

Los desarrolladores profesionales y el software libre El reto de adaptarse a la nueva realidad



Ignacio Correas Usón
Consultor tecnológico y profesor de Informática. Universidad de Zaragoza

El 10 de mayo de 1940, Alemania comenzó la ofensiva *Fall Gelb* en la que la *Wehrmacht* - ejército alemán - tenía como objetivo conseguir el control de la costa del Canal de la Mancha. Cuarenta días más tarde Francia capitulaba mientras que Gran Bretaña había perdido gran parte del equipamiento de su ejército en las playas de Dunquerque y se encontraba con escasa capacidad de defender las islas frente a una probable invasión. Resulta curioso analizar el cómo la nación perdedora de la Gran Guerra y obligada a pagar compensaciones desorbitadas a sus adversarios, sin posibilidad de rehacer su ejército hasta bien entrada la década de los treinta, casi sin recursos naturales y con escasas reservas de divisas extranjeras para la adquisición de materias primas consiguió una victoria tan aplastante contra los dos imperios más importantes de la época, siendo que éstos habían tenido condiciones más favorables durante los veinte años anteriores. La razón no está en el acceso a los avances tecnológicos, puesto que los grandes cambios de la guerra anterior estaban disponibles para las tres potencias de una forma

similar, sino en cómo se adaptaron a estos cambios. Mientras que los aliados se limitaron a integrar las nuevas armas en sus tácticas tradicionales, los alemanes idearon la *Blitzkrieg* - guerra relámpago -, una nueva táctica conceptualmente distinta e innovadora que consistía en la coordinación de la infantería, los tanques, la artillería y los aviones, concepto denominado hoy en día "armas combinadas" y que se siguió utilizando hasta la primera Guerra del Golfo.

La historia está llena de lecciones que nos muestran lo esencial de adaptarse a los cambios y que la diferencia entre el éxito y el fracaso radica no en la adopción de los avances científico-tecnológicos sino en adecuar la forma de pensar y las estructuras a las nuevas reglas del juego que se establecen. En este aspecto la industria informática está viviendo un momento crítico, ya que la aparición del software libre implica un cambio esencial en la propia noción de software. Desde los años 70 se ha considerado que los programas eran productos que una empresa desarrollaba y distribuía cobrando licencias por su uso e instalación. Sin

embargo, el movimiento del software libre propone un concepto radicalmente distinto: el software es un bien común a disposición de todos para su uso, distribución y modificación. Por lo tanto, el negocio ya no puede ir orientado a vender licencias de un programa y promover su comercialización sino que el trabajo debe encaminarse hacia ofrecer servicios de mayor valor añadido. En este sentido el software libre significa una revolución, puesto que cambia el modelo de negocio en que se basa la mayor parte de la industria informática. A la vez, el salto hacia una industria de servicios sigue también una evolución natural del propio mercado. Cada vez los usuarios piden más soluciones completas sobre formatos estándar en lugar de productos aislados, principalmente por la integración de las tecnologías web en los lugares y métodos de trabajo, y exigen mayor calidad (ver el artículo <http://www.thesmalls.com/press/WhySoftwaresSoBad.pdf>). Sobre todo esta evolución está siendo natural porque Internet ha hecho que consideremos la información como algo gratuito y al alcance de todos y ha ido formando una cultura global de

cooperación. El software libre forma parte de un movimiento más extenso que afecta a todas aquellas industrias basadas en la propiedad intelectual, como el hardware (www.opencores.org), la música, las artes (www.creativecommons.org), el periodismo (www.doaj.org) o hasta la industria farmacéutica.

Aprovechar los desarrollos existentes

Este cambio fundamental en la industria informática afecta con especial relevancia a los desarrolladores y en un mercado cada vez más dinámico y competitivo es necesario adaptarse con rapidez a la nueva situación antes de quedarse fuera. En primer lugar, un desarrollador no debería confiar en ganarse la vida vendiendo licencias de productos informáticos, puesto que con alta probabilidad se disponen en la red otros libres que realizan funciones similares y es sólo cuestión de tiempo que, a largo plazo, los programas de código abierto dominen el mercado. Sin embargo, lo que sí se pueden ofrecer son soluciones avanzadas y personalizadas partiendo de una o varias aplicaciones libres existentes (<http://freshmeat.net>). Es elemental que resulta más rápido y eficaz el adaptar y mejorar una solución ya existente, probada, depurada y optimizada que comenzar desde cero. La ciencia en general es un ejemplo de esto, basando su avance en descubrimientos anteriores. Es famosa la frase de Newton para expresar que había logrado descubrir las leyes fundamentales de física gracias a avances de otros, "camino a lomos de gigantes". Se trata de aprovechar el trabajo realizado por otros para no inventar la rueda y dedicar el esfuerzo a aportar un valor añadido a la aplicación en la que se basa. Evidentemente, no se debe

escoger cualquier programa libre de partida, sino que su selección conviene efectuarse cuidadosamente. La mejor garantía sobre la calidad de una aplicación libre es que tenga detrás una amplia comunidad activa de usuarios y desarrolladores. Una vez terminado un desarrollo basado en una herramienta libre es conveniente hacer lo posible para que las modificaciones realizadas se integren en la versión oficial del producto. De esta manera se facilitará considerablemente el mantenimiento del sistema instalado ya que las nuevas versiones oficiales con otras mejoras serán directamente aplicables. Los beneficios de ser co-desarrollador de un proyecto libre son numerosos. Además de mejorar una herramienta conocida y de extender su uso el desarrollador aumenta su prestigio profesional y sobre todo gana una experiencia profunda de la aplicación empleada.

Los desarrolladores que utilicen aplicaciones libres deben tener un conocimiento particularmente sólido de los sistemas con los que trabajan. Por un lado, los desarrollos se basan y se integran en aplicaciones existentes que hay que conocer previamente. Por otro, los programas libres suelen ofrecer unas posibilidades de configuración mucho más variadas que sus homólogos propietarios con lo que es recomendable analizar la potencia de un sistema libre antes de decidirse a comenzar un desarrollo largo. Hay que tener en cuenta, además, que los especialistas de sistemas libre, al tener a su disposición todos los entresijos del sistema, disponen de un nivel de conocimientos mejor que sus análogos en herramientas propietarias, con lo que, si se quiere destacar, el esfuerzo en formación debe ser aún mayor. A la larga este esfuerzo añadido se ve compensado con una mejora competitiva en calidad

y eficiencia. Por suerte, existe una ingente cantidad de documentación disponible en la red y orientada a aplicaciones prácticas (por ejemplo, en www.tldp.org). También es sencillo mantenerse al día de la evolución del software libre, como nuevas versiones, nuevos programas o tendencias, gracias a los numerosos sitios de noticias y listas de correo que existen en la red (como <http://linuxtoday.com/> o <http://lwn.net>). Finalmente, hay en Internet muchos foros de expertos donde se puede solicitar ayuda o consejo para cualquier problema técnico que se encuentre y en donde, por lo general, se reciben rápidamente respuestas de calidad (por ejemplo en www.linuxforum.com/forums).

Asumir los objetivos del cliente

Quizás lo más difícil para un desarrollador en la adaptación a la filosofía del software libre esté en el hecho de que el cliente contrata la prestación de un servicio y no el desarrollo de un producto. Muchos contratos de desarrollo imponen unas limitaciones sobre el derecho de uso del programa, incluso limitado a un número reducido de instalaciones. Asimismo, los contratos de mantenimiento y evolución del sistema se hacen con las condiciones que impone el desarrollador, ya que éste es el único que tiene el código. Considerando que los costes de mantenimiento suelen estar entre el 100 y el 150 % del coste inicial, suponen un negocio aún mayor que el propio desarrollo del programa. No obstante, a largo plazo el cliente quedará insatisfecho y reticente a contratar un nuevo desarrollo a medida o a mantener la evolución de su sistema, con lo que estas condiciones resultan perjudiciales para ambas partes. Con el software libre los contratos no pueden incluir esta

Software libre

trampa oculta debido a que el código es público y se acuerda el servicio de desarrollarlo. Una vez creado un sistema el cliente es libre de elegir al proveedor para su mantenimiento, aunque lo lógico es que estos contratos vayan al que mejor conozca el sistema, es decir, al desarrollador.

Además, los clientes exigen soluciones completas, no productos aislados de los que tengan que ocuparse una vez adquiridos. Hace ya tiempo que se habla del TCO (Total Cost of Ownership), es decir, el conjunto de costes que incluyen la adquisición o desarrollo, instalación, adopción y mantenimiento de un sistema informático. El desarrollo de software exige que el trabajo no se limite a un producto sino que se planifique su instalación y mantenimiento y una formación continua que minimice el TCO. Este objetivo sólo es posible si desde el principio el desarrollo se orienta a ofrecer un servicio integral.

En realidad, los dos puntos anteriores se pueden resumir diciendo que es necesario una integración total entre cliente y desarrollador de forma que los objetivos del primero queden asumidos por el segundo.

Personalmente he tenido la oportunidad de vivir esta situación laboral muy de cerca en Finlandia, un país que está considerado uno de los tecnológicamente más avanzados del mundo. Pero en mi opinión su mayor ventaja está en aspectos que pertenecen a la cultura del trabajo. Concretamente, los temas culturales en los que más destacan son el compromiso con la calidad y la integración entre cliente y proveedor. Centrándonos en éste último, la mayor parte de las grandes y medianas empresas de Finlandia han llevado el concepto de especialización y subcontratación hasta el extremo. Con frecuencia los proveedores tienen su lugar de trabajo en la empresa



contratante, asumen los objetivos del cliente como propios e incluso se realizan proyectos de desarrollo en los que no interviene nadie de la empresa cliente. Los que, como yo, han trabajado años en un país nórdico subcontratados por una gran empresa habrán vivido momentos en los que defendían antes los objetivos de la empresa cliente que los de la propia.

La competencia puede ser tu mejor aliado

La principal ventaja competitiva en un mercado basado en la oferta de servicios radica en tener un conocimiento y habilidades destacados. Si, como en el caso del software libre, el esfuerzo necesario para adquirir un importante saber-hacer es elevado, resulta indispensable especializarse en un área concreta. Sin embargo, así no es posible ofrecer soluciones completas a clientes potenciales. Por ello, la posición más fuerte la consigue aquél que, aun especializándose en un sector determinado de la informática pueda crear una red de alianzas eficaz de profesionales y empresas especializados a su vez en otras áreas complementarias. En realidad, la

diferencia entre un competidor y un colaborador es muy pequeña y la cooperación puede ser una base muy sólida para poder ofrecer un amplio abanico de servicios y soluciones integrales. De hecho, el software libre se ha desarrollado de esta manera, mediante la cooperación de especialistas independientes con un objetivo común.

Conclusión

En los próximos años vamos a asistir a grandes cambios en la industria informática debido a la adaptación natural del mercado a una forma radicalmente opuesta de concebir el negocio. El software propietario podrá coexistir con el libre o incluso ser dominante en algunos segmentos del mercado, pero la forma tradicional de vender programas informáticos va a cambiar. Todas las grandes compañías informáticas han reaccionado frente al software libre, algunas haciéndolo parte esencial de su estrategia (como Novell, IBM o HP), otras, como Oracle, simplemente adaptándose a la nueva tecnología y sólo una, Microsoft, prefiriendo un choque frontal contra el movimiento libre. El futuro dirá quién está aprovechando mejor el momento.