

# Software Livre X Mundo Corporativo: Confronto ou Cooperação?

*Felipe Buzatti Nascimento, Leonardo Vilela Teixeira, Lucas Alves Guerra França, Rafael Almeida Batista*

**Abstract** É comum associarmos o software livre e código aberto com ideologias distantes do mundo corporativo, das grandes empresas. Ainda assim, por mais incoerente que isso possa parecer em um primeiro momento, há vários exemplos de empresas que investem em software livre. O nosso objetivo é explorar esse tema e entender melhor qual o papel do software livre no mundo corporativo e vice-versa, a fim de melhor compreender as vantagens para cada uma das partes desse relacionamento, aparentemente ilógico, mas não incomum.

## 1 Introdução

Muitos projetos de sucesso na área do software livre têm uma ligação com empresas, tais como o MySQL, o PostgreSQL, o CUPS (mantido pela Apple), ou o Android (mantido pelo Google). Então não há dúvidas de que software livre e empresas são termos miscíveis.

Muitas companhias floresceram na onda do software *open source*, que passou a crescer a partir da década de noventa, com o surgimento de produtos como o Linux. Essas são empresas que escolheram liberar seus produtos sob uma licença open-source ao invés de uma licença proprietária, como se costumava fazer.

Entre essas empresas podemos citar: Liferay Inc., Red Hat, MySQL AB e JBoss. Tais companhias oferecem produtos com o código aberto enquanto produzem receita com manutenção, suporte e serviços. ([http://en.wikipedia.org/wiki/Professional\\_open-source](http://en.wikipedia.org/wiki/Professional_open-source))

## 2 Como viver de software livre

Empresas, pelo menos na definição com que trabalhamos aqui, e dado o sistema econômico em que vivemos, têm como objetivo obter lucro. Desse modo, é preciso descobrir como uma empresa pode obter lucro com o software livre.

À primeira vista, a resposta é simplesmente que a empresa não pode obter lucros. Se o software é livre e eu passo a vendê-lo por um preço  $x$ , uma pessoa pode simplesmente comprá-lo de mim, e revender quantas cópias quiser a um preço  $y$ , menor que  $x$ , e quem compra por  $y$ , poderá fazer o

mesmo, e assim por diante. Até que o preço atinge zero, e aí não faz mais sentido “vender” o software.

Mas isso não significa que não é possível ganhar dinheiro vendendo software livre. Isso porque podemos pensar na palavra *vender* de muitas formas. Ou seja, pode haver mais na venda de um software do que repassar cópias. Para ficar mais claro, vamos enumerar alguns dos possíveis caminhos que uma empresa de software livre pode tomar para fazer negócios:

- **Serviço de suporte:** Se uma empresa desenvolve um produto, mesmo que esse seja liberado como software livre, e ainda que conte com contribuições da comunidade, ela tende a ter um conhecimento do software com profundidade. Então sua posição é a ideal para prover um serviço de suporte. Nesse modelo esse é o seu produto, o suporte. A vantagem é que não há nenhuma restrição de liberdade aqui. Inicialmente, qualquer empresa poderia prestar o suporte. Mas a empresa que estava desenvolvendo tem maior conhecimento, e por isso mais capacidade para oferecer esse serviço. Empresas como a Red Hat e MySQL utilizam essa modalidade de negócio.
- **Múltiplas licenças:** Outro modelo consiste em vender o mesmo software sob diferentes licenças. Um exemplo, a biblioteca Qt fornece seu código sob uma licença LGPL, que permite que essa biblioteca seja usada para produzir software livre ou proprietário. Porém, essa licença exige que alterações na biblioteca sejam também publicadas sob a licença LGPL. O usuário que queira realizar alterações proprietárias na biblioteca deve comprar uma licença comercial. Esse modelo tem alguma semelhança com o anterior.
- **Versionamento duplo:** A idéia aqui é que existam uma versão livre e uma versão proprietária. A comunidade contribui para a versão livre, que normalmente é mais atualizada. Mas a empresa usa as características desenvolvidas na versão livre, reimplementadas, em sua versão paga, garantindo um menor número de erros através de um número maior de testes e inspeções. Um exemplo desse modelo é o utilizado pela SugarCRM, que produz uma solução de CRM em ambiente web.
- **Serviço de customização:** Mesmo havendo uma solução para um problema, muitas vezes é preciso um bocado de trabalho para que a solução fique aceitável para o usuário. Muitas vezes o usuário está disposto a pagar por essa personalização. Aí entram as empresas responsáveis por esse serviço, que são, em geral, as mesmas empresas que desenvolvem as soluções iniciais, e que aqui, como no caso do suporte, são mais capazes de realizar o serviço. Empresas que trabalham segundo esse modelo incluem Drupal e Joomla, dois frameworks de gestão de conteúdo utilizados por inúmeras empresas.
- **Serviços de hospedagem e manutenção:** Um usuário, mesmo possuindo a aplicação sob um regime livre, pode decidir que é mais simples pagar a uma empresa para que ela gerencie a aplicação em suas próprias máquinas, acessíveis ao cliente pela web, arcando com a gerência do hardware e o suporte ao software. Esse serviço pode ser tanto prestado pela própria empresa que desenvolve a aplicação quando por terceiros. Um exemplo desse tipo de aplicação é o serviço de teleconferência DimDim.

Esses são apenas alguns exemplos de modelos de negócios para o software livre, mas são o suficiente para mostrar que é possível ganhar dinheiro com software livre. Para uma prova numérica, basta lembrar que a venda do MySQL para a Sun envolveu bilhões de dólares.

Não é difícil encontrar casos de sucesso de empresas que seguem o modelo de negócios explicado acima: temos, por exemplo, o caso da Red Hat, que baseia seu faturamento na venda de suporte para sua distribuição Linux, ao mesmo tempo que incentiva o uso da plataforma para promover seus serviços e produtos como efeito colateral.

### **3 Empresas que contribuem com projetos de Software Livre**

Na seção anterior, discutimos casos em que o desenvolvimento de certo produto *open source* é tido como atividade principal. Um outro modelo de interação entre empresas e projetos de software livre é, obviamente, o de consumidor: certo software livre é necessário para a atividade da empresa e por isso é utilizado por ela.

Por exemplo, o ramo de infraestrutura computacional de servidores é um onde o software livre marca presença forte. Afinal, o próprio Unix, base do Linux, utilizado em grande parte dos servidores, já completou seu quadragésimo aniversário. E a existência da liberdade de alteração em componentes muitas vezes críticos para uma empresa produz um efeito interessante: muitas vezes, a opção de uma empresa patrocinar uma alteração específica em determinado software é comercialmente atraente.

Não é difícil perceber que ideologicamente essa idéia faz muito sentido: um software com boas idéias amadurece com a injeção de esforço patrocinada por uma empresa que o utiliza, o que faz esse software atrair mais usuários, aumentando a quantidade e qualidade de contribuições.

Embora - devido ao ruído envolvido nos dados - uma análise que comprovasse isso empiricamente seja talvez impraticável, não é difícil de encontrar exemplos de projetos que são 'adotados' por empresas diretamente interessadas em seu amadurecimento: os casos mais famosos são o Google, que contribui com uma miríade de projetos de software livre, muitas vezes empregando seus criadores, como aconteceu com o criador da linguagem Python, Guido van Rossum; o Facebook, que chegou a criar uma versão do PHP otimizada para alto desempenho para uso interno, e o Twitter, que recentemente adotou o projeto Cassandra, um banco de dados não-relacional que foi o escolhido para ser o responsável pelo armazenamento da empresa.

Uma outra modalidade de cooperação interessante entre empresas e a comunidade de software livre é a de doações. Podemos citar o exemplo do Google Summer of Code, um evento global que acontece no verão do hemisfério norte - ou seja, quando as faculdades americanas estão em período de férias - que oferece remuneração atraente a graduandos de áreas relacionadas a computação para contribuir em projetos open-source. Não é difícil perceber que ações como essa possuem efeito altamente catalisador no sentido de disseminar a cultura livre entre pessoas aptas a fazer contribuições significativas com a comunidade.

### **4 Software livre ou proprietário?**

Várias empresas, por motivos diversos, acabam tomando um caminho contrário ao software livre, adotando e/ou criando software proprietário. O atual estado do mercado de sistemas operacionais para celulares mostra que este nem sempre é um bom caminho, principalmente se a empresa estiver em busca da liderança de mercado. O sistema operacional de código fechado iOS, da Apple, antigo líder de mercado, já foi superado pela sensação do momento, o software livre Android, do Google, que engole a cada mês uma nova fatia do mercado de smartphones. O fato de todo o hardware que faz uso do iOS vir da própria Apple condiz com a política histórica da empresa, mas acaba deixando os compradores sem opção, o que leva muitos a preferirem o Android, presente em smartphones de várias marcas, como Sony, LG, Samsung, entre outras.

Outra situação recorrente é a ausência de competição em aplicações para algum fim específico, onde simplesmente não há uma opção livre com a mesma qualidade da(s) opção(ões) de código fechado, que domina(m) o mercado. Um exemplo pode ser observado nos software para edição avançada de imagens, onde o Photoshop, da Adobe, lidera com folga há vários anos. A sua adoção é tão massiva que o seu nome já virou um verbo, sendo comum escutar alguém dizer que vai 'photoshopar' uma imagem, como sinônimo de editar a imagem. Não há um competidor à sua altura, que proporcione toda a gama de ferramentas de edição do Photoshop. O que mais se aproxima talvez seja o GIMP, editor de imagens livre, disponível em várias plataformas. Ainda assim, a liderança do Photoshop já dura anos, e ele, como todo software de sucesso, está sempre evoluindo, o que torna difícil a tarefa de alcançá-lo.

Vários outros exemplos semelhantes existem, mostrando que a adoção de um software livre ou proprietário depende de vários fatores, únicos a cada nicho de mercado, que levam empresas e/ou consumidores a tomarem um caminho ou outro.

## 5 Conclusão

A partir dessa análise, podemos perceber como e quais são as mais comuns interações entre o software livre e o mundo corporativo. Uma ideia que inicialmente parece impraticável, mas que, como vimos, possui vários casos de sucesso. Apesar de existirem conflitos, casos em que o software proprietário é mais utilizado, vimos que é possível conceber uma estratégia de negócio onde o software é livre e o seu entorno (suporte, personalização, etc.) é a verdadeira fonte de lucro. Em suma, o software livre não veio como uma forma anárquica de acabar com o mercado capitalista, mas sim como uma outra filosofia que não é contrária ao lucro, mas promove a disseminação de conhecimento em detrimento do individualismo e protecionismo.

## Referências:

ASAY, Matt. **Red hat: the mother of all open source business models** . , 2006. Disponível em: <<http://Red Hat: The mother of all open source business models>>. Acesso em: 03 jun. 2011.

**JOHN, Anoop. Making money with free software. , 2009. Disponível em:**  
<<http://www.thondomraughts.com/2009/03/making-money-with-free-software.html>>. Acesso em:  
03 jun. 2011.

**OSBORNE, Brian. Android continues to overtake iOS in the smartphone market. ,2011.**  
Disponível em: <<http://www.geek.com/articles/mobile/android-continues-to-overtake-ios-in-the-smartphone-market-20110428/>>. Acesso em 06 jun. 2011.

**QUORA. What is a good business model for open source projects? Disponível em:**  
<<http://www.quora.com/Open-Source-Software/What-is-a-good-business-model-for-open-source-projects>>. Acesso em 06 jun. 2011.

**URBAN DICTIONARY. Photoshop, 2011. Disponível em:**  
<<http://www.urbandictionary.com/define.php?term=photoshop>>. Acesso em 06 jun. 2011.

**WIKIPEDIA. Professional open-source. , 2011. Disponível em:**  
<[http://en.wikipedia.org/wiki/Professional\\_open-source](http://en.wikipedia.org/wiki/Professional_open-source)>. Acesso em: 03 jun. 2011.