegación encargada de ocultar los procesos necesarios para realizar la comunicación.

#### 3.3.2. Servidor **Fisterra**

La novedad en **Fisterra 2** se encuentra principalmente en que se separa del cliente todo el comportamiento de gestión de negocio, autenticación y acceso a datos en un servidor. Éste proporciona distintos servicios que son utilizados por el cliente para la ejecución de la aplicación.

El sistema servidor permite disponer de un único punto de acceso a la base de datos, de forma que to-

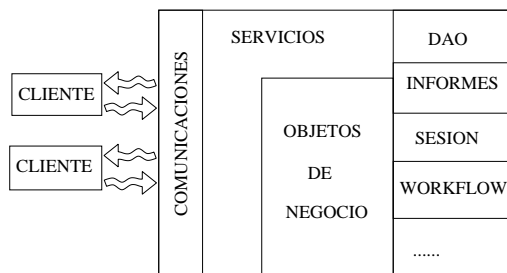


Figura 4: Diseño servidor *Fisterra* 2

das las operaciones contra ella se ejecutarán desde un único punto. Esto además elimina la dependencia de la base de datos para la gestión de la concurrencia. Para la creación y gestión de datos se ha creado una estructura basada en un sistema de generación de código, que permite construir la estructura de objetos de gestión de datos de negocio de forma eficiente. EL objeto principal es el *barnacle* del que heredan el resto de objetos y que les proporciona una serie de características estándares.

El servidor permite gestionar la operativa de distintos clientes y disponer de una gestión global de autorizaciones y control de las operaciones.

La capa de comunicaciones implementa las interfaces disponibles para su utilización en el cliente, se encarga de aceptar las peticiones en el cliente y redirigirlas al servicio correspondiente. La tecnología de comunicaciones se mantendrá abstracta al sistema, pero para las implementaciones iniciales se utilizará CORBA.

En la capa de servicios se dividen en 2 tipos, servicios de utilidad y servicios de negocio. Ambos tipos se han implementando utilizando una estructura de objetos con un tipo padre encargado de implementar las características generales de los mismos, los EGBs (*Enterprise Gnome Barnacles*). Los EGBs nos permiten la separación del código de gestión de negocio. Los EGBs específicos de servicio se encargan de la implementación las funciones que exportan los módulos. Entre los servicios utilidad se encuentran el acceso a datos, gestión de workflow, gestión de sesión y elaboración de informes. Los servicios de negocio implementan comportamientos del negocio soportado por la aplicación y se apoya sobre los servicios utilidad, de esta forma las personalizaciones del modelo de negocio consistirían en modificar o parametrizar estos servicios.

La nueva arquitectura está detallada en la figura 4

### 3.4. Tecnologías utilizadas

*Fisterra* utiliza como base el conjunto de tecnologías que conforman el Gnome SDK, para las interfaces de usuario y el servidor de aplicaciones. Sobre éstas, integra e introduce otras tecnologías, para dar soporte

a persistencia, e implementar la comunicación, tal y como hemos podido ver en apartados anteriores. Sobre ellas, *Fisterra* constituye por sí mismo un completo framework de desarrollo de aplicaciones.

A continuación detallamos las principales tecnologías que se utilizan en *Fisterra*.

#### 3.4.1. Tecnologías GNOME

**GTK+** Librería que proporciona los widgets y controles. En *Fisterra* se utilizan los estándares de desarrollo de interfaces del escritorio GNOME, y se observa el respeto por los estándares que define, en especial las HIG<sup>2</sup>.

**LibGlade** Librería para la descripción visual de interfaces de usuario, y el uso de estas descripciones en aplicaciones. La mayor parte de las interfaces de usuario de *Fisterra* se desarrollan con esta herramienta.

**GObject** Sistema que permite realizar orientación a objetos en C, usado por el proyecto GNOME. Los objetos de negocio y la persistencia se implementan utilizando GObject.

**LibXML, libXSLT** Librerías que implementan operaciones para manipulación de ficheros XML, y transformaciones XSL. Se utilizan en *Fisterra* como formato de intercambio.

**Orbit** Implementación ligera de CORBA, protocolo de invocación remota de métodos, y comunicación de objetos interoperable. Se utiliza para la comunicación entre los distintos servicios del sistema, y entre cliente y servidor.

**libGDA** Librería de acceso a bases de datos del proyecto Gnome. Se utiliza para implementar el acceso genérico a bases de datos relacionales, y proporcionar así persistencia a la capa de negocio.

#### 3.4.2. Otras tecnologías

**PostgreSQL** Gestor de bases de datos relacional, que implementa un gran conjunto de funcionalidades de SQL, de forma eficiente. Es la base de datos elegida para el almacenamiento de información en *Fisterra*.

### 3.5. Publicación y colaboración

En el momento de elaboración de este artículo se encuentra publicada (además de la versión 1.5 completa y funcional, pero menos genérica) en la web la versión

<sup>2</sup>HIG (*Human Interface Guidelines*): Documento de normas y estándares sobre la implementación de interfaces de usuario en GNOME.

0.0.2 de **Fisterra 2**, está formado por un subconjunto de las funcionalidades de **Fisterra** sobre las que se han realizado las pruebas tecnológicas sobre el entorno GNOME 2, consistentes en la migración de la interfaz de Glade1 a Glade2 y la actualización de la librería libgda a su versión 0.12 desde la 0.2. Esta versión es simplemente para realizar pruebas de desarrollo, pero en el CVS público está ya disponible la versión inicial de *fisterra-base* y el código inicial sobre el que se está construyendo un ejemplo de aplicación para el sector de la distribución. La primera versión estable de *fisterra-base* y el código para el sector de la distribución serán publicadas en el primer y segundo trimestre de 2004, respectivamente.

Igalia está trabajando actualmente en buscar puntos de encuentro con otros proyectos para intercambiar experiencias y tratar de clarificar el futuro de las aplicaciones de gestión empresarial como Software Libre.

## 4. Conclusiones y trabajo futuro

En este artículo hemos presentado el presente y planes futuros de un nuevo proyecto de software libre llamado **Fisterra**. El desarrollo del ERP Open Source se encuentra en estos momentos inmerso en la segunda etapa de generalización, con la definición de una arquitectura que marca el rumbo cara a un framework que permita definir con facilidad nuevas aplicaciones de gestión empresarial para sectores distintos de los ya implementados. Con la versión 1 ya estabilizada y en funcionamiento, el proyecto y sus desarrolladores están embarcados en una segunda fase intensa y compleja de la que debería salir, si todo sigue como está previsto, una versión avanzada de la segunda versión, siguiendo las decisiones de diseño arquitectónico que se explican en el artículo. Esta nueva versión será publicada en sus distintas versiones durante la primera mitad del año 2004.

Con estas líneas trazadas, se deja claro cuál es el enfoque actual del proyecto, que camina hacia un ERP abierto en dos niveles diferentes: por un lado los componentes básicos que forman el framework de desarrollo de aplicaciones basadas en **Fisterra** (solución horizontal común), y por otro las implementaciones de los distintos sectores verticales especializadas y posiblemente adaptables para las necesidades de una empresa en concreto. Tres sectores están siendo desarrollados ya, y sus partes comunes están siendo abstraídas, y dichas implementaciones formarán parte de las próximas versiones de **Fisterra 2**, completamente integrada con las más recientes tecnologías del proyecto GNOME.

A pesar de que los desarrolladores del proyecto se encuentran satisfechos con las decisiones tomadas, tanto tecnológicas como de diseño de arquitectura, es muy importante el intercambio y la colaboración con otros

proyectos con objetivos similares, con lo que en estos momentos se están llevando a cabo iniciativas para contactar con los coordinadores de distintos proyectos de software de gestión empresarial libre. En el futuro, **Fisterra** buscará todavía más la sinergia con otras iniciativas, intentando ser una solución mejor y más completa para los usuarios de software libre.

## 5. Agradecimientos

Los autores del artículo quieren agradecer a Javier Vázquez Lamas, Xavier Castaño García, Alberto García González, José Juan González Alonso, Javier Fernández García-Boente, Alejandro Piñeiro Iglesias, Xavier Rodríguez Calvar, Eloy Froufe y Sergio Villar Senín su colaboración en el desarrollo del proyecto **Fisterra**.

## Referencias

- [1] GNU Enterprise: <http://www.gnenterprise.org>
- [2] GNU Cash - Open Source accounting Software: <http://www.gnucash.org>
- [3] Compiere, Smart Open Source ERP Software with integrated CRM Solutions: <http://www.compiere.org>
- [4] ASPL-Fact, facturación para Linux: <http://fact.aspl.es>
- [5] G-CTB: GNU ConTaBilidad: <http://www.tecnopro.net/gctb>
- [6] Bulmages, contabilidad para Linux: <http://bulmages.bulma.net>
- [7] FacturaLUX - Software ERP de código libre: <http://www.facturalux.org>
- [8] GestíONG, Software solidario para organizaciones solidarias: <http://gestiong.sourceforge.net>
- [9] Gestión Libre: <http://gestion-libre.hispalinux.es>
- [10] LUCA, programa de gestión: <http://www.vialibre.org.ar>
- [11] GNUe Small Business Edition: <http://www.gnu.org/software/gnue-sb>
- [12] Rent-Free: <http://savannah.nongnu.org/projects/rentfree>
- [13] JBoss, Professional Open Source: <http://www.jboss.org>
- [14] RoadRunner, Industrial Strength BEEP Toolkit: <http://rr.codefactory.se>
- [15] Open Source Initiative: <http://www.opensource.org>
- [16] García Castro, Dapena Paz. GNOME for business appliances: case study and architecture proposal. GUADEC IV Dublin, 2003.