



RAM. Revista de Administração Mackenzie

ISSN: 1518-6776

revista.adm@mackenzie.com.br

Universidade Presbiteriana Mackenzie

Brasil

ARROYO, CRISTIANE SÔNIA; MONFORTE MERLO, EDGARD; SIMÕES, ANDRÉ XAVIER
A ECONOMIA DO SOFTWARE DE FONTE ABERTA: RAZÕES QUE LEVAM OS
DESENVOLVEDORES DE SOFTWARE A PARTICIPAR DAS COMUNIDADES DE FONTE ABERTA

RAM. Revista de Administração Mackenzie, vol. 5, núm. 1, 2004, pp. 124-143

Universidade Presbiteriana Mackenzie

São Paulo, Brasil

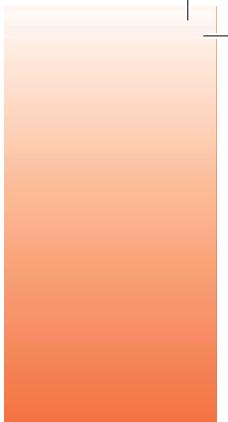
Disponível em: <http://www.redalyc.org/articulo.oa?id=195417779007>

- ▶ Como citar este artigo
- ▶ Número completo
- ▶ Mais artigos
- ▶ Home da revista no Redalyc

redalyc.org

Sistema de Informação Científica

Rede de Revistas Científicas da América Latina, Caribe, Espanha e Portugal
Projeto acadêmico sem fins lucrativos desenvolvido no âmbito da iniciativa Acesso Aberto



A ECONOMIA DO SOFTWARE DE FONTE ABERTA: RAZÕES QUE LEVAM OS DESENVOLVEDORES DE SOFTWARE A PARTICIPAR DAS COMUNIDADES DE FONTE ABERTA

THE ECONOMY OF OPEN SOURCE SOFTWARE: REASONS THAT
LEAD SOFTWARE DEVELOPERS TO PARTICIPATE IN THE OPEN
SOURCE SOFTWARE COMMUNITIES

CRISTIANE SÔNIA ARROYO

Mestre em Ciências pelo Instituto de Ciências Matemáticas e de Computação, São Carlos, Universidade de São Paulo. Doutoranda em Administração pela Faculdade de Economia, Administração e Contabilidade da Universidade de São Paulo.
Av. Prof. Luciano Gualberto, 908 – Cidade Universitária – São Paulo – SP – CEP 05508-900
E-mail: csarroyo@terra.com.br

EDGARD MONFORTE MERLO

Doutor em Administração pela Faculdade de Economia, Administração e Contabilidade da Universidade de São Paulo. Professor da Faculdade de Economia, Administração e Contabilidade da Universidade de São Paulo, Campus de Ribeirão Preto.
Av. Bandeirantes, 3.900 – sala C 36 – Cidade Universitária – Ribeirão Preto – SP – CEP 14040-900
E-mail: edgardmm@usp.br

ANDRÉ XAVIER SIMÕES

Graduado em Administração pela Faculdade de Economia, Administração e Contabilidade da Universidade de São Paulo, Campus de Ribeirão Preto.
Av. Bandeirantes, 3.900 – sala C 36 – Cidade Universitária – Ribeirão Preto – SP – CEP 14040-900
E-mail: andre-simoes@uol.com.br





RESUMO

Atualmente o desenvolvimento de *software* é uma das atividades econômicas de maior evidência no mundo dos negócios internacionais. Grandes fortunas e corporações, como a Microsoft, se fizeram com o desenvolvimento e a venda de *software* proprietário. Nos últimos anos, contudo, tem ocorrido grande aumento do interesse no desenvolvimento do *software* livre, pois a expansão de seu uso o levou a ser competitivo e até dominante em algumas áreas. Do ponto de vista econômico, o comportamento dos programadores e das companhias comerciais empenhadas em projetos de fonte aberta é intenso. Por que programadores e grandes empresas investiriam tempo e dinheiro em um projeto que não poderiam vender, para obter retornos financeiros imediatos, nem poderiam registrar direitos de propriedade sobre seu trabalho? O objetivo deste artigo foi buscar e discutir as razões que levam empresas e profissionais a participar das comunidades de *software* de fonte aberta. Além disso, estabeleceu-se, por meio do estudo de caso das empresas Red Hat e Grisoft, a diferença entre *software* livre e gratuito.

PALAVRAS-CHAVE

Software livre; Comunidade de fonte aberta; Economia de contribuição.

ABSTRACT

Currently the software development is one of the most lucrative economic activities. Great wealth and large corporations, as the Microsoft, were built through the proprietary software development and commercialization. In the last years it occurred a great increase of the free software development, therefore the recent and strong expansion of its use took to be competitive and until being dominant in some areas. Evaluating this process by the economic point of view, the individual behavior of the commercial programmers and effort of company to develop open source projects has been intensive. Why programmers and large companies would invest time and money in a project that could not give an immediate financial return and even not provide a register right of property? The objective of this work is the investigation and discussion of the reasons that lead companies and people to participate in the open source software communities.

KEYWORDS

Free software; Open source community; Contribution economy.

1 INTRODUÇÃO

A globalização da atividade econômica encontra um de seus maiores exemplos na indústria de *software*, onde ocorre o desenvolvimento de tecnologias que rapidamente passam a ter adoção e disseminação em escala mundial.

A indústria de *software* tem como um de seus pilares o desenvolvimento de produtos baseado em conhecimento. Entretanto, esse ramo de atividades pauta-se pela comercialização de programas que possuem códigos de programação que não podem ser alterados pelos usuários finais (*software* proprietário).

Na década de 1960 começou um movimento centrado na difusão de programas que possuíam o código fonte aberto, ou seja, outros programadores teriam acesso ao conjunto de programas de um *software*. Esse desenvolvimento cooperativo era coordenado de maneira informal e não havia tentativas de garantir direitos de propriedade ou restringir o uso dos *softwares* produzidos.

Duas décadas depois, surgiram os primeiros esforços para formalizar as regras fundamentais para o desenvolvimento cooperativo do *software*. A principal instituição durante esse período foi a Free Software Foundation (FSF), iniciada por Richard Stallman, pesquisador do Massachusetts Institute of Technology (MIT), em 1983. Procurava-se desenvolver e disseminar grande variedade de *softwares* sem custo. Outra importante inovação foi o procedimento de licença que procurou evitar o patenteamento de *software* desenvolvido cooperativamente (FSF, 2004).

Entretanto, dessa situação surgiu uma contradição: quais seriam as possibilidades de desenvolvimento e os atrativos originados com essa nova faceta da indústria de *software*? E, principalmente, quais os estímulos econômicos que viabilizariam seu desenvolvimento?

2 REVISÃO DA LITERATURA

Segundo Niemi (1998), *software* de fonte aberta é um termo definido recentemente para descrever os *softwares* cujo código fonte está disponível para o público e que não possui restrições que limitem seu uso, modificação ou redistribuição. O *software* de fonte aberta é geralmente protegido por direitos autorais, e sua licença deve conter restrições, com o objetivo de preservar seu *status* de fonte aberta, notas sobre autoria e controle de desenvolvimento.

Para Lerner e Tirole (2002), *software* de fonte aberta é um processo que envolve programadores em diferentes localidades e organizações que compartilham o código fonte, para desenvolver e refinar os programas. Segundo os auto-

res, o crescente interesse no assunto deve-se à rápida difusão do *software* de fonte aberta, como o Apache, servidor de Internet, e o Linux, que já possui mais de 20 milhões de usuários em todo o mundo.

Considerando a definição apresentada pela Open Source Initiative (OSI, 2002), de forma sintética o *software* de fonte aberta deve atender aos seguintes aspectos:

- não possuir restrições que limitem seu uso, modificação e redistribuição;
- garantir o acesso ao código fonte;
- respeitar a autoria do código fonte original;
- os trabalhos derivados devem ser licenciados como fonte aberta e não discriminá-las pessoas ou campos de empreendimento.

2.1 QUANTAS PESSOAS ESTÃO USANDO SOFTWARE DE FONTE ABERTA?

Segundo Garzarelli (2002), em março de 2002 o Web Server mais usado foi o Apache, com 53% do mercado, para um total de 64% de todos os *sites* ativos. De acordo com o *Linux Counter*, em abril de 2002 havia 125 mil usuários registrados e 95 mil máquinas registradas, estimando-se o uso total de usuários no mundo em 18 milhões. É interessante também notar que alguns dos mais bem-sucedidos fabricantes de microcomputadores (Compaq, 25%; IBM, 10%; HP, 10%; Dell, 7%) recentemente se tornaram uns dos maiores vendedores de servidores Linux.

2.2 O QUE DIFERE SOFTWARE LIVRE/FONTE ABERTA DE SOFTWARE GRATUITO?

Existe uma diferença entre *software* gratuito (*free software*) e *software* de fonte aberta ou livre (*open source software* ou *free software*). Segundo Niemi (1998), os que desenvolvem *software* gratuito o chamam de *free software* ou *freeware*, o que implica custo zero de compra. Muitas empresas de *software* proprietários (código de fonte fechada) às vezes usam o termo *free software* para referir-se a gratuito, porque essas empresas colocam seus produtos disponíveis para demonstração.

Software de fonte aberta ou *software* livre (*open source software* ou *free software*) referem-se à liberdade que o código de fonte aberta possibilita de executar, copiar, distribuir, estudar, adaptar, modificar, aperfeiçoar, corrigir e revisar o *software*. Esse processo é a principal fonte geradora da comunidade de fonte aberta, porque permite a participação no desenvolvimento e no melhoramento da qualidade do *software*.

Existem diferentes tipos de licenças utilizadas para a comercialização de *softwares* proprietário, mas esses não devem ser considerados livres, pois não tornam disponível o código fonte. Alguns exemplos:

- *shareware*: são programas que permitem o uso sem custos e geralmente sem suporte técnico. Entre os objetivos desse tipo de comercialização estão o recebimento de doações ou o estabelecimento de uma marca;
- demonstrações: são versões de *softwares* comerciais geralmente gratuitas, sem suporte técnico e cujas principais funções estão desabilitadas;
- *bundled software*: programa fornecido ou adicionado a um *software* comercial, sob licença severamente restritiva, por exemplo, o Microsoft Internet Explorer;
- *consortium-ware*: código fonte compartilhado entre um grupo de empresas;
- uso somente não-comercial: *download* de *software* pela Internet, às vezes com o código fonte, mas com restrições severas em sua licença, cujo uso é limitado.

O que diferencia o *software* de fonte aberta é o tipo de licença, que impede que seja estabelecida uma patente do *software* desenvolvido e obriga os desenvolvedores, em troca de poder modificar e distribuir, a manter o código fonte disponível e a não impor restrições a outros. Todos os aprimoramentos no código devem ser licenciados nos mesmos termos. Por comparação, o *shareware*, outra forma de licença, torna disponível apenas o código binário livremente e por determinado período de tempo.

2.3 ALGUNS EXEMPLOS DE LICENÇAS DE SOFTWARES DE FONTE ABERTA

Embora o conceito central que define o *software* de fonte aberta seja o mesmo para todos os *softwares* que se encaixem nessa definição, que é a disponibilidade do código fonte para livre utilização, existem vários tipos de licenças para *software* de fonte aberta e elas diferem quanto ao grau de liberdade sobre sua utilização e modificação. Por exemplo, Niemi (1998) cita as principais licenças atualmente em uso e suas características são apresentadas no Quadro 1.

QUADRO I
TIPOS DE LICENÇA

Licença	Características
BSD (Berkeley Source Distribution)	Requer notificação de autoria e <i>copyright</i> .
GPL (General Public License)	A Licença Pública Geral (GPL) proíbe restrições em produtos derivados e distribuídos.
LGPL (Library General Public License)	Difere da GPL no sentido de que as “bibliotecas” podem ser livremente utilizadas em <i>softwares</i> proprietários.
Artistic License	Permite ao autor manter controle de desenvolvimentos futuros.
NPL (7Netscape Public License)	Uma nova licença no espírito da GPL, mas mantendo mais controle e posse do código fonte.
Domínio Público / Sem direitos autorais / Sem restrições	Geralmente usado para algoritmos essenciais, para encorajar amplo uso, e pelo governo dos EUA.

2.4 VANTAGENS E DESVANTAGENS DE USAR SOFTWARES DE FONTE ABERTA

A natureza aberta do código fonte produz certas características únicas, que se revelam na forma de vantagens e desvantagens em usá-lo, em relação ao *software* proprietário. Antes de se passar a usar um *software* de fonte aberta, devem-se pesar com atenção os custos e benefícios associados, para ter certeza se é vantajoso ou não sua adoção. Muitas vezes é melhor adquirir ou continuar com um *software* proprietário, quando for mais adequado e ajustado às necessidades de uma organização. Uma breve análise das vantagens e desvantagens do *software* de fonte aberta, baseada em Niemi (1998), é apresentada no Quadro 2.

QUADRO 2
VANTAGENS E DESVANTAGENS ENTRE
SOFTWARE LIVRE E PROPRIETÁRIO

Tipo de Software	Vantagens	Desvantagens
Software Livre / Software de Fonte Aberta	Flexibilidade e transparência	Não adequação às especificidades da organização
	Confiabilidade no processo de desenvolvimento (comunidade)	Suporte técnico, ter habilidade técnica para implementação
	Risco reduzido de descontinuidade do <i>software</i> , sem custo de licença	Inéria de mudar o <i>software</i> proprietário para o livre
Software Proprietário	Facilidade de uso	Licença já paga
	Fornece suporte técnico	Descontinuidade do <i>software</i>
	Diversidade da oferta de produtos, mais recursos do produto	Restrições na licença

2.5 SEMELHANÇAS ENTRE COMUNIDADE DE FONTE ABERTA E DE PESQUISA CIENTÍFICA

O processo de desenvolvimento da fonte aberta é de certa forma remanescente do tipo de “inovação orientada pelo cliente” encontrada em muitas outras indústrias. Os valores e normas que garantem a qualidade e o compartilhamento do conhecimento são a peça-chave para compreender a cultura fundamental das comunidades de fonte aberta, conforme Bergquist e Ljungbert (2001). Essa possui muita semelhança com a pesquisa acadêmica.

Contribuições científicas geralmente não geram pagamentos pela publicação de resultados. Ao compartilhar o conhecimento e divulgar resultados e

métodos, esses podem ser justificados e reproduzidos. Outros podem dar contribuições, respondendo ou continuando o trabalho publicado e impulsionando o desenvolvimento da ciência. Ao escrever e publicar textos e se tornar referência para outros, o pesquisador passa a ser visível na comunidade acadêmica e sua reputação é garantida pela regra de que qualquer pessoa pode usar o conhecimento produzido pelo pesquisador, desde que se indique claramente de onde a idéia se originou. Quanto mais pesquisadores mencionarem as publicações de um cientista, maior será sua reputação. As comunidades de fonte aberta são guiadas por normas similares.

2.6 FATORES ECONÔMICOS E MOTIVACIONAIS DA COMUNIDADE DE FONTE ABERTA

Nesta parte serão analisados os aspectos econômicos, sociais e administrativos que regem as comunidades de fonte aberta.

2.6.1 Altruísmo

Seria o altruísmo o fator motivador das comunidades de fonte aberta? De acordo com Lerner e Tirole (2002), é surpreendente o comportamento individual dos programadores e das companhias comerciais envolvidas nos processos de fonte aberta, mas uma explicação baseada no altruísmo não se mostra suficiente.

Os meios de comunicação gostam de retratar a comunidade de fonte aberta como um grupo cujo principal objetivo é o de ajudar a espécie humana. Enquanto usuários em países pouco desenvolvidos se beneficiam do acesso a *software* livre, muitos indivíduos ricos e grandes empresas também são beneficiados. Se o altruísmo nunca teve papel importante em nenhuma indústria, seria interessante saber por que os indivíduos na indústria do *software* seriam mais altruístas que os outros?

2.6.2 Custos de oportunidade

130

Segundo Lerner e Tirole (2002), um programador participa de um projeto, comercial ou não, apenas se dele tiver lucro. Um programador que trabalhe no desenvolvimento de um projeto de fonte aberta obtém uma variedade de benefícios e custos, entre os quais está o custo de oportunidade de seu tempo, que pode ser medido pelo dinheiro que deixou de ganhar se estivesse trabalhando em um projeto comercial.

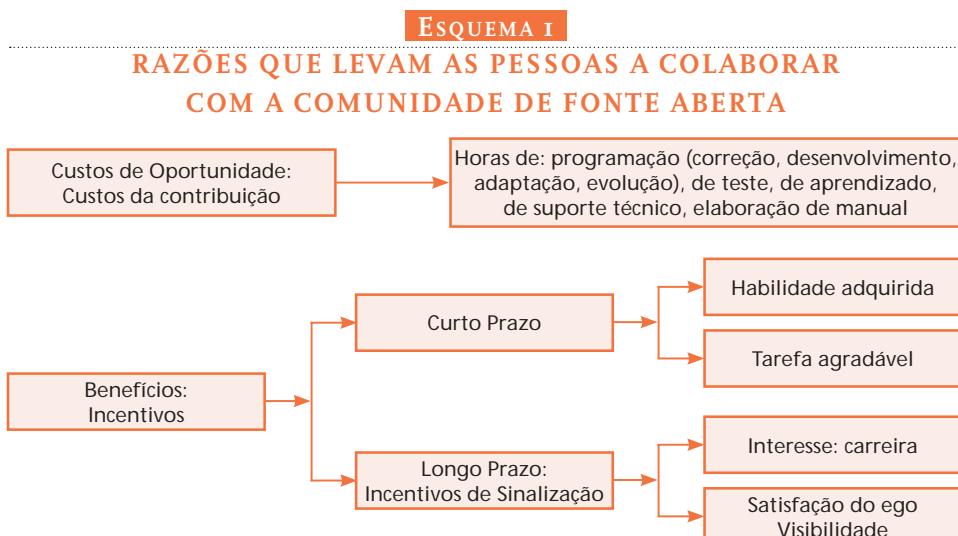
2.6.3 Incentivos

Ao participar das comunidades de desenvolvimento de *software* de fonte aberta, segundo Lerner e Tirole (2002), o programador encontra dois benefícios que se contrapõem aos custos de oportunidade.

No curto prazo, um incentivo é o benefício de adquirir a habilidade técnica ao consertar um erro ou personalizar um programa de fonte aberta; outro é a satisfação do programador que compara quão agradável é a missão determinada por seu empregador em relação a sua contribuição na comunidade de fonte aberta.

No longo prazo, também existem dois incentivos distintos: um é relativo à carreira, quanto às possíveis propostas de emprego, às participações em empresas comerciais de fonte aberta ou ao futuro acesso aos capitais de investimentos; o outro se refere à *satisfação do ego*, que se origina de um desejo pelo reconhecimento de seus colegas. Provavelmente muitos programadores respondam aos dois grupos de incentivos. Há algumas diferenças entre eles, mas de uma perspectiva econômica ambos são muito parecidos e serão tratados sob um mesmo título: incentivos de sinalização.

Assim, o benefício de modificar um *software* de fonte aberta está no desenvolvimento da habilidade técnica, no reconhecimento dessa capacidade, na conquista de uma reputação, na recompensa que é dada ao desenvolvedor do controle sobre a evolução do projeto, além de benefícios materiais, como a contratação por grandes empresas a partir dos relacionamentos construídos durante a participação no movimento de fonte aberta. O esquema I apresenta as razões que levam as pessoas a colaborar com a comunidade de fonte aberta.



2.6.4 Incentivos entre fonte aberta e fonte proprietária

Para poder comparar os incentivos que os programadores recebem em um projeto de fonte aberta com os de participar de um projeto de fonte proprietária, devem-se examinar os incentivos de curto e longo prazo.

De acordo com Lerner e Tirole (2002), os projetos comerciais possuem uma vantagem nos incentivos de curto prazo devido à natureza proprietária do código, que gera um retorno financeiro imediato e torna particularmente compensador o salário oferecido pelas empresas privadas. Tal situação está de acordo com o argumento econômico de que a possibilidade de lucro encoraja o investimento, que tem sido usado, por exemplo, para justificar a concessão de patentes como fator motivador das invenções.

Em contraste, um projeto de fonte aberta pode baixar o custo para o programador, por duas razões: a primeira é o *Alumni effect* (efeito do aprendizado dos alunos), dado que o código fonte está disponível gratuitamente para todos, ele pode ser usado em escolas e universidades, com propósitos educativos. Isso reduz o custo de programação e treinamento, como no caso do Unix. A segunda razão é os *benefícios na personalização e ajustes de erros*, ou seja, o custo de contribuir para um projeto de fonte aberta pode ser compensado se a atividade produz um benefício privado para o programador e sua firma.

Os benefícios de longo prazo estão relacionados aos incentivos de sinalização e são mais evidentes nos projetos de fonte aberta devido a três razões, segundo Lerner e Tirole (2002): 1) melhor medida de desempenho devido às pessoas poderem avaliar também a qualidade intrínseca do produto; 2) controle sobre a iniciativa das contribuições; 3) maior flexibilidade pela grande diversidade dos programadores se comparada com as especificidades/uniformidades dos contratados pelas firmas.

2.6.5 O poder na economia baseada na contribuição

Segundo Bergquist e Ljungbert (2001), existe a impressão de que as trocas de contribuições nas comunidades virtuais criam uma atmosfera cooperativa de desenvolvimento. Tais contribuições não são feitas a alguém em particular, são tornadas públicas e, portanto, disponíveis a qualquer um que queira usá-las. Um *software* ou uma informação passam a ser uma contribuição somente quando alguém os encontra e faz uso deles. Se o colaborador buscar atrair a atenção, as pessoas transformarão as informações oferecidas em contribuições, o que cria um relacionamento entre o colaborador e o usuário. Como as contribuições não implicam uma compensação monetária, nas comunidades virtuais elas são “pagas” pelo respeito e fama conquistada.

Quanto ao controle, os projetos de fonte aberta diferem de outros tipos de organizações virtuais pela importância colocada no controle social e no autocontrole. Essas duas classes gerais de controle são amplamente usadas nos projetos de fonte aberta (GALLIVAN, 2001). Especificamente, identificaram-se quatro grupos de mecanismos de controle que têm sido usados nos projetos de fonte aberta: 1) importância da reputação individual; 2) administração dos membros, que regulam quem deve participar; 3) regras e instituições; 4) sanções, ou ameaças de sanções, contra comportamentos indesejáveis.

Segundo Lerner e Tirole (2002), as características favoráveis para a produção de *software* de fonte aberta são sua modularidade, em que o projeto é dividido em partes menores e com tarefas bem definidas; e a existência de desafios a perseguir. Um projeto bem-sucedido de fonte aberta também requer um líder ou liderança confiável e uma organização consistente com a natureza do processo.

2.6.6 Confiança em projetos de fonte aberta

Para Gallivan (2001), existem três tipos de confiança nos projetos de fonte aberta: 1) confiança na liderança, em que os colaboradores potenciais devem acreditar nos compromissos de longo prazo e na qualidade da liderança – confiança e fidelidade são aspectos importantes no crescimento de um projeto de fonte aberta; 2) confiança no julgamento de outros, que é fundamental no processo de avaliação – o programador confia na comunidade de usuários que testam os códigos de *software* identificando possíveis problemas e apresentando relatórios de erros; 3) confiança em uma comunidade de estranhos, isto é, nos desenvolvedores de *software* que contribuem com sua “propriedade intelectual” para projetos de fonte aberta e se beneficiam diretamente de seus trabalhos por também serem usuários desses produtos.

2.6.7 Novas formas de distribuição e a criação de vantagem competitiva

Conforme Porter (1989), a base fundamental do desempenho acima da média no longo prazo é a vantagem competitiva sustentável. O autor argumenta que as duas formas básicas de criação dessa vantagem são a diferenciação de produtos e o baixo custo. Em artigo recente, Porter e Millar (1999) afirmam que a tecnologia de informação tem acentuado as possibilidades de diferenciação de produtos, por exemplo, ao possibilitar maior flexibilização e individualização dos produtos.

Neste estudo observou-se que as empresas podem atuar gerando uma diferenciação ao se posicionarem com sua estrutura de comunicação, baseada em

um *software*, perante seus usuários e consumidores. Nesse sentido, como será apresentado nos casos estudados, a empresa gera a diferenciação, ainda que esse mercado cativo somente se transforme em valor no futuro, por meio da venda de serviços (consultoria) ou também pela migração dos usuários de *softwares* pessoais para formas de produtos profissionais sujeitas a preços.

Essas estratégias encontram possibilidade de efetivação nas empresas baseadas em conhecimento, que conseguem incorporar a possibilidade de lucro futuro nos valores atuais de suas ações de mercado. De acordo com Castells (1999), o valor esperado é a regra prática do investimento na nova economia.

3 METODOLOGIA

A pesquisa foi desenvolvida por meio de estudo de caso, que, conforme Yin (2001), é um método de pesquisa que deve ser usado quando as questões levantadas pelo estudo são do tipo “como?” ou “por quê?”, quando o investigador não possui controle sobre os eventos e quando o foco é um fenômeno contemporâneo em uma situação da vida real. Adicionalmente foi utilizada a metodologia de análise de conteúdo para a definição de categorias de análise dos dois casos selecionados. A análise de conteúdo, conforme Berelson (1952), pode ser definida como uma técnica de pesquisa para uma descrição objetiva, sistemática e quantitativa do conteúdo manifesto de comunicação.

Para este artigo foram analisados os documentos disponíveis na Internet de duas empresas de *software*, a Grisoft (GRISOFT, 2003), fabricante do aplicativo de segurança antivírus AVG, de fonte proprietária, e a Red Hat (RED HAT, 2003), distribuidora do sistema operacional Linux, de fonte aberta. Para a análise das informações disponíveis nos sites de ambas as empresas foram determinadas unidades temáticas de conteúdo semelhante. Utilizando-se esse método de análise, foram identificadas as seguintes categorias: características operacionais do produto, serviços oferecidos, custo de desenvolvimento e formas de comercialização.

Quanto à razão da escolha dessas empresas, a Red Hat é uma das mais antigas e conhecidas empresas que utilizam, distribuem e prestam consultoria de *software* de fonte aberta. Por outro lado, a Grisoft é uma empresa que adota a política no Brasil de permitir o uso de seu antivírus gratuitamente por usuários de computadores pessoais, e, em decorrência dessa facilidade, sua adoção é muito grande entre as pessoas que utilizam computadores. Ambas as empresas podem ser caracterizadas como fornecedoras de *software* e serviços de produtos.

4 APRESENTAÇÃO DOS RESULTADOS

Esta seção mostra os principais aspectos de cada caso. Na seqüência, ambas as empresas são analisadas e comparadas, a fim de se evidenciarem os motivos econômicos envolvidos em cada uma das diferentes abordagens de negócio.

4.1 A RED HAT

Antes de se apresentar a Red Hat, será feito um breve histórico do Linux, o sistema operacional de fonte aberta que é comercializado pela companhia.

A palavra Linux é a combinação de “Linus” e “Unix”. Esse sistema operacional foi criado por Linus Torvalds em 1991. Devido a sua ampla difusão, o Linux atraiu grande quantidade de investimentos comerciais para os projetos de fonte aberta. Muitas firmas dedicadas a dar suporte ao Linux foram criadas, sendo as pioneiras a VA Linux, fundada em 1993, e a Red Hat, de 1995. Essas empresas comerciais vendem “pacotes” de *software* Linux, os quais são mais fáceis de instalar e operar do que as versões livres disponíveis, providenciam suporte técnico para os usuários finais e revendedores e também vendem produtos complementares proprietários. O Linux é licenciado sob a Licença Pública Geral (GPL – Gnu Public License).

A Red Hat era uma pequena distribuidora de *software* que oferecia aplicativos para Unix, livros e CD-ROMs. Em adição aos produtos Unix, começou a vender uma nova linha: CD-ROMs do Linux, que rapidamente se tornaram *best sellers*. O papel da Red Hat é trabalhar com todas as equipes de desenvolvimento por meio da Internet, pegar cerca de 400 pacotes de *software* e montá-los dentro de um sistema operacional útil. A empresa opera mais como uma montadora de carros, testando produtos e oferecendo suporte e serviços para os usuários do sistema operacional Linux Red Hat (RED HAT, 2003).

A Red Hat oferece suporte técnico por telefone apenas para a instalação do produto. Também está disponível pela Internet um serviço pago que dá suporte e faz atualizações, chamado Red Hat Network. Além disso, a página da empresa na Internet contém os manuais do produto.

O uso do Linux possibilita uma série de benefícios: confiabilidade, facilidade de uso, robustez ou as ferramentas incluídas no sistema. Entretanto, outra vantagem é o controle sobre o sistema operacional: o código fonte aberto completo e a possibilidade de modificá-lo da maneira desejada. A Red Hat oferece benefícios como o suporte técnico e a confiabilidade, com a adição de *softwares* testados pelas equipes da empresa, como o Web Browser Mozilla, o aplicativo OpenOffice, o Office Suite, um Firewall customizável, entre outros produtos.

A estratégia da Red Hat pode ser classificada como orientada para a obtenção de vantagem competitiva baseada na diferenciação. Conforme Porter (1992), para o desenvolvimento de uma vantagem competitiva baseada na diferenciação, a empresa deve localizar os atributos mais valorizados pelos compradores e adicioná-los a seus produtos, de modo que os clientes percebam a superioridade do produto relativamente ao dos concorrentes. No caso da Red Hat, a diferenciação se faz pelo oferecimento de um pacote de *software* certificado e de suporte técnico. Basicamente o trabalho da Red Hat é juntar vários componentes de fonte aberta, testá-los e aprimorá-los e oferecer suporte a seus clientes. Dessa maneira a empresa constrói uma imagem de confiabilidade perante o cliente. Portanto a empresa busca posicionar-se como a marca de Linux mais confiável do mercado (TERRA INFORMÁTICA, 2003).

Ao prestar suporte técnico a seus produtos, a Red Hat está ocupando um nicho de mercado inexplorado e que nunca seria preenchido pela comunidade de fonte aberta, mas que é de vital importância para os usuários comuns do Linux.

A estratégia da Red Hat é considerar o mercado de *software* livre como um mercado de *commodities*, onde os produtos ou os ingredientes estão disponíveis para todos. Sendo assim, a diferenciação entre as marcas se faz pelo posicionamento de mercado adotado pelas empresas.

Young (1999) afirma que o mais importante é o crescimento do mercado do Linux como um todo, e não a participação da Red Hat nesse mercado. É mais vantajoso para a Red Hat trabalhar em um mercado Linux grande do que buscar aumentar a participação em um mercado pequeno, com poucos usuários, e que poderia até vir a desaparecer.

Outro aspecto importante é que o principal benefício do Linux é o controle que o usuário tem sobre o sistema operacional. A única força desfavorável para a Red Hat seria o *poder de negociação dos compradores*. Os clientes é que vão determinar o futuro da Red Hat. Os clientes trocarão um *software* proprietário caro, no caso o Microsoft Windows, mas que é fácil de usar e já amplamente difundido, por um de fonte aberta mais barato? O que determinará isso são questões como: o suporte técnico é suficiente? O programa é de fácil instalação? É fácil de usar? Possui grande número de aplicativos disponíveis?

4 . 2 A GRISOFT

A Grisoft (GRISOFT, 2003) é uma companhia baseada nos Estados Unidos, estabelecida em 1998 como uma *holding* da Grisoft. s.r.o., empresa de alta tecnologia localizada na República Tcheca, especializada no desenvolvimento e venda de *software* antivírus para sistemas de computadores desde 1990.

A empresa desenvolveu seu *software* antivírus AVG e conseguiu significativa participação de mercado em seu país. Atualmente está dedicando seus esforços de *marketing* para o mercado internacional de antivírus, cujo crescimento estimado é de 35% ao ano. Hoje a empresa conta com dois milhões de usuários de seus produtos (GRISOFT, 2003).

A Grisoft vem desenvolvendo um programa de cadastramento de revendedores, afastando-se, dessa maneira, das atividades de venda do *software* e focalizando suas energias em suas competências específicas, que são o desenvolvimento de *software* antivírus e o suporte técnico ao cliente.

A empresa possui três tipos de licença para antivírus: a profissional, para um computador; a profissional multilicença, para mais de um computador; e a *server*, para servidores de rede. Distribui também uma versão gratuita, de fonte proprietária, com algumas funções desabilitadas e sem suporte técnico, para apenas um computador, sem garantias, sendo que a licença não pode ser transferida.

A licença do *software* garante a propriedade, segredos de transações, patentes, marcas registradas e todos os direitos de propriedade relacionados ao *software*. A empresa proíbe copiar, reproduzir, modificar, decompilar, desmontar, fazer engenharia reversa (busca recuperar informações utilizando-se de engenharia de *software*, com base em métodos de especificação e de projeto de *software*) ou criar trabalhos baseados no *software*.

O suporte técnico é oferecido aos usuários registrados, que recebem o número de série do produto, e está disponível de três maneiras: diretamente pelo *software* antivírus, que é altamente recomendado pela empresa, quando o usuário apenas escreve os problemas que está tendo e os envia; pelo preenchimento de um formulário *on-line*; por *e-mail*. Não é fornecido suporte técnico para a versão gratuita do programa.

O programa oferece como benefícios a atualização gratuita das definições de vírus, suporte técnico pelo programa, rede de alerta de vírus, um serviço que envia para o *e-mail* do usuário as últimas notícias sobre infecções de vírus, *hoaxes* e novas versões do programa, página de perguntas mais freqüentes. O produto é certificado pela International Centre for Security Analysis (ICSA), empresa de certificação de *software* de segurança. A Grisoft reembolsa o cliente até o prazo de 60 dias, caso ele não esteja satisfeito com o produto.

Uma estratégia da empresa é o oferecimento de uma versão gratuita do programa, com algumas funções desabilitadas, além disso, terceiriza a distribuição do produto, focando suas energias no desenvolvimento e suporte do produto. Esse oferecimento gratuito do programa pode ser visto como uma estratégia de *marketing*, que busca promover o programa aumentando sua base de usuários, que, em consequência, difunde o conhecimento da marca e cria a massa crítica de usuários, dos quais uma parcela adquire a versão paga.

Pode-se analisar a estratégia competitiva da Grisoft segundo os conceitos de estratégia competitiva propostos por Porter (1992), em que uma empresa adota uma estratégia de *Enfoque*. No caso, a Grisoft emprega o *enfoque no custo*, pois produz apenas *software* antivírus e pratica preços mais baixos que as maiores empresas do setor.

Fazendo uma análise das forças competitivas que atuam no segmento adotado pela Grisoft, segundo o modelo de estratégia competitiva de Porter (1992), pode-se concluir que as maiores ameaças para essa empresa são os novos entrantes e a intensidade da rivalidade. Novas empresas de antivírus podem invadir o mercado e diluir os retornos de todos, enquanto a rivalidade pode fazer com que alguma empresa baixe tanto o custo ou mesmo ofereça seus produtos gratuitamente, inviabilizando o lucro com a venda do *software*. Nesse caso, a Grisoft teria de buscar outra estratégia ou novos produtos para vender.

4 . 3 COMPARAÇÃO ENTRE AS ESTRATÉGIAS ADOTADAS PELA GRISOFT E A RED HAT

A Grisoft é uma empresa comercial, que visa ao lucro e comercializa um *software* proprietário. A Red Hat também é uma empresa comercial, que igualmente visa ao lucro, mas que vende um *software* de fonte aberta e seu suporte.

A natureza fechada do antivírus da Grisoft torna seu desenvolvimento tarefa exclusiva do proprietário de seus direitos, e todos os custos desse desenvolvimento têm de ser arcados pela empresa sozinha. Em comparação, a natureza aberta do Linux permite que seu desenvolvimento seja uma tarefa de todas as empresas que trabalham com esse sistema operacional, o que dilui os custos de desenvolvimento. Isso é uma grande vantagem quanto aos custos de desenvolvimento de *software*, e talvez por isso a Red Hat busque uma estratégia de diferenciação, focando na construção de sua marca. Com custos de desenvolvimento reduzidos, sobram recursos para investir no suporte técnico e na certificação de produtos complementares. Além disso, como o produto é basicamente o mesmo para todos e pode até ser obtido de graça, não adianta focar somente o preço como o único caminho possível da construção de uma marca baseada na diferenciação. A marca precisa inspirar confiabilidade.

Ambas as empresas têm por objetivo aumentar sua participação no mercado, mas o fazem de maneira diferente. A Red Hat utiliza-se de uma estratégia de diferenciação, enquanto a Grisoft busca uma estratégia de enfoque no preço.

Podem-se também diferenciar suas estratégias na ampliação de sua base de clientes, a da Grisoft não é desenvolver o mercado de antivírus, buscando novos

clientes, mas sim conquistá-los de outros participantes, em razão de um preço mais baixo, e para isso oferece inclusive uma versão totalmente gratuita e funcional, embora com algumas funções desativadas.

Tal estratégia visa à ampliação de sua base de clientes e à formação de uma massa crítica de usuários que torne o produto forte e com boa participação no mercado. Além disso, a empresa adotou um programa para terceirizar a promoção e comercialização do produto, por meio de seu plano de cadastramento de revendedores, buscando, com isso, economizar recursos para poder focar seus esforços no desenvolvimento e suporte.

Já a estratégia da Red Hat é promover e desenvolver o Linux, buscando o crescimento do mercado como um todo, aumentando também o número de seus clientes. Essa estratégia de tornar o Linux um sistema operacional amplamente difundido é o maior objetivo de todas as empresas que o comercializam. Dessa maneira, a estratégia da Red Hat é a diferenciação do produto e do serviço, procurando construir a marca da empresa baseada em uma imagem de confiabilidade.

Deve-se observar também que os custos com treinamento de usuários e programadores são maiores com o *software* proprietário da Grisoft, pois os novos contratados só se familiarizarão com o programa ao entrarem para a empresa, enquanto o Linux, por ser de fonte aberta e estar disponível ao acesso das pessoas e dos programadores, tem agilizado e barateado o treinamento dos novos contratados.

Como se pode notar, a comparação das estratégias das duas empresas deixa bem clara a grande diferença existente entre *software* gratuito e *software* de fonte aberta, e do ramo de atuação de cada uma delas.

O Quadro 3 resume os resultados que foram apresentados neste artigo e suas implicações para a estratégia empresarial.

QUADRO 3
ESTRATÉGIAS DE MERCADO –
COMPARAÇÃO ENTRE GRISOFT E RED HAT

Categorias de Análise	Grisoft	Red Hat
Empresa	Visa ao lucro	Visa ao lucro
Características operacionais do produto	Código fonte fechado, antivírus da Grisoft, <i>software</i> gratuito para demonstração	Código fonte aberta, Linux – <i>software</i> livre
Serviços oferecidos	Supporte técnico	Supporte técnico, pacotes aplicativos certificados

QUADRO 3 (CONTINUAÇÃO)
ESTRATÉGIAS DE MERCADO –
COMPARAÇÃO ENTRE GRISOFT E RED HAT

Categorias de Análise	Grisoft	Red Hat
Formas de comercialização	Oferece <i>software</i> gratuito para atrair novos clientes	Colaboração da comunidade pela melhoria do produto Linux como fonte de novos negócios
Custo de desenvolvimento do <i>software</i>	Arcado pela própria empresa, os profissionais recebem remuneração	Dilui custos de desenvolvimento, baseado nos incentivos de sinalização, reduz custo com treinamento pelo “efeito aluno”, porque o código fonte é disponível
Estratégia de conquista do mercado	Enfoque no preço, aumentar participação no mercado com conquista de clientes de seus concorrentes	Diferenciação centrada na construção de sua marca, que inspire confiabilidade
Incentivos de sinalização	Não possui, não tem acesso ao código fonte, não pode publicar suas contribuições	As contribuições realizadas são publicadas e reconhecidas por toda a comunidade. O programador adquire habilidade técnica, visibilidade na comunidade e cria oportunidade de futuro emprego
Custo de oportunidade	Não existe custo de contribuição	Horas empregadas no processo de desenvolvimento de <i>software</i>

5 CONSIDERAÇÕES FINAIS

A primeira observação a ser feita é sobre a grande diferença que existe entre fonte aberta/*software* livre e *software* gratuito, ou *freeware*, como é conhecido. *Software* livre não precisa ser gratuito e *software* gratuito não significa *software* livre.

Na análise sobre a Red Hat e a Grisoft, ambas são empresas claramente voltadas ao lucro e ao crescimento na participação de mercado. A Grisoft oferece o *software* de graça como estratégia de *marketing*, visando à penetração de mercado; já a Red Hat busca a melhoria do Linux como fonte de novos negócios.

No caso das empresas que participam das comunidades de fonte aberta, o retorno que elas recebem por atuar é imediato, pois quando solucionam um problema da comunidade, na verdade, o estão resolvendo também para elas mesmas. Outro incentivo que as empresas têm para participar das comunidades de fonte aberta é o “efeito aluno”: o programa, em razão de o código fonte livre ser utilizado pelos alunos, já é bem conhecido deles, o que reduz enormemente o custo com treinamento.

A vantagem da Red Hat está baseada no fato de o Linux ser um sistema operacional estabelecido no mercado e em plena expansão, cujo desenvolvimento é feito pela comunidade de fonte aberta, e carecer justamente dos serviços que a Red Hat oferece: suporte técnico e pacotes de aplicativos certificados. Por outro lado, a Grisoft precisa concorrer com várias empresas de antivírus já estabelecidas e que atuam em um mercado saturado e desenvolver sozinha seu produto.

Os principais incentivos dos softwares de fonte aberta identificados neste artigo inexistem para a Grisoft. Por exemplo, não há incentivos de sinalização, pois o programador não tem acesso ao código e, portanto, não pode fazer contribuições nem publicá-las, uma vez que o código é patenteado pela empresa, que impede sua publicação; não existe custo de oportunidade, uma vez que os programadores são pagos diretamente por seus trabalhos.

As recompensas de curto prazo que ocorrem no software de fonte aberta, como o “efeito aluno”, em que os programadores já estão familiarizados com o código, também não estão presentes no caso da Grisoft, que comercializa um software proprietário, que nem sempre está disponível para estudo. Também não existe o benefício da personalização e do ajuste de erros, pois os usuários que utilizam o programa não podem modificá-lo.

O mesmo pode ser dito sobre as recompensas de longo prazo, por exemplo, sobre as medidas de desempenho, pois, como o código fonte não está disponível, é difícil para um programador de fora do projeto observar a qualidade intrínseca do software, avaliar o código e, em consequência, torná-lo mais confiável. Ele acaba tendo a mesma visão de um usuário qualquer, que apenas utiliza o software sem poder modificá-lo. Se o código fosse aberto, estaria acessível a uma multidão de programadores, que identificariam com mais facilidade alguma inconsistência do programa.

No caso da Grisoft, o controle da iniciativa nem sempre é responsabilidade do programador e, segundo a teoria econômica, tem-se observado que o desempenho é melhor quando o programador possui a responsabilidade sobre seu projeto. Quando o programa é proprietário, perde-se também em relação à flexibilidade, dado que os colaboradores do código de fonte aberta são de diversos locais no mundo.

6 REFERÊNCIAS

- BERELSON, B. *Content analysis in communications research*. New York: Free Press, 1952.
- BERGQUIST, M.; LJUNGBERT, J. The power of gifts: organizing social relationships in open source communities. *Information Systems Journal*, p. 305-320, Nov. 2001.
- CASTELLS, M. *A sociedade em rede*. São Paulo: Paz e Terra, 1999.
- FSF. Free Software Foundation. Disponível em: <<http://www.gnu.org/fsf/fsf.html>>. Acesso em: jan. 2004.
- GALLIVAN, M. J. Striking a balance between trust and control in a virtual organization: a content analysis of open source software case studies. *Information Systems Journal*, Blackwell Science Ltd., p. 277-204, 2001.
- GARZARELLI, G. *Open source software and the economics of organization*. Disponível em: <<http://opensource.mit.edu/papers/garzarelli.pdf>>. Acesso em: out. 2002.
- GRISOFT. Disponível em: <<http://www.grisoft.com>>. Acesso em: out. 2003.
- KENWOOD, C. A. *A business case study of open source software*. Washington C3 Center, Bedford, Massachusetts: The MITRE Corporation, July 2001. Disponível em: <http://www.mitre.org/support/papers/tech_papers_01/kenwood_software/>. Acesso em: out. 2002.
- KRISHNAMURTHY, S. *Cave or community? An empirical examination of 100 mature open source project*. Disponível em: <http://www.firstmonday.org/issues/issue7_6/krishnamurthy/>. Acesso em: out. 2002.
- LERNER, J.; TIROLE, J. Some simple economics of open source. *The Journal of Industrial Economics*, Blackwell Publishers Ltd., Jun. 2002.
- LINUX COUNTER. Disponível em: <<http://counter.li.org>>. Acesso em: out. 2002.
- MILLER, B. P. et al. *Fuzz revisited: a re-examination of the reliability of unix utilities and services*. Disponível em: <<http://www.gnu.org/software/reliability.pt.html#TOCUtilitariosGNU>>. Acesso em: out. 2002.
- NETCRAFT. Netcraft Web Server Survey. Disponível em: <<http://www.netcraft.co.uk/survey>>. Acesso em: out. 2002.
- NIEMI, D. C. *Why software developers benefit from open-source software*. 1998. Disponível em: <<http://www.tux.org/~niemi/opensource/developer-motivation.html>>. Acesso em: out. 2002.
- OSI. Open Source Initiative. *Open Source Definition*. Disponível em: <<http://www.opensource.org/docs/definition.php>> Acesso em: out. 2002.
- PORTER, M. E. *Vantagem competitiva*: criando e sustentando um desempenho superior. Tradução Elizabeth Maria de Pinho Braga. Rio de Janeiro: Campus, 1992.
- PORTER, M. E.; MILLAR V. E. *Como a informação proporciona vantagem competitiva*. Rio de Janeiro: Campus, 1999.
- RAYMOND, E. *The Cathedral Bazaar*. Disponível em: <http://www.firstmonday.org/issues/issue3_3/raymond>. Acesso em: out. 2002.
- RED HAT. Disponível em: <<http://www.redhat.com>>. Acesso em: out. 2003.
- TERRA INFORMÁTICA. *Estudo compara versões do Linux*. Disponível em: <http://informatica.terra.com.br/interna/0_5862,O1107565-E1553,00.html>. Acesso em: 19 maio 2003.

YIN, R. K. *Estudo de caso: planejamento e métodos.* 2. ed. Porto Alegre: Bookman, 2001.
YOUNG, R. Giving it away – how red hat software stumbled across a new economic model and helped improve an industry. In: DIBONA, C. et al. *Open sources: voices from the open source revolution.* Califórnia: O'Reilly & Associates, 1999.

TRAMITAÇÃO

Recebido em 15/03/2004

Aceito em 28/08/2004

Copyright of Revista de Administração Mackenzie is the property of Universidade Presbiteriana Mackenzie, RAM-Revista de Administracao Mackenzie and its content may not be copied or emailed to multiple sites or posted to a listserv without the copyright holder's express written permission. However, users may print, download, or email articles for individual use.