

Gestión Libre

Una oportunidad para todos

Fernando Antonio Acero Martín

Copyright © 2002 Fernando Antonio Acero Martín

Este documento se publica para el V congreso de HispaLinux. Se otorga permiso para copiar, distribuir y/o modificarlo bajo los términos de la Licencia de Documentación Libre GNU, Versión 1.1 o cualquier otra versión posterior publicada por la Free Software Foundation, no habiendo Secciones Invariantes. Puede consultar una copia de la licencia en: <http://www.gnu.org/copyleft/fdl.html>

Historial de revisiones

Revisión 01

03/07/2002

FAAM

Tabla de contenidos

1. Introducción al proyecto	1
2. Análisis inicial	3
2.1. Las licencias y los análisis funcionales	4
2.2. El caso particular de la Administración	5
3. Análisis de las libertades y el modelo de desarrollo del software libre en relación a las aplicaciones de gestión. 6	
3.1. Libertades del software libre	6
3.2. El modelo de desarrollo del software libre	7
3.3. Otros problemas que se han dectado	8
4. El efecto de la presión que se está ejerciendo	9
4.1. Lo que está ocurriendo en realidad	9
5. La solución propuesta por HispaLinux	10
5.1. Lo que permitiría el uso de análisis funcionales	11
5.2. Materialización del proyecto Gestión Libre.	11
5.3. El proyecto Gestión Libre es una revolución	12
5.4. Lo que permite el proyecto Gestión Libre	12
5.5. Líneas de trabajo abiertas en la actualidad	13
5.6. Relación entre los análisis funcionales y los programadores	15

El proyecto Gestión Libre

El proyecto Gestión Libre de HispaLinux intenta hacer atractivo para la comunidad de programadores de software libre el desarrollo de aplicaciones de gestión para la Administración y las empresas. Para lograrlo, HispaLinux propone crear una base de datos de conocimiento basada en análisis funcionales, con todo lo que pueda necesitar la comunidad de software libre al desarrollar aplicaciones de gestión para la Administración y las empresas.

1. Introducción al proyecto

Ante sus evidentes ventajas, el software libre está comenzando a despertar el interés de la Administración y las empresas. Desgraciadamente, a pesar de los enormes beneficios que puede reportar el software libre la sociedad en general, se está haciendo muy difícil su implantación por la escasez de aplicaciones de gestión. Si consideramos que las ventajas del software libre deben tener su eco en la sociedad y no quedar restringidas a la comunidad que lo ha hecho posible, es básico que esta comunidad ayude a crear el software de gestión que necesitan la Administración y las empresas. Para lograr este objetivo tan interesante, puede ser de gran ayuda que la Administración y las empresas expresen sus necesidades de gestión, usando un lenguaje formal que pueda ser entendido por los programadores de la comunidad de software libre. Partiendo de esta idea tan simple, el proyecto Gestión libre intenta ser ese punto de encuentro en el que utilizando el lenguaje común de los análisis funcionales, la Administración y las empresas pueden expresar sus necesidades a la comunidad de software libre.

El proyecto Gestión Libre (<http://gestion-libre.hispalinux.es>) quiere ser una respuesta viable y consensuada por todos los participantes en el proyecto, a todas las necesidades de gestión de las empresas y la Administración, que se puedan satisfacer con software libre. Gestión Libre también es un intento racional para mejorar, optimizar y vertebrar la capacidad de respuesta de la comunidad de código abierto ante la necesidad evidente de aplicaciones de gestión, basadas en software libre, para campos estratégicos en la economía.

La materialización de este proyecto se basa en la creación de una base de datos de conocimiento, que contendrá análisis funcionales con licencia libre de las aplicaciones de gestión que necesitan la Administración y las empresas. En esa base de datos, junto con los análisis funcionales, también se encontrarán la normativa, la legislación, las directrices de desarrollo, las especificaciones de requisitos, los módulos de formación, las herramientas aplicables y su documentación. Con toda esta información depurada, actualizada y estructurada, intentamos hacer atractivo para los programadores de software libre el desarrollo de las aplicaciones de gestión que necesita la sociedad moderna. Sin embargo, el desarrollo de este tipo de aplicaciones, tan lejanas a lo que habitualmente usa y necesita la comunidad de software libre es relativamente difícil si no se proporciona algo más. Por ello, mediante el uso de herramientas de e-learning y contenidos específicos, también se buscará la capacitación plena del programador en el desarrollo de aplicaciones de gestión.

Debemos tener en cuenta que aunque los análisis funcionales implican un modelo de desarrollo más formal, no suponen un cambio radical en los métodos y costumbres de la comunidad de código abierto. Solamente representan una matización en la forma de desarrollar el software de gestión. De hecho, es algo que ya se está implantando en algunos proyectos de software libre de gran envergadura, pero de forma restringida al equipo de trabajo que los lleva a cabo. Sin embargo, para algunos programadores este modelo puede implicar la necesidad de una capacitación en el uso de los análisis funcionales, algo que también habrá que atender. Sin embargo, con este pequeño coste educativo logramos potenciar y optimizar las características positivas de la comunidad de software libre, en un momento en el que puede ser interesante responder a las necesidades de la Administración y las empresas. Estos análisis funcionales seguirán un proceso similar al que se usa actualmente con el código fuente, serán depurados y después de la liberación de una versión estable, serán utilizados para el desarrollo de las aplicaciones de gestión. De hecho, en algunos casos muy concretos, los análisis funcionales se pueden asimilar al código fuente.

Una de las ventajas más interesantes de usar análisis funcionales en el ciclo de desarrollo, es la de permitir la participación de otros actores, como la Administración, asociaciones profesionales, universidades, particulares, fundaciones o empresas. De esta forma y sin que suponga un problema para nosotros, ampliamos el desarrollo a actores, que sin tener conocimientos de programación, pueden aportar ideas o soluciones muy interesantes y válidas, que se incorporarán a los análisis funcionales y a los programas, mejorando su funcionalidad y ahorrando trabajo al programador. Otra consecuencia directa de lo que acabamos de decir, es que el programador podrá rentabilizar su trabajo con facilidad y reducirá los riesgos derivados del desarrollo de aplicaciones que no tienen demanda en el mercado. Con el uso de los análisis funcionales, también se pretende que el desarrollo del software libre sea independiente del lenguaje de programación y de la ubicación física de los programadores. De esta forma, un programador residente en Barcelona no tendrá problemas a la hora de realizar una aplicación para el Ayuntamiento de Madrid y viceversa. Si los análisis funcionales son lo suficientemente completos y claros, el programador no necesitará contactar con los generadores de la necesidad y podrá desarrollar el software rápidamente y reduciendo los costes.

Es posible que nos encontremos ante una revolución en la forma de realizar software libre, que puede llegar a tener una gran relevancia en la comunidad de software libre. Revolución, que soluciona o reduce algunos de los problemas que aparecen cuando intentamos aplicar el software libre en la sociedad.

2. Análisis inicial

En la actualidad hay dos nichos de mercado que son muy interesantes de informatizar con software libre, las pymes y los ayuntamientos de menos de 50.000 habitantes. En estos dos ámbitos, por falta de recursos económicos, es frecuente que no se recurra a soluciones basadas en la programación a medida y se utilicen aplicaciones "pret a porter", mucho más económicas y rápidas de implementar. A la vista de ello, podría ser interesante cubrir estas necesidades con software libre de unas características similares, pero personalizable y con un buen soporte. Si se logra este objetivo inicial, incluso se podría conseguir un mejor sostenimiento de la comunidad de software libre, que también puede contribuir a mejorar el PIB. Aquí nos encontramos ante unas economías de escala, en las que un pequeño esfuerzo de programación, se abre un mercado de grandes dimensiones. Para hacernos una idea de este mercado en España, podemos decir que las pymes representan el 80% de las empresas españolas, mueven el 90% del PIB y reciben una gran cantidad de ayudas oficiales.

A pesar de las ventajas que tiene el mercado del que hemos hablado antes, los programadores de software libre tienen algunos problemas para desarrollar estas aplicaciones. Por ejemplo, es frecuente que el programador de software libre tenga que controlar y ejecutar por él mismo todos los pasos y procesos de la implementación del software. Este hecho complica enormemente el desarrollo de ciertas aplicaciones que son lejanas a la comunidad de software libre. Con el sistema de desarrollo actual, se tienen que producir muchas "casualidades" para que aparezca una aplicación de gestión válida, lo que representa un enorme problema. Es evidente que no todo el mundo tiene el tiempo, los conocimientos o el acceso necesario a la Administración o a las empresas, para poder realizar un buen análisis funcional. Además, si el proyecto resulta ser muy complejo y no se dispone de análisis funcionales, es posible que no se pueda llevar a cabo y acabe siendo abandonado, impidiendo además, que otros programadores lo puedan retomar por las carencias en la documentación. Hay que destacar, que algunos intentos de desarrollo de aplicaciones de gestión libres, por no contar con unos buenos análisis funcionales, están muy limitadas y no responden a las necesidades reales del mercado.

Esta dificultad para acceder a los análisis de las aplicaciones de gestión, provoca que muchos programadores de código abierto se dediquen a trabajar en proyectos relacionados directamente con el sistema operativo y sus utilidades, cuando podrían dedicarse a proyectos relacionados con las empresas o la Administración, que pueden ser más interesantes profesional o económicamente. Quizás sea el momento de pensar en el sistema operativo como una vía para el desarrollo de aplicaciones libres, en lugar de considerarlos como la finalidad última de la comunidad de software libre. No quiero decir que se abandone el desarrollo y la mejora del sistema operativo, pero puede ser interesante dar algo de prioridad a ciertas tareas estratégicas, que pueden expandir las posibilidades del software libre y abrir otros caminos interesantes para la sociedad y para la comunidad de software libre.

Por todo lo que hemos dicho, el proyecto Gestión Libre se manifiesta muy interesante para muchos colectivos, pero vamos a analizar sus ventajas para uno muy especial, el de los estudiantes. Este colectivo es posible que sea el que tenga más dificultades para realizar análisis funcionales o aplicaciones de gestión realistas y viables de ser utilizadas. En la mayoría de los casos, los trabajos de los estudiantes de informática, además de no estar orientados al software libre, tampoco están orientados a casos reales, por lo que tienen un limitado valor pedagógico y no facilitan su integración laboral. Esto representa una pérdida de tiempo y un uso poco productivo de este interesante potencial. Si proporcionamos a los estudiantes el material de Gestión Libre, podrían basar sus prácticas en el desarrollo de estas aplicaciones, lo que puede implicar unos enormes beneficios para ellos y para la sociedad. Por si fuera poco, los programas que elaboren en sus prácticas tendrán el éxito asegurado por responder a necesidades reales del mercado. ¿hay una mejor forma de inserción laboral, auto-promoción profesional o de realizar prácticas realistas? Además, si consideramos que el software libre no es lo mismo que software gratis, se podrían crear empresas de base tecnológica en la universidad usando estos trabajos.

En nuestro análisis no debe faltar la posible influencia que tiene la ausencia de este tipo de aplicaciones en las empresas dedicadas a la comercialización de software libre. Muchos indicadores muestran que el mercado del software libre, considerado atípico por unos y condenado al fracaso por otros, están acusando una carencia de aplicaciones específicas para la Administración y empresas, en las que basar negocios de soporte y mantenimiento. Estas empresas están acusando las dificultades de vivir solamente de las oportunidades de negocio que les brinda el sistema operativo y sus herramientas y en especial, del mercado de los servidores. Sin un software de gestión de uso general, en muchos nichos del mercado empresarial y de la Administración, solamente se implantarán soluciones parciales o mixtas. Esto significa que por cada servidor que funcione con software libre, aparecerán un gran número de estaciones de trabajo funcionando con software propietario. Para minimizar esta situación, las presiones para la implantación del software libre en la Administración y las empresas deberían ir emparejadas con la aparición de soluciones de gestión que puedan satisfacer las necesidades que se generan. Como referencia tenemos el resultado de algunas estadísticas realizadas sobre el mercado del software libre. En ellas se indica que aunque muchas empresas tienen la certeza de que el software libre es mejor y más estable, pero uno de los principales problemas para su adopción se centra en la ausencia de software de gestión de bajo coste y rápida implantación.

2.1. Las licencias y los análisis funcionales

Antes hemos dicho que los análisis funcionales pueden tener una licencia libre, pero tenemos que considerar que no hay ninguna licencia de la FSF (Free Software Foundation) que se encuentre especialmente diseñada para los análisis

funcionales. Lo ideal sería conseguir una fórmula legal, válida en Europa, para que todo los trabajos derivados del uso de análisis funcionales libres también sea libre, de forma similar a lo que ocurre con el código fuente de los programas.

Por lo general y a la espera de poder disponer de una licencia adecuada en el ámbito nacional o europeo, los tipos de licencia que se pueden aplicar a los análisis funcionales dependerán de su formato, contenido y alcance. De este modo, si los análisis se encuentran contenidos en documentos de texto y contemplan conceptos generales de la creación del software, se les puede aplicar una licencia GDFL de acuerdo con la Ley de la Propiedad Intelectual. Sin embargo, si los análisis se creen con herramientas de modelado basadas en UML (Unified Modelling Language), Métrica 3 o similares, que se encuentran cercanos al código fuente, podrían liberarse con licencia GPL.

2.2. El caso particular de la Administración

Si la creación de software libre de gestión para las empresas se ve dificultada por el problema de los programadores para conocer lo que se necesita, esta situación se agrava especialmente cuando es necesario crear software para Administración. Las necesidades de gestión de la Administración son mucho más difíciles de analizar y conocer por la comunidad de software libre. Para solucionar este problema, es posible que la única solución esté en que la Administración proporcione sus necesidades, en la forma de análisis funcionales en Métrica 3, a la comunidad de software libre.

También debemos considerar que algunas iniciativas gubernamentales basadas en las nuevas tecnologías y dotadas de interesantes asignaciones económicas, se podrían materializar con software libre si se tuvieran proyectos viables. Gestión Libre, con sus características únicas, puede ser una perfecta incubadora para proyectos en los campos de las tecnologías de la información, desarrollo económico, apoyo a las pymes, apoyo a las corporaciones locales, e-learning, etc.

2.2.1. Las ventajas del uso de software libre

A pesar de los problemas para realizar los análisis funcionales, la utilización de software libre en la Administración y las empresas, tiene innumerables ventajas para la sociedad. Por ello y pensando en el bien común, debemos intentar que se implante a todos los niveles, por lo que consideramos muy interesante que la comunidad de software libre colabore con la Administración y las empresas para conseguir este objetivo. Entre las ventajas de utilizar software libre encontramos las siguientes:

- Permite la libertad de elección del software, hardware, soporte, formación y servicios.
- Protege la inversión en software, hardware, soporte, formación y servicios.
- Mejora la relación calidad/precio del software, hardware, formación y servicios.
- Garantiza la comunicación e interoperabilidad de aplicaciones.

- Mejora la seguridad.
- Favorece la privacidad.
- Permite el enriquecimiento tecnológico y económico del país.
- Permite la escalabilidad y adaptabilidad de las aplicaciones.
- Permite invertir en lo que realmente interesa, es decir, en la gestión, evitando los gastos en licencias.
- Aumenta la productividad a la hora de crear software.
- Permite el desarrollo de las ideas propias para la administración o el negocio.
- Minimiza el impacto de los códigos maliciosos.

Si reflexionamos sobre la lista anterior, veremos que algunas de las ventajas anteriores, podrían ser argumento suficiente para impulsar y promocionar el uso del software libre en la Administración y en las empresas. En HispaLinux estamos convencidos de que el software libre es la única alternativa para conseguir muchos de los objetivos de una Administración moderna y una mayor competitividad de las empresas.

3. Análisis de las libertades y el modelo de desarrollo del software libre en relación a las aplicaciones de gestión.

Hemos analizado algunas cuestiones relativas a la creación de software de gestión libre para la Administración y las empresas. Entre otras cosas, hemos justificado la necesidad de usar análisis funcionales y las bondades del software libre en la Administración y las empresas. Pero al margen de las dificultades que tiene la comunidad de software libre, para saber lo que necesitan la Administración y las empresas, puede haber otros problemas intrínsecos a la comunidad que dificulten o impidan la creación de estas aplicaciones de gestión. Comenzaremos analizando las libertades que definen al software libre y su posible influencia a la hora de crear aplicaciones de gestión.

3.1. Libertades del software libre

El software libre tiene una serie de características únicas y que lo diferencian del software propietario, que se materializan en cuatro libertades básicas:

- Libertad para ejecutar el programa con cualquier propósito.
- Libertad para estudiar el programa y adaptarlo a sus necesidades.

- Libertad para distribuir copias.
- Libertad para mejorar el programa y liberar las mejoras, con lo que se beneficia toda la comunidad. Para ello, es necesario disponer del código fuente.

3.1.1. Necesitamos los análisis para avanzar

A la vista de estas libertades y considerando los avances en el desarrollo de software libre, también podemos intentar justificar la necesidad de utilizar análisis funcionales en otros proyectos distintos a la gestión. Para ello, podemos comenzar por preguntarnos ¿son realmente suficientes las libertades de la licencia GPL para el desarrollo y evolución del software libre? Como veremos seguidamente, es posible que para determinados tipos de aplicaciones, entre las que se encuentran las de gestión, la disponibilidad del código fuente no sea suficiente y necesitemos algo más.

Desde el punto de vista práctico, si consideramos que las aplicaciones modernas están formadas por una gran cantidad de líneas de código que son el fruto de años de evolución y de la colaboración de mucha gente, podemos pensar que el código fuente puede no ser lo suficientemente útil como para conocer el funcionamiento de un programa. Para entender y facilitar la colaboración en grandes proyectos de software, es indispensable disponer de una buena documentación que debería incluir los análisis funcionales y las especificaciones de requisitos.

En la mayoría de los proyectos actuales, no estamos trabajando con una simple utilidad de consola formada unos cientos o miles de líneas de programa, se trata de programas mucho más complejos y hablamos de cientos de miles o de millones de líneas de código. En muchas ocasiones es necesario contar un número suficiente de programadores, pero sin una documentación adecuada, es difícil fragmentar las tareas y avanzar con rapidez.

Además, la exigencia de disponer del código fuente puede que no proporcione toda la libertad que creemos tener. Por ejemplo, que se nos proporcione del código fuente de un programa en un lenguaje de programación que no conocemos, tampoco nos serviría de mucho. De nuevo, disponer de análisis funcionales podría paliar este problema y permitiendo que un colaborador pueda trabajar en un proyecto usando el lenguaje que mejor se adapte a sus gustos o conocimientos. Plataformas de desarrollo avanzadas, como MONO, en las que un ejecutable puede estar formado por código en varios lenguajes, podrían ayudar a evidenciar esta necesidad.

Por los tres motivos anteriores, podemos considerar que para poder desarrollar ciertos tipos de aplicaciones, entre las que se encuentran las de gestión, además del código fuente, necesitamos los análisis funcionales para poder hablar de una auténtica libertad. Es posible que las libertades del software libre acaben teniendo en cuenta los análisis funcionales, pero hasta ese momento podemos intentar aplicarlos cuando lo consideremos conveniente.

3.2. El modelo de desarrollo del software libre

Al margen de las posibles limitaciones que encontramos en las libertades del software libre, también podemos analizar las posibles consecuencias del modelo de desarrollo del software libre. El modelo de desarrollo tradicional del software libre tiene una serie de características únicas que han permitido la creación de un sistema operativo sólido y muy

completo. Desgraciadamente, algunas de estas características, que en su momento fueron positivas, puede que no favorezcan la aparición de ciertos tipos de aplicaciones, entre las que se encuentran las de gestión. Sin embargo, algunos de estos problemas se pueden eliminar o minimizar usando análisis funcionales en el desarrollo. Veamos algunas de las características del modelo de desarrollo tradicional y sus posibles problemas.

- El sistema de desarrollo está basado en modelos evolutivos y en conocimientos nebulares. Es posible que algunas habilidades y conocimientos necesarios para la elaboración de determinados programas no se encuentren en el seno de la comunidad. Los análisis funcionales pueden paliar este problema, abriendo una vía para introducir esos conocimientos en la comunidad.
- El software se crea mediante la colaboración y la cesión de conocimientos, pero en grandes aplicaciones y sin una documentación adecuada, se hace difícil esta colaboración y cesión de conocimientos. De nuevo, los análisis funcionales pueden ser la solución.
- Las herramientas de desarrollo son económicas y tienen una calidad técnica adecuada.
- Hay un gran número de programadores y muy competentes, pero sin una estructura vertebrada. La utilización de análisis funcionales puede ayudar a organizar mejor el trabajo y facilitar su repartición entre los participantes.
- Hay demasiada entropía y caos al decidir lo que se desarrolla y sobre la forma de hacerlo. El uso de análisis funcionales podría hacer atractiva para los programadores la creación de determinadas aplicaciones que se consideran necesarias.
- Los programadores suelen trabajar a tiempo parcial y puede que no reciban una remuneración por su trabajo. La utilización de análisis funcionales ayudaría a unir la oferta con la demanda, facilitando la apertura de canales de remuneración para los programadores.

3.3. Otros problemas que se han detectado

Ha habido otras personas que han llegado a la conclusión de que son necesarias las aplicaciones de gestión de software libre. En la historia reciente del software libre se han producido intentos para la creación de aplicaciones de gestión, principalmente para empresas, que no se han acabado materializando en programas. Por ello, hemos considerado interesante analizar las posibles causas que han motivado algunos de estos fracasos y las conclusiones de este análisis son las siguientes:

- El programador se debe encargar del ciclo completo del desarrollo.
- El software aparece como resultado de una serie de casualidades, por lo que no siempre se desarrolla lo que hace falta.

- Problemas de comunicación y mantenimiento en los proyectos grandes.
- Cuando hay problemas de continuidad, se trabaja a tiempo parcial y no existe una documentación adecuada, el proyecto puede desaparecer.
- Los programadores tienen pocos alicientes económicos.
- Cuando se logran aplicaciones suelen aparecer problemas de compatibilidad o de eficacia por carencias en los análisis.
- El modelo evolutivo es poco útil para las aplicaciones de gestión en las que se requieren resultados óptimos desde el principio.
- Faltan herramientas y librerías especialmente diseñadas para la creación de aplicaciones de gestión y para la migración de datos.

4. El efecto de la presión que se está ejerciendo

Ahora que hemos justificado la necesidad de los análisis funcionales desde varios puntos de vista, vamos a analizar lo que está pasando con la presión que se ejerce en favor del software libre. Vistas las ventajas enumeradas anteriormente, no cabe duda de que el software libre es algo muy conveniente para las empresas y la Administración, lo que nos mueve a intentar universalizar su uso. Un caso cercano lo tenemos en HispaLinux, que a través de las campañas de sus socios, intenta implantar este software en la Administración y en las empresas.

4.1. Lo que está ocurriendo en realidad

Por una parte tenemos una carencia de aplicaciones de gestión de general para la Administración y las empresas y al mismo tiempo, se está ejerciendo una gran presión para lograr la implantación del software libre en estos ámbitos. Como se dispone de argumentos sólidos, buenos y compartidos, esta presión está teniendo una respuesta muy positiva en la Administración y las empresas. Tanto es así, que la Administración está comenzando a establecer iniciativas, basadas en software libre, con fechas límite relativamente cercanas. Por ejemplo, en la pasada cumbre de Sevilla se estableció el calendario siguiente:

Calendario europeo

- Establecimiento del Grupo Operativo de Seguridad (2003).
- Red para el fomento del negocio electrónico para las pymes (2004).
- Solución interoperable de firma electrónica (2004).

- Marco europeo de servicios interoperables de administración electrónica, para ciudadanos y empresas (2004).
- Programa europeo de e-learning (2004-2006).
- Tarjeta sanitaria electrónica común (2003).
- Extensión y abaratamiento de las redes de banda ancha en toda Europa (2005).

Por las ventajas que ello supone, sería deseable que todas o la mayoría de estas iniciativas se materializasen en las fechas previstas usando software libre. Desgraciadamente, salvo algunas excepciones, es tradicional que la comunidad de software libre desarrolle las aplicaciones que necesita internamente, las que mejor conoce o las que más apetecen a sus autores. Por ello, podemos intentar buscar una fórmula para que se desarrollen aquellas aplicaciones que realmente se necesitan.

4.1.1. Los logros reales y los efectos no deseados

Por sus notables características, el software libre se está implantando con fuerza en el mercado de los servidores, mercado en el que es casi un líder absoluto. Del mismo modo, más lentamente y gracias en las mejoras en los escritorios gráficos, en las herramientas de configuración y en las suites ofimáticas, también está teniendo cierto éxito en las estaciones de trabajo ofimáticas. Sin embargo, tiene muchas carencias en las aplicaciones de gestión de uso general y en las utilidades que permitan la migración de datos procedentes de aplicaciones propietarias. Sin embargo, cuando se ejerce presión a favor de su uso y no hay aplicaciones para satisfacer la demanda generada, se producen los problemas siguientes:

- Se están generalizando las soluciones parciales o mixtas, con servidores basados en software libre y clientes propietarios.
- Se están demostrando carencias que pueden llevar a pensar que el software libre no es una solución tan buena como se pretende.
- Se están cerrando nichos de mercado que luego será muy difícil recuperar para el software libre.
- La falta de "credibilidad" que se genera por la ausencia de aplicaciones, no favorece la aparición de soluciones integrales hardware+software+servicios.
- Ante la ausencia de soluciones de software libre, algunas empresas se están creando soluciones propietarias que funcionan sobre plataformas de software libre, cerrando nichos de mercado al software libre.
- Se favorecen las soluciones basadas en software a medida, en detrimento de las soluciones "pret a porter".

Por todo lo anterior y antes de que la situación se agrave más, se considera necesario tomar algunas medidas que favorezcan la aparición de las aplicaciones de gestión que se necesitan.

5. La solución propuesta por HispaLinux

Ante los problemas enumerados anteriormente y las ventajas que podrían suponer la utilización de análisis funcionales en el desarrollo, HispaLinux propone utilizarlos como herramienta fundamental para el desarrollo de aplicaciones de gestión para la Administración y las empresas. Pero además de adoptar el uso de los análisis funcionales, también es necesario salir al paso de otras necesidades y problemas que hemos visto a lo largo del estudio. Por ejemplo, también será necesario capacitar a los programadores para que puedan utilizar e interpretar esos análisis funcionales, por lo que la solución a desarrollar es más compleja que la simple propuesta de usar los análisis funcionales.

5.1. Lo que permitiría el uso de análisis funcionales

Durante el estudio hemos visto que los análisis funcionales pueden solucionar o paliar algunos de los problemas que se producen cuando queremos desarrollar aplicaciones de gestión para la Administración y las empresas. Dada la importancia que tienen en el proyecto, es conveniente que hagamos un pequeño resumen de las ventajas del uso de los análisis, en relación con el software libre. La utilización de análisis funcionales nos permite:

- La entrada de otros actores en el ciclo de desarrollo.
- La fragmentación de grandes proyectos.
- Independencia del lenguaje y una mejor comprensión del código fuente.
- La migración de las aplicaciones actuales a software libre.
- Mayor continuidad en el desarrollo y la recuperación de proyectos.
- Unir la demanda de aplicaciones con la programación de las mismas.
- Facilitar y posibilitar la remuneración de los trabajos de programación.
- Una mayor estandarización y la garantía de comunicación entre módulos y aplicaciones.
- Una imagen más creíble y una metodología de trabajo más rigurosa.

5.2. Materialización del proyecto Gestión Libre.

Como hemos dicho anteriormente, la materialización del proyecto Gestión Libre es algo más complejo que recomendar a los desarrolladores el uso de los análisis. De hecho, estos análisis deberían ser proporcionados por la Administración y las empresas interesadas en la creación de estos programas. Lo que se intenta con el proyecto, es que se convierta en un punto de encuentro de la Administración y las empresas con la comunidad de software libre. Al mismo tiempo, se intenta que el proyecto contenga todo aquello que pueda necesitar el programador a la hora de desarrollar aplicaciones de gestión, incluida su formación integral. Con ello se intenta que sea atractivo para la comunidad de software libre la creación de las aplicaciones de gestión que demandan la Administración y las empresas. El proyecto se materializa en una base de datos de conocimiento con los contenidos siguientes:

- Herramientas de desarrollo adecuadas.
- Información útil para los programadores, junto con la normativa y la legislación aplicable.
- Documentación sobre temas de calidad, análisis, desarrollo, etc..
- Contenidos de formación específicos (e-learning).
- Criterios de desarrollo, especificaciones de requisitos y análisis funcionales libres de aplicaciones de gestión.

5.3. El proyecto Gestión Libre es una revolución

Es posible que el proyecto Gestión Libre represente una revolución dentro de la comunidad de software libre y entre los motivos que avalan esta afirmación se encuentran los siguientes:

- Logra la participación de otros actores, como la Administración y las empresas, para que colaboren en el desarrollo del software libre que necesitan.
- Tiene como objetivo la creación de software de calidad, que se pueda homologar en base a la norma ISO 9000/2000 u otras normas de calidad que se consideren interesantes.
- Pretende crear aplicaciones de software, basadas en la arquitectura cliente/servidor, que sean escalables y personalizables. El cliente no se adaptará a la gestión y en su lugar, la gestión se adaptará al cliente, algo casi imposible con el software propietario.
- Se basa en que los análisis funcionales sigan un proceso de desarrollo similar al del código fuente. Para lograrlo se usarán herramientas de modelado avanzado que generen código UML o compatible con Métrica 3, que es el estándar de la Administración.

- Explora las metodologías y las herramientas de desarrollo de aplicaciones más novedosas y potentes.
- Coloca en segundo plano el desarrollo nebuloso y darwinista tradicional del software libre, para centrarse en un modelo más formal y eficiente.

5.4. Lo que permite el proyecto Gestión Libre

El proyecto Gestión Libre, al basarse en los análisis funcionales, hereda todas las ventajas y posibilidades de estos, pero además, nos permite otras muchas más cosas interesantes, como por ejemplo:

- Hace atractivo, para la comunidad de software libre, el desarrollo de las aplicaciones que necesitan la Administración y las empresas. Ayudando a comunicar a los programadores aquello que demandan la Administración y las empresas.
- Permite el desarrollo de aplicaciones de alta calidad, modulares y escalables.
- Favorece el desarrollo de software "pret a porter" competitivo y económico.
- Permite definir herramientas y los criterios de desarrollo comunes.
- Facilita el trabajo, reduciendo los tiempos y los costes.
- Desvincula parte del desarrollo del lenguaje de programación.
- Activa la economía en torno al software libre.
- Favorece a programadores, estudiantes, Administración y empresas.
- Permite la migración de aplicaciones de gestión al software libre.
- Facilita la asignación económica a los desarrollos, uniendo la oferta de programación con la demanda de aplicaciones.
- Armoniza la presión para la implantación del software libre con la oferta real de aplicaciones.

5.5. Líneas de trabajo abiertas en la actualidad

El proyecto Gestión Libre es muy ambicioso y extenso, por lo que necesita mantener varias líneas de trabajo simultáneas. Para conseguir los objetivos en cada una de estas líneas de trabajo, es imprescindible delegar en coordinadores de área, que dependiendo del coordinador general, se encargan de los distintos aspectos del proyecto. Las líneas de trabajo que hay abiertas en la actualidad son:

- Definición y mantenimiento de los fundamentos y objetivos del proyecto. El proyecto se desarrolla como un sistema vivo y en constante evolución mediante la colaboración en la lista de correo y en la web inversa de HispaLinux (Wiki).
- Creación y mantenimiento de fondos documentales:
 - ISO 9000. Se están definiendo los procesos y se está estudiando su compatibilidad con el modelo de desarrollo y las herramientas propuestas. La implantación de la ISO 9000 se debe a la necesidad de crear aplicaciones de gestión de calidad. El proyecto no certifica las aplicaciones, solamente entregará la documentación necesaria para permitir, si se desea, una certificación posterior. Este concepto, que es pionero en el mundo del software libre, puede llegar ser un argumento muy positivo a la hora de implantar el software en la Administración o las empresas.
 - Análisis funcionales y herramientas asociadas (UML, dia, dia2code, umlmodeller, Métrica 3, etc.). En este momento se está comenzando a documentar algunas herramientas libres como dia, dia2code, umlmodeller y se están evaluando otras. También se está trabajando en la interpretación de análisis UML y en Métrica 3. Para facilitar aún más el trabajo y normalizar criterios, también se está elaborando un libro de estilo de UML que permitirá acercar aún más el código UML al código fuente, introduciendo reglas más estrictas.
 - Programación extrema. Aquí se están evaluando y documentando algunas prácticas que pueden ser compatibles con el modelo de desarrollo propuesto.
 - Herramientas de programación (PostgreSQL, libgda, XML, etc.).
 - Normativa y legislación aplicable a la gestión de la Administración y las empresas.
- Plan de actividades en curso dentro del proyecto. En este plan de actividades destaca la participación de Gestión Libre e HispaLinux, junto con la Universidad de León, en el Observatorio de Voto Electrónico (OVE). El objetivo del OVE es explorar y fomentar la creación de un sistema de voto electrónico basado en software libre.
- Repositorio de análisis funcionales y de especificaciones de requisitos para la Administración y las Empresas.

- Sistema de e-learning (enseñanza distribuida) orientado a la formación de analistas y programadores de gestión. La idea es la de lograr la capacitación plena de los programadores y una de las metas propuestas, es la creación de un master universitario orientado al desarrollo de aplicaciones de gestión con software libre. El motivo de crear este grupo de trabajo se deriva de la necesidad de formación de los programadores, lo que entendemos que es fundamental para el proyecto. Para ello y en colaboración con la UNED, intentamos crear unos contenidos educativos específicos y muy pragmáticos, orientados a las herramientas propuestas en el proyecto. Los objetivos de este grupo de trabajo son:
 - Documentar las herramientas seleccionadas para facilitar su utilización por los programadores.
 - Seleccionar y adaptar la información que sea relevante desde el punto de vista práctico.
 - Organizar seminarios, talleres y conferencias sobre el uso de las herramientas propuestas en el proyecto Gestión Libre.
 - Suplir las carencias de la universidad a la hora de elaborar contenidos y cursos orientados al software libre, usando para ello, sistemas de enseñanza distribuida e-learning.

5.6. Relación entre los análisis funcionales y los programadores

Para garantizar que el uso de los análisis funcionales no sea un problema para los programadores, se intentará formar a los programadores en el uso de las tecnologías UML y Métrica 3. Para ello, se intentará dotar al proyecto de los manuales y de los contenidos educativos adecuados. Esta formación incidirá tanto en la teoría, como en la práctica y se basará en herramientas libres disponibles para la realización de estos análisis funcionales.

Aunque algunos de los análisis se podrían mantener elaborar, en el seno de la comunidad de software libre, la intención es que estos análisis sean aportados por la Administración y las empresas, usando repositorios organizados para ello. Para garantizar la calidad de los análisis, estos seguirán un ciclo de desarrollo similar al del código fuente. Es decir, se liberará una versión estable que será actualizada por un equipo de mantenedores y que lo tratarán como código fuente. Estos análisis, que serán libres, en virtud de las libertades 2 y 4 de la licencia GPL que hemos visto anteriormente, quedarán ligados al software que se genere a partir de ellos. El uso de estas licencias con su efecto "vírico", pueden suponer una ventaja estratégica y una garantía adicional, cuando se aplican a los análisis funcionales como paso previo a la codificación de los programas.

Dado que HispaLinux es una asociación sin ánimo de lucro, los equipos de desarrollo, que pueden ser particulares o empresas de software libre, deberán ser ajenos al proyecto. La idea de Gestión Libre, no es la de realizar aplicaciones de gestión, es la de facilitar y hacer que sea atractivo su desarrollo a la comunidad de software libre. No obstante, se mantendrá una estrecha colaboración entre los equipos de mantenimiento de los análisis funcionales y los programadores que los utilicen con la intención de facilitar y dinamizar el desarrollo. A pesar de lo dicho anteriormente, si se considerase necesario desde el proyecto Gestión Libre, se podrían crear o adaptar herramientas de software específicas para la creación de aplicaciones de gestión.