

Rafael de Almeida Evangelista

Política e Linguagem nos debates sobre o software livre

Dissertação de mestrado apresentada ao
Departamento de Linguística do Instituto de
Estudos da Linguagem da Universidade
Estadual de Campinas sob a orientação do
professor Dr. Eduardo Guimarães

BANCA

Prof. Dr. Eduardo Guimarães (orientador)

Prof. Dr. Suzy Lagazzi Rodrigues

Prof. Dr. Telma Domingues da Silva

Fevereiro de 2005

Ficha catalográfica elaborada pela Biblioteca do IEL - Unicamp

Evangelista, Rafael de Almeida.
Política e linguagem nos debates sobre o software livre / Rafael de Almeida Evangelista. -- Campinas, SP : [s.n.], 2005.

E14p

Orientadora : Eduardo Roberto Junqueira Guimarães.
Dissertação (mestrado) - Universidade Estadual de Campinas, Instituto de Estudos da Linguagem.

1. Tecnologia de informação. 2. Política. 3. Software livre. 4. Linguagem. 5. Governo. I. Guimarães, Eduardo Roberto Junqueira. II. Universidade Estadual de Campinas. Instituto de Estudos da Linguagem. III. Título.

(tjj-iel)

Título: Politics and language on the debates about free software.

Palavras-chave em inglês (Keywords): Information technology; Free software; Politics; Language; Government.

Área de concentração: Análise de Discurso.

Titulação: Mestrado.

Banca examinadora: Prof^a Dr^a Suzy Lagazzi-Rodrigues, Prof^a Dr^a Telma Domingues da Silva Prof^a Dr^a Mônica Macedo-Rouet(suplente).

Data da defesa: 24/02/2005

Resumo

Esta dissertação procura refletir discursivamente sobre os debates que tratam das vantagens e desvantagens da adoção de sistemas livres em computadores (software livre). A partir de uma concepção que considera que o acontecimento de linguagem é um acontecimento político, procura-se entender como e onde o político inscreve-se nesse debate. São objeto de análise nesse trabalho as licenças de software (proprietário e livre); artigos, entrevistas e notícias; e Projetos de Lei que postulam a adoção prioritária de sistemas livres em órgãos governamentais. É também estabelecida uma reflexão sobre o uso de certos termos e nomes (GNU/Linux, software livre)– e não outros (Linux, código aberto) – na referência aos objetos do debate e, para isso, investiga-se a história dos sentidos a eles atribuídos.

Abstract

This dissertation discursively reflects on the debates that deal with the advantages and disadvantages of the adoption of free systems in computers (free software). From a conception that considers that the language event is an political event, it is looked to understand as and where the political enrolls itself in this debate. The software licenses (proprietor and free); articles, interviews and news; and Projects of Law that claimss for priority adoption of free systems in governmental bodies. are object of analysis in this work. This work also reflects on the use of certain terms and names (GNU/Linux, free software) - and not others (Linux, open source) - in the reference to objects of the debate and, for this, it is investigated the history of the means attributed to them.

Agradecimentos

À equipe e ao laboratório de Estudos Avançados em Jornalismo, que me ofereceu estrutura e apoio para a realização desta pesquisa. Aos amigos da revista ComCiência, de hoje e de ontem, que acompanharam meu crescente fascínio pelo mundo do software livre. Ao professor Carlos Vogt, que sempre ofereceu liberdade e incentivo a essa equipe.

Ao professor Eduardo Guimarães, pelo empurrão e apoio na hora certa e pela paciência com minhas idas e vindas.

À professora Telma Domingues da Silva pelo apoio e incentivo ao estudo da linguagem desde os primeiros dias no curso de Jornalismo Científico.

À professora Suzy Lagazzi Rodrigues pelas aulas no Instituto de Estudos da Linguagem e pelos frutíferos comentários na qualificação.

À Mônica Macedo, pela confiança.

Aos melhores amigos Alexandre Zarias, Fábio Vidal, Gustavo Gazotti e Rafael Rachid.

À Juliana Schober pelo maior elogio que eu poderia ter recebido ;-)* :-*

À toda equipe do Planeta Porto Alegre por entender que o outro mundo possível não acontecerá sem o conhecimento livre.

À Rubens Queiroz de Almeida, um dos maiores incentivadores do software livre no Brasil

Ao povo da lista do Quilombo Digital que discute com coragem e inteligência a filosofia do software livre há muito tempo.

Aos novos amigos do software livre, Carlinhos Cecconi, Luiz Fuzaro e aos grandes hipatianos Rafael Gomes, Stefano Barale, Juan Carlos Gentile, Diego Saraiva e Marco Ciurcina.

Sumário

1. Capítulo 1. Introdução, metodologia e contexto	7
2. Capítulo 2. Duas licenças, diferentes restrições	19
3. Capítulo. 3 GNU/Linux ou Linux? Software livre ou código aberto?	42
4. Capítulo 4. Software Livre na periferia do sistema	66
5. Capítulo 5. Conclusões gerais	91
6. Epílogo	95
7. Bibliografia	98
8. Anexos	100
Anexo a. GPL - General Public License	
Anexo b. EULA – Windows 95	
Anexo c. Open Source Definition	
Anexo d. Projeto de Lei de Walter Pinheiro	
Anexo e. Projeto de Lei de Edgar V. Nuñez	

Capítulo 1. Introdução, metodologia e contexto

Desde o fim da década de 1990 do século passado é crescente o debate no Brasil em torno das vantagens e desvantagens da adoção de softwares livres, tanto pela esfera pública quanto pela iniciativa privada. Em especial, a discussão intensifica-se a partir de 1999, quando grupos de usuários e desenvolvedores de software livre, com apoio da administração do município de Porto Alegre e do governo do estado do Rio Grande do Sul, promovem o I Fórum Internacional de Software Livre, na capital gaúcha. Alguns desses usuários são parte integrante do governo municipal de Porto Alegre e do governo do Rio Grande do Sul, ambos controlados pelo Partido dos Trabalhadores (PT).

Desde esse período, é referida a presença de “políticos” entre os grupos de usuários e defensores do software livre, algo que inquieta e gera reclamações por parte do grupo de pessoas que chamavam a si mesmos como “a comunidade software livre”. Essa comunidade, formada essencialmente por indivíduos com experiência em trabalhos descritos como “técnicos”, incomoda-se com a presença dos “políticos” e inicia um debate que dura até hoje, discutindo até que ponto é interessante para os objetivos gerais do movimento o envolvimento com a política. A palavra “política” é usada, na maior parte das vezes, para fazer referência ao envolvimento de partidos políticos no movimento.

Nos últimos cinco anos, a referência à “política” como algo externo ao movimento e capaz de atrapalhá-lo ou ajudá-lo, tem se repetido com frequência e é lembrada tanto pelo setor que defende o uso dos softwares livres como pelo setor crítico à sua adoção. Mais recentemente (nos últimos dois anos), a escolha por parte do governo federal brasileiro do software livre como a plataforma

preferencial para o uso pela administração pública e em projetos de inclusão digital fez com que o debate caminhasse ainda mais na direção do questionamento do “político” no software livre. Uma das marcas fundamentais do debate em torno do software livre será, então, a discussão entre o que é reconhecido como “técnico” e o que é reconhecido como “político”.

E essa questão não se restringe à comunidade software livre brasileira. A origem do movimento software livre é atribuída aos EUA, cujo representante institucional de maior peso é a Free Software Foundation (FSF) - presidida por Richard Stallman, o programador que sintetizou os princípios fundamentais do movimento – onde também é marcado pelas categorias Técnica X Política e outras divisões em categorias que são desta derivadas como: eficiência X ideais; razões práticas X razões ideológicas; não-politização X politização, entre outras. Richard Stallman muitas vezes é chamado de “comunista”: a qualificação tem sido tão freqüente que já se repete até mesmo em tom de piada em websites dedicados a notícias do setor de Tecnologia da Informação¹. Produz-se o efeito de uma rejeição à FSF, pois no contexto desse debate (tecnologia da informática/ mercado), e sobretudo nos EUA, o termo “comunista” carrega um sentido negativo/ pejorativo. Aqui se estabelece uma relação em que uma posição reconhecida como “de esquerda” – comunista, socialista ou progressista – significa “estar ligado à política”, enquanto a associação com o livre mercado ou a não rejeição ao sistema econômico capitalista significaria, por oposição, uma imaginada neutralidade.

Na área da sociologia, vários pesquisadores têm debatido a ligação entre o software livre, marxismo e a esquerda de uma maneira geral.

1 Em 1o de abril de 2004, o site News Forge, um dos mais populares entre a comunidade software livre publicou uma notícia-trote em que afirmava que o GNU Emacs, software criado por Stallman, exibia mensagens subliminares como “o Cristianismo é estupidez” ou “o Comunismo é bom”. “Secret messages found in GNU Emacs”, em News Forge, publicado e visualizado em 01/04/2004.

Mais recentemente, em 7/01/2005, o presidente da Microsoft, Bill Gates, comparou todos aqueles que buscam uma flexibilização do sistema atual de propriedade intelectual com comunistas, dando a seguinte declaração (“Gates taking a seat in your den”, em *News.com*, http://news.com.com/Gates+taking+a+seat+in+your+den/2008-1041_3-5514121.html?part=rss&tag=5514121&subj=news.1041.5:

“No, I'd say that of the world's economies, there's more that believe in intellectual property today than ever. There are fewer communists in the world today than there were. There are some new modern-day sort of communists who want to get rid of the incentive for musicians and moviemakers and software makers under various guises. They don't think that those incentives should exist.”

Nesse sentido, um dos trabalhos mais interessantes é o artigo de Johan Söderberg, intitulado “Copyleft vs Copyright: a marxist critique”. Trata-se de uma revisão da publicação de diversos autores de filiação marxista que discutem o impacto e o papel dos movimentos que questionam as leis de direito autoral. O artigo estabelece uma reflexão que pretende filiar-se ao materialismo histórico, definindo este como uma ênfase no papel da infra-estrutura em lugar da superestrutura como motor de desestabilização das relações históricas de produção. Para Söderberg, a filosofia hacker, que está na base do movimento software livre, em seus itens principais, dialoga com conceitos de Karl Marx sobre alienação, divisão do trabalho perda de qualificação e reificação. Uma de suas conclusões principais é que a contradição que se instaura não é entre um regime de propriedade e licenças, mas entre licenças que dão base a um regime proprietário e outras que fortificam um sistema comunal.

Entretanto, embora a análise de Söderberg coloque-se como um estudo da literatura e contribua ao afirmar a relevância do texto das licenças como documentos que definem a contradição instaurada, sua abordagem do tema não se dá pela análise do interdiscurso como objetividade material, desconsiderando que “é no discurso que o homem produz a realidade com a qual está em relação”². O autor trabalha a exterioridade como evidência empírica e assim, considera fenômenos como o open source/código aberto de derivados da Ideologia da Califórnia³, ou seja, uma manifestação de direita que contradiz a raiz do movimento software livre.

A contribuição das ciências da linguagem

Nesse debate, a Lingüística, em especial a Análise do Discurso e os estudos da enunciação, pode oferecer uma contribuição bastante interessante. Não se trata de confirmar ou refutar a idéia de que “há

2 Orlandi, 1996, p.39

3 A Ideologia da Califórnia é descrita por Richard Barbrook (Barbrook, Richard e Cameron, Andy. Californian Ideology, 1995. Em: http://www.alamut.com/subj/ideologies/pessimism/califideo_1.html) como a confluência de ideais hippies com certas condições privilegiadas de emprego na região do Vale do Silício (Califórnia, EUA) que teriam dado origem a uma visão otimista do potencial emancipatório das novas tecnologias, fundindo ideais libertários com o empreendedorismo individualista dos yuppies.

muitos políticos no movimento”, nem mesmo discutir se este dialoga mais fortemente com o campo da esquerda. Responder a tais perguntas é manter a oposição entre Técnica e Política. No presente trabalho, dada a sua inscrição teórico-metodológica, considera-se que a política é inerente à produção do social, atravessando todo e qualquer debate. Tendo em vista a Semântica Histórica da Enunciação, conforme Guimarães, o “acontecimento de linguagem por se dar nos espaços de enunciação é um acontecimento político”⁴.

O software livre, embora seja usualmente ligado a uma certa arquitetura de construção de um sistema operacional, não pode ser classificado, em si, como um objeto técnico. O que define um software como livre ou proprietário não está dado em sua arquitetura, mas pela sua forma de licenciamento isto é, no modo como é regulamentado juridicamente, regulamentação que configura/autoriza determinadas relações na sociedade, e não outras. Um software será livre se for enunciado que seu usuário terá quatro liberdades fundamentais: cópia, uso, modificação e distribuição. Em geral, a idéia de software livre está ligada a um sistema operacional em especial, o GNU/Linux, mas softwares livres produzidos para funcionarem em sistemas proprietários como o Windows também existem em grande número. Um software será, então, livre, se assim for enunciado.

O debate que se estabelece entre software livre e software proprietário será marcado pela contradição entre duas ordens. As licenças livres, ou seja, as “liberdades” que os software regulados por elas oferecem, serão objeto de uma tentativa de silenciamento. As licenças livres, como veremos, serão classificadas como “impróprias”, serão comparadas a patologias virais que afetam a saúde do corpo social pelos defensores do licenciamento proprietário. No centro desse debate está a contradição entre uma norma que estabelece uma certa divisão do real (a licença proprietária) e outra, que questiona essa divisão, afirmando-a desigual e afirmando uma nova ordem. Guimarães⁵ dá ao político o sentido de uma “contradição entre uma normatividade que estabelece (desigualmente) uma divisão do real e a

4 Guimarães, 2002, p. 17

5 Idem, p. 16

afirmação do pertencimento dos que não estão incluídos”. Veremos que as licenças são os documentos de referência e que a defesa, pelas partes, de uma ou de outra forma de licenciamento ou tecnologia (GNU/Linux ou Windows, os sistemas operacionais que são tomados como os maiores exemplos de sistemas livres ou proprietários), será a defesa de uma certa divisão do real estabelecida por esses documentos

Sujeito, ideologia e sentido

Também importante na presente análise é o conceito de interdiscurso⁶ e uma reflexão sobre a memória do dizer. Procurarei aqui estabelecer um acompanhamento da história de certos sentidos postos em jogo no debate sobre o software livre. Esse acompanhamento deve ser descrito não como uma cronologia de fatos, mas como o estabelecimento de relações de sentido que se dão na história e que só podem se estabelecer tendo como referência o já-dito, o que está na memória e o que será possível dizer a partir do momento em que novas coisas são ditas, ou melhor, são resignificadas. De acordo com Orlandi, “os sentidos não estão nas palavras elas mesmas. Estão aquém e além delas.” E ainda: “Conseqüentemente podemos dizer que o sentido não existe em si mesmo, mas é determinado pelas posições ideológicas colocadas em jogo no processo sócio-histórico em que as palavras são produzidas”⁷.

Assim, o que pretendo é, por meio da análise dos sentidos dados a certos termos (como livre, aberto, liberdade, atraso, eficiência e outros), certas idéias (o que é ser livre, o que é ser economicamente viável, o que é ser comunista, o que é a política, por exemplo), certos nomes próprios (Linux e GNU/Linux ou Windows), entender como se constitui o político e onde se posicionam os sujeitos ideologicamente nesse debate, que em certos momentos se pretende exclusivamente

6 No sentido que este conceito tem para Orlandi, 1992

7 Orlandi, 2003. p.42

tecnológico.

Entender a constituição desses sentidos será também, como veremos, entender a constituição e o posicionamento dos sujeitos. Aqui, penso o sujeito como descrito teoricamente pela Análise de Discurso, como interpelado pela ideologia. Assim, este não escolhe livremente seu posicionamento, é tornado sujeito a partir de certas condições históricas de produção de sentidos, que o colocam em ação (em processo de significação) sob o desígnio da ideologia. De acordo com Orlandi, “a ideologia faz parte, ou melhor, é a condição para a constituição do sujeito e dos sentidos”. E ela continua, citando e reformulando M. Pecheux, “o indivíduo é interpelado em sujeito pela ideologia para que se produza o dizer”⁸.

Veremos como um dos elementos mais importantes no debate sobre a adoção do software livre será a constante re-significação de certas palavras e termos. O modo como esses sentidos serão reformulados e re-significados, em relações de polissemia e paráfrase, implicará em deslocamentos e em tomadas de posições dos sujeitos. Ao mesmo tempo, esses movimentos de significação serão derivados de certas possibilidades do dizer e do interpelamento desses sujeitos pela ideologia. O debate em torno do software livre (e que enfatiza certas qualidades ou defeitos dos softwares ou então certos benefícios ou desvantagens das licenças livres) mostra-se um campo bastante interessante para se refletir sobre que tipo de interação entre máquina e usuário está em questão e é imaginada por aqueles que, ao pleitearem o uso de certas tecnologias, falam também sobre desenvolvimento, inclusão, autonomia, independência, entre outros.

Entre pingüins e gnus

A gênese do movimento software livre está no início dos anos 1970. Nesse período, pesquisadores e programadores costumavam trocar livremente o produto de seu trabalho, as linhas de

8 Idem, p.46

código que formam o software. O sistema operacional mais usado – a "plataforma" sob a qual os outros softwares são executados – era o Unix, de propriedade da empresa norte-americana AT&T. Em 1979, entretanto, a empresa decide transformar seu sistema em um produto para a venda, aproveitando-se do que era uma tendência no mercado de informática. Isso significou o surgimento de algumas dificuldades e restrições para os pesquisadores e programadores da época, até então acostumados com um livre intercâmbio de códigos e idéias.

Para que as restrições não os atingissem, pesquisadores da Universidade de Berkeley decidiram criar o BSD Unix, uma versão própria do Unix. Essa iniciativa se fortaleceu em 1983, quando Richard Stallman decidiu ir além do BSD e iniciou os trabalhos no GNU (acrônimo para Gnu is not Unix), um sistema operacional gêmeo do Unix (perfeitamente compatível) mas mais completo e construído coletivamente. Dois anos depois ele funda a Free Software Foundation⁹ e escreve a licença do GNU, a General Public License (GPL) a primeira licença de software livre.

Foi na década de 1980 que se estabeleceu o modelo de negócio que até hoje dá a tônica de atuação de boa parte das empresas de informática. Nesse modelo, distinguem-se e delimitam-se dois produtos: hardware – a máquina física, os equipamentos eletrônicos, chips, placas de memória e outros – e software – o conjunto de instruções codificadas que são inseridas nos computadores e que organizam e regulam seu funcionamento. As empresas, nesse momento, dividem-se – e congregam esforços, quando a situação se mostra mais lucrativa - basicamente entre aquelas que atuam na concepção e fabricação desses dois produtos, físicos (hardware) ou intangíveis (software). Até esse período, o que seria o "produto software" era algo de pouco valor, ou melhor, sobre o qual não era

9 "The Free Software Foundation (FSF), founded in 1985, is dedicated to promoting computer users' right to use, study, copy, modify, and redistribute computer programs. The FSF promotes the development and use of free (as in freedom) software ---particularly the GNU operating system(used widely today in its GNU/Linux variant)--- and free (as in freedom) documentation. The FSF also helps to spread awareness of the ethical and political issues of freedom in the use of software. Many organizations distribute whatever free software happens to be available. In contrast, the Free Software Foundation concentrates on development of new free software---and on making that software into a coherent system which can eliminate the need to use proprietary software. Besides working on software development, the FSF protects, preserves, and promotes free software. The FSF distributes copies of GNU software and manuals for a distribution fee, and accepts tax-deductible gifts to support GNU development. Most of the FSF's funds come from its distribution service. This is why we urge you to order CD-ROMs and manuals (but especially CD-ROMs) from the FSF when you can." <http://www.fsf.org/fsf/fsf.html>

estabelecido um valor de troca. Ele era considerado como parte da máquina e aqueles que se ocupavam profissionalmente de seu desenvolvimento estavam habituados a compartilhar e trocar o produto de seu trabalho. O cenário em que esses hábitos se desenvolveram era o universitário.

Ao contrário do que era a tendência da época (e que hoje vive seu clímax), em que as grandes empresas buscam auferir o máximo possível de lucros a partir dos direitos autorais sobre o software, limitando seu uso cópia e alteração, o GNU estabelecia liberdades a serem respeitadas em conjunto com certas obrigações. O software, que no limite não passa de um conjunto de instruções em código transmitidas ao hardware, em geral permanece com seus direitos vinculados ao seu autor ou à corporação que contratou o trabalho de um conjunto de autores. Quando alguém paga por um software, não compra esses direitos, mas obtém a permissão para um uso restrito desse código, regras essas descritas na forma de uma licença.

A licença criada por Stallman, a GPL (General Public License – Licença Pública Geral), funciona de forma distinta das que são obtidas no mercado hoje e que regulam o uso da maioria dos softwares. Ela estabelece, previamente, que todos têm o direito de copiar, alterar, usar e distribuir esse código, desde que a autoria seja mantida¹⁰. O autor, ou qualquer pessoa ou entidade que adquiriu uma cópia desse código, pode vendê-lo, mas não pode impedir aquele que o comprou de distribuir a quem quiser, em troca de dinheiro ou não.

Assim como as obras de arte e como qualquer trabalho intelectual, o software, se não houver um pronunciamento em contrário do autor, é imediatamente objeto de copyright. A licença criada por Stallman, bem como outras que vieram a seguir, estabeleceram um meio termo entre o copyright e o domínio público (a permissão de qualquer uso sem restrições). Em contraposição ao termo copyright

10 O software livre, regulado por qualquer licença, deve pressupor os quatro itens abaixo, definidos pela Free Software Foundation:

"The freedom to run the program, for any purpose (freedom 0).

The freedom to study how the program works, and adapt it to your needs (freedom 1). Access to the source code is a precondition for this.

The freedom to redistribute copies so you can help your neighbor (freedom 2).

The freedom to improve the program, and release your improvements to the public, so that the whole community benefits (freedom 3). Access to the source code is a precondition for this." <http://www.fsf.org/philosophy/free-sw.html>

foi criado o termo copyleft, um trocadilho com a palavra direitos (*rights*) e o verbo deixar, liberar (*to left*). Mas o termo também faz uma alusão interpretada por alguns como política, entre as palavras direita (*right*) e esquerda (*left*). Os dois termos indicam ainda a contraposição entre direita/ esquerda, como emblema do confronto que fundamenta a disputa de poder em nossa sociedade.

Após a GPL surgiram outras licenças alternativas, mais ou menos restritivas quanto ao uso do software. Todas elas, para serem consideradas “livres” e opostas às “proprietárias”, deveriam introjetar o espírito do copyleft¹¹ e serem chanceladas por uma organização responsável, a Free Software Foundation (FSF). Atualmente, esse modelo de licenciamento alternativo do trabalho intelectual já se estende a outros produtos culturais, como livros, músicas, filmes, imagens e outros, também regulados por certas organizações (como a Creative Commons¹²). A justificativa para elas tem sido fazer frente à forte restrição ao uso que o copyright estabelece.

Rapidamente, uma “comunidade” passou a se estruturar em torno da modalidade software livre. Como a cópia e o compartilhamento do código não estão restritos, criaram-se condições materiais para que os programadores pudessem reescrever, inspecionar e utilizar partes do código criado por outros. Mas o grande impulso para o trabalho colaborativo e para a comunidade aconteceu nos anos 1990, com a internet. Tendo um meio material de comunicação acessível e veloz com o mundo todo, a comunidade floresceu. Foi nesse período, também, que surgiu aquela que seria a face mais visível dessa comunidade, quando Linus Torvalds, um jovem programador finlandês, buscou parceiros (lançando a idéia por meio um e-mail enviado a algumas listas de discussão) para o desenvolvimento do Linux, um

11 Copyleft é definido pela Free Software Foundation como: "Copyleft is a general method for making a program free software and requiring all modified and extended versions of the program to be free software as well. The simplest way to make a program free is to put it in the public domain, uncopyrighted. This allows people to share the program and their improvements, if they are so minded. But it also allows uncooperative people to convert the program into proprietary software. They can make changes, many or few, and distribute the result as a proprietary product. People who receive the program in that modified form do not have the freedom that the original author gave them; the middleman has stripped it away. In the GNU project, our aim is to give all users the freedom to redistribute and change GNU software. If middlemen could strip off the freedom, we might have many users, but those users would not have freedom. So instead of putting GNU software in the public domain, we ``copyleft" it. Copyleft says that anyone who redistributes the software, with or without changes, must pass along the freedom to further copy and change it. Copyleft guarantees that every user has freedom."

<http://www.gnu.org/copyleft/copyleft.html>

12 [Http://www.creativecommons.org](http://www.creativecommons.org)

kernel (cerne, uma parte do sistema operacional) para o GNU. Hábil no gerenciamento do trabalho comunitário, Linus conseguiu transformar o Linux em algo mais popular que o próprio GNU, criando, inclusive, um mascote-símbolo (um pingüim) que é mais popular como símbolo do movimento do que o mascote do GNU (um gnu). Embora a vasta maioria dos sistemas livres rode na verdade a dobradinha GNU/Linux, o pingüim identifica mais as iniciativas livres para o grande público do que o desconhecido gnu¹³.

No Brasil, embora algumas publicações especializadas e algumas iniciativas localizadas tenham sido levadas a cabo em meados da década de 1990, é no final desta e no início do século seguinte que as idéias do movimento software livre tornaram-se mais populares.

Entre os responsáveis por essa popularização estiveram, é claro, programadores, boa parte deles trabalhando nas universidades - embora registrem-se iniciativas empresariais como a Conectiva¹⁴ - mas também setores da mídia alternativa e políticos.

O final dos anos 1990 é marcado por iniciativas consistentes partindo de políticos ligados a partidos como o PT, o PDT e o PC do B, que procuram adotar softwares livre em administrações estaduais e municipais. Registra-se, inclusive, um Projeto de Lei enviado ao Congresso Nacional propondo a adoção preferencial de softwares livres na administração federal. Há também projetos semelhantes apresentados a câmaras de vereadores e assembleias legislativas.

Entre as justificativas estão a maior segurança e estabilidade e a redução de custos, pois os sistemas livres podem ser copiados e instalados em várias máquinas com o mesmo custo – que muitas

13 Eric Raymond coloca a internet como condição fundamental para a explosão do Linux e do software livre, embora também ressalte também o desenvolvimento de uma nova metodologia de trabalho: "Linux foi o primeiro projeto a fazer um esforço consciente e bem-sucedido a utilizar o mundo inteiro como sua reserva de talentos. Eu não acho que seja uma coincidência que o período de gestação do Linux tenha coincido com o nascimento da World Wide Web, e que o Linux tenha deixado a sua infância durante o mesmo período em 1993-1994 que viu a expansão da indústria de ISP e a explosão do principal interesse da Internet. Linus foi a primeira pessoa que aprendeu como jogar com as novas regras que a onipresente Internet fez possível. Embora uma Internet barata fosse uma condição necessária para que o modelo do Linux evoluísse, eu penso que não foi uma condição por si só suficiente. Outro fator vital foi o desenvolvimento de um estilo de liderança e conjunto de formalidades cooperativas que permitiria aos desenvolvedores atrair co-desenvolvedores e obter o máximo suporte do ambiente." Raymond, E. *A catedral e o bazar*. <http://www.geocities.com/CollegePark/Union/3590/pt-cathedral-bazaar-10.html>

14 A Conectiva, empresa paranaense fundada em 1995, foi a primeira iniciativa empresarial de distribuição do GNU/Linux a se mostrar viável financeiramente.

vezes é zero – enquanto que os sistemas proprietários exigem o pagamento de uma nova licença para cada máquina. Como os produtores de software e os detentores dos direitos autorais estão, em grande parte, localizados no exterior, isso significa também o pagamento de royalties e a remessa de dinheiro ao exterior.

Para realizar, então, o objetivo deste trabalho – o exame do funcionamento do político no debate sobre o software livre tendo em vista uma análise sobre a história dos sentidos – proponho a realização de alguns passos essenciais.

O primeiro deles, e um dos mais fundamentais, é a análise de duas licenças que exemplificam o ordenamento do real proposto pelo software livre e pelo software proprietário. As licenças são contratos jurídicos que enunciam que tipo de relação aquele que paga poderá estabelecer com o produto software adquirido. Analisarei a licença padrão do windows 95 para pensar o software proprietário e a General Public License (GPL) para refletir sobre o software livre.

No capítulo seguinte, procuro analisar um momento que penso fundamental para o software livre, o surgimento do termo código aberto e a criação da Open Source Initiative (OSI), instituição que em muitos momentos rivaliza com a Free Software Foundation (FSF). A OSI possui funções muito semelhantes às da FSF, mas seu funcionamento discursivo e institucional se mostrará diferenciado.

No terceiro capítulo, volto a análise para o contexto dos países periféricos, mais especificamente para países da América do Sul (Brasil e Peru). Interessa-me entender como funciona a idéia de produção e uso de softwares livres em países com pouco acesso à tecnologia e com alta desigualdade social. Para isso, analiso alguns Projetos de Lei que instituem a preferência do uso de sistemas livres para a administração pública desses países. O poder legislativo de Brasil e Peru, de maneira pioneira no contexto mundial, propuseram essas leis que marcam uma intervenção direta do poder público sobre o uso de determinadas tecnologias de informação, ação justificada por princípios éticos e economicos, os quais examino.

Nesses três capítulos, faço uso de material jornalístico de diversos gêneros (comentários, entrevistas, artigos, crônicas) como elemento importante para a análise. As fontes foram tanto sites de notícias com grande visitação como blogs, páginas independentes na internet e listas de discussão. Assim, procuro combinar a análise de documentos como contratos e projetos de lei com pronunciamentos e comentários públicos, que circulam na velocidade da internet e que partem de fontes diversas como executivos, programadores, jornalistas e ativistas. Interessa-me, em especial, entender como circulam os sentidos e as interpretações sobre o que é ou não dito.

Capítulo 2. Duas licenças, diferentes restrições

Introdução

Embora o termo *software livre* seja ligado, usualmente, ao sistema operacional GNU/Linux (o projeto GNU de Richard Stallman mais o kernel Linux de Linus Torvalds¹⁵) o termo designa muito mais do que isso. Ele opõe-se diametralmente ao termo software proprietário – ou, como veremos e discutiremos em outro momento, software comercial¹⁶ -, de acordo com condições que examinaremos a seguir.

Para que um software seja livre, ele não precisa ter nem uma linha de código desses programas mais famosos. Não é o modo como um programa é construído tecnicamente que define se este é livre ou proprietário, mas sim o modo como este é licenciado, ou seja, como é feito o registro legal dessa obra técnica. Basta que a sua licença incorpore certos princípios, que foram sistematizados por Richard Stallman, um dos pioneiros do movimento software livre e fundador da Free Software Foundation. A lei de registro da propriedade de software, na maioria dos países, incluindo o Brasil, estabelece que a expressão do programa, ou seja, as linhas de código que o constituem, é que deve ser objeto de registro.

Assim, as idéias contidas em um programa podem ser repetidas em outro, sem que o autor legal tenha

15 GNU é um acrônimo para GNU não é Unix. O projeto GNU foi fundado por Richard Stallman. GNU e Linux são as duas partes que formam o sistema operacional livre mais utilizado no mundo. GNU é a base do sistema, onde está quase a totalidade dos programas necessários para o funcionamento. O Linux designa uma pequena parte desse sistema operacional, o chamado kernel (cerne), mas de vital importância para o funcionamento do sistema, pois realiza a o gerenciamento dos dispositivos (mouse, teclado, monitor e outros). A discussão e a história de ambos será tratada posteriormente, assim como as consequências do uso de cada um dos nomes.

16 Software Proprietário e software comercial não são sinônimos. O software livre também é passível de venda comercial. Entretanto, os defensores do software proprietário tratam os termos comercial e proprietário estabelecendo uma relação parafrástica. Assim, o software livre teria como antônimo o software, produzindo-se a idéia de que ele não poderia ser vendido. A análise dessa relação será aprofundada no próximo capítulo.

direito sobre ela. No entanto, a expressão dessas idéias (assim como as notas de uma música ou a ordem das palavras contidas em um livro) é que é de controle de quem a registra.

Quando o consumidor vai até uma loja de software e compra uma caixa contendo os CDs de instalação de algum programa, essa pessoa está, na verdade, adquirindo apenas o direito de usar aquilo. Ela nunca será dona do software, que continua de propriedade de seus autores. É como quando se compra um CD de música ou um filme em DVD: é permitido ouvir ou assistir aquilo mas não é permitida a venda de cópias e nem mesmo a venda de uma versão alterada do original sem a permissão do autor – um remix, por exemplo.

Porém, no caso do software, as condições de venda são ainda mais duras. Acompanhado dos CDs de instalação está esse documento de valor jurídico chamado licença. Na licença estão descritos os direitos e os deveres dos “usuários” - não donos, nem proprietários - daquele programa. Os verdadeiros donos, os detentores dos direitos autorais ou das patentes, colocam certas restrições para os usuários, estipulando o que ele pode fazer ou não com programa. Na prática, é como se as gravadoras pudessem dizer em que aparelho os CDs podem ser tocados ou como se uma editora pudesse limitar quem pode ler os livros que ela publica.

Para que um software seja livre, essa licença precisa ser filosoficamente, ideologicamente diferente das licenças proprietárias (do Windows, por exemplo). Quando alguém compra um programa proprietário, na verdade adquire o direito de usar algo que é de propriedade de outro. Para usá-lo, é preciso aceitar as limitações impostas pelo proprietário. Já para usar (apenas para o uso) um programa livre não é necessário nem ao menos aceitar a licença nele contida. Como veremos aqui, a liberdade é considerada um princípio básico, é uma palavra utilizada com frequência e ancorada numa reivindicação política. Para a Free Software Foundation, um programa é livre quando a cópia, alteração, uso e distribuição pode ser feita por qualquer um, independente de autorização prévia.

Mas o software que é qualificado como livre também coloca restrições, até porque, como

veremos, o sentido dado à palavra liberdade não é o da ausência de regras. Para se estabelecerem as liberdades reivindicadas pelo movimento software livre é preciso que sejam colocadas certas normas. Veremos que isso decorre do fato de que a disputa de sentidos em torno da palavra liberdade descortina um embate que só pode ser entendido no campo da política. Uso o sentido de político aqui como o desenvolvido por Eduardo Guimarães:

“O político, ou a política, é para mim caracterizado pela contradição de uma normatividade que estabelece (desigualmente) uma divisão do real e a afirmação de pertencimento dos que não estão incluídos. Deste modo o político é um conflito entre uma divisão normativa e desigual do real e uma redivisão pela qual os desiguais afirmam seu pertencimento. Mais importante ainda para mim é que, deste ponto de vista, o político é incontornável porque o homem fala. O homem está sempre a assumir a palavra, por mais que esta lhe seja negada

Esta concepção nos leva a algumas considerações aparentemente contraditórias em princípio. O Político está assim sempre dividido pela desmontagem da contradição que o constitui. De tal modo que o estabelecimento da desigualdade se apresenta como necessária à vida social e a afirmação de pertencimento, e de igualdade, é significada como abuso, impropriedade”¹⁷.

Como veremos mais adiante, a afirmação de pertencimento, de igualdade, que impulsiona o movimento software livre, será classificada, por executivos de empresas de software proprietário, como abuso, impropriedade, algo que seria capaz de corromper a integridade social e do mercado capitalista.

Um programa livre sempre permite o uso, o estudo, a alteração, e distribuição de seu código e de sua documentação – os manuais de uso. Mas, enquanto os dois primeiros pontos nunca são restringidos, a alteração e a distribuição é disciplinada. Ninguém pode pegar um programa livre e torná-lo proprietário, tornar exclusivamente seu o que foi feito por outro, mesmo que essa pessoa tenha alterado o código, mesmo tendo melhorado o programa que recebeu.

As restrições, na verdade, são muito pequenas se comparadas às impostas pelo software proprietário. Elas apenas disciplinam algo que é impossível em softwares como o Windows, ou seja, olhar e modificar o código. Já aquilo que os sistemas proprietários disciplinam – a execução e o estudo do código – é algo permitido de forma irrestrita pelas licenças livres. Cidadãos de países considerados

17 Guimarães, 2002, p. 16

inimigos dos EUA, como Cuba, que são sujeitos a sanções comerciais, não podem usar os produtos da Microsoft, uma empresa estadunidense, devido às leis daquele país.

Uma análise do texto de licenças proprietárias e livres pode nos dizer algo sobre a posição discursiva de cada uma delas, ajudando-nos a situá-las umas em relação às outras e nos dando pistas para melhor desvendarmos e compreendermos o que há de político em cada uma delas. A licença é um documento-chave, é aquele em que está contida e pode ser entendida a divisão entre dois grupos que propõem relações diferentes com o código e com o mundo: os defensores do software livre e os advogados do software proprietário. Quando um dos lados critica ou questiona o outro está, de forma direta ou indireta, fazendo referência às licenças.

A GPL, Licença Pública Geral, é a mais empregada das “livres”, embora não seja a primeira em seu grupo¹⁸. A EULA (*End User License Agreement*), Acordo de Licença para o Usuário Final, é a que rege, com algumas pequenas variações de acordo com os softwares, os produtos da Microsoft.

A relevância das licenças

Em um evento realizado na Universidade de Nova Iorque, em maio de 2001, o vice-presidente da Microsoft, Craig Mundie, fez comentários e críticas diretas à GPL, atitude que gerou respostas contundentes dos principais defensores do software livre. O título da palestra foi “The Commercial Software Model” e há um longo trecho que trata da GPL e do modelo de desenvolvimento de software que ela gera:

“The phrase “open source software,” or OSS, is often used as an umbrella term for a collection of product development, distribution and licensing practices, many of which have existed individually since the early days of computing. There are actually a number of different approaches within this community, but the common traits are providing people with access to source code and allowing others to modify and redistribute that code.

¹⁸ Outras licenças livres, como a do BSD, que coloca menos condições à alteração e redistribuição, assemelhando-se ao chamado “domínio público”, já existiam.

(...)

The OSS development model leads to a strong possibility of unhealthy "forking" of a code base, resulting in the development of multiple incompatible versions of programs, weakened interoperability, product instability, and hindering businesses' ability to strategically plan for the future. Furthermore, it has inherent security risks and can force intellectual property into the public domain.

Some of the most successful OSS technology is licensed under the GNU General Public License or GPL. The GPL mandates that any software that incorporates source code already licensed under the GPL will itself become subject to the GPL. When the resulting software product is distributed, its creator must make the entire source code base freely available to everyone, at no additional charge. This viral aspect of the GPL poses a threat to the intellectual property of any organization making use of it. It also fundamentally undermines the independent commercial software sector because it effectively makes it impossible to distribute software on a basis where recipients pay for the product rather than just the cost of distribution.

(...)

In contrast, two decades of experience have shown that an economic model that protects intellectual property and a business model that recoups research and development costs have shown repeatedly that they can create impressive economic benefits and distribute them very broadly.”¹⁹

No trecho acima, a crítica reside no fato de que os projetos desenvolvidos sob a GPL, por terem o código aberto, possível de ser lido e alterado por qualquer um, podem gerar diversos projetos paralelos (chamados por ele, usando um jargão da área, de “fork”). Esses forks são adjetivados como “unhealthy”, não saudáveis. Na prática, eles levariam, segundo o texto, os códigos dos programas, protegidos por direitos autorais, a tomarem o caminho do domínio público. Já o modelo de desenvolvimento que ele defende, o modelo proprietário, garantiria uma recompensa aos investimentos em pesquisa, e geraria grandes benefícios econômicos distribuídos a todos.

O modelo do software livre é chamado de *open source software*, ou software de código aberto. O modelo por ele defendido, o do software proprietário, é nomeado como “modelo do software comercial”. Esses nomes estão sendo tratados por esse sujeito como equivalentes. O impacto e o sentido da escolha de que nome utilizar será tratado em outro momento.

No mês de fevereiro, no mesmo ano, declarações de outro executivo da Microsoft, Jim Allchin,

¹⁹ “Prepared Text of Remarks by Craig Mundie, Microsoft Senior Vice President - The Commercial Software Model” em Microsoft, site institucional. Visualizado em 15/12/2004 <http://www.microsoft.com/presspass/exec/craig/05-03sharedsource.asp>

também geraram grande repercussão entre a comunidade envolvida com o software livre. Allchin afirmou que o software livre ameaça a propriedade intelectual e afirmou que sua empresa, até aquele momento, ainda não tinha feito o suficiente para mostrar isso àqueles que são responsáveis pelas políticas governamentais. A frase de Allchin, na formulação dada por uma reportagem, circulou intensamente pela internet:

“I'm an American, I believe in the American Way," he said. "I worry if the government encourages open source, and I don't think we've done enough education of policy makers to understand the threat.”²⁰

Um dos motivos de a frase ter sido reproduzida intensamente pode ter sido a identificação de algo que pretensamente contraria o *american way*. Em diversos sites, a frase foi reformulada para algo como: “a Microsoft diz que o software livre é algo não-americano”. O comentário abaixo é parte de um pequeno texto publicado no blog de um pesquisador (um biólogo interessado em software) e é uma reformulação desses dois comentários dos executivos da empresa e da declaração do executivo-chefe da Microsoft, Steve Ballmer, feita no mesmo período, e que comparou o software livre a um “câncer”²¹:

“Freely available source code undermines the commercial software business model based on selling programs as products that customers cannot modify or share. The GPL's "pass along" effect continually enlarges the code base GPL programmers can draw from when writing new programs. This is why Microsoft describes the licence as being "viral," "a cancer," and "unamerican"--it undermines the way they do business and keeps getting bigger.”²²

É interessante notar como o “ataque” verbal dos executivos da Microsoft ao software livre foi interpretado, por um autor brasileiro, como uma crítica à GPL. Na verdade, apenas o comentário de Mundie refere-se especificamente a características da licença, os comentários de Balmer referem-se ao

20 “Microsoft Executive Says Linux Threatens Innovation” em *News.com*, visualizado em 15/12/2004 <http://web.archive.org/web/20011201063013/news.cnet.com/investor/news/newsitem/0-9900-1028-4825719-RHAT.html>

21 Um desenvolvedor de software livre brasileiro reproduziu assim a fala de Ballmer: “Mais uma declaração polêmica do atual CEO da Microsoft, Steve Ballmer: “O Linux é um câncer que se apodera de tudo o que toca” (“Linux is a cancer that attaches itself in an intellectual property sense to everything it touches,”), dita durante uma entrevista ao Chicago Sun-Times.”. Morimoto, Carlos. “Microsoft: “O Linux é um câncer” (não seria o contrário??)” em Guia do Hardware.net. Visualizado em 15/12/2004 <http://www.guiadohardware.net/artigos/157/>. Mais adiante o autor revela que Ballmer estava se referindo a uma característica da GPL, que é exigir que os software derivados de software por ela licenciados também sejam GPL.

22 Ahmed, Zimran.”Viral software production” em Winterspeak.com. Publicado em 17/07/2001 e visualizado em 14/12/2004. <http://www.winterspeak.com/columns/071701.html>

Linux e os de Allchin dirigem-se mais amplamente ao modelo do software livre. GPL, Linux e software livre são entendidos como um só. Como Balmer, assim como Mundie, usou uma de metáfora médica, foi possível criar a seguinte formulação:

*“Agora foi a vez da GPL. Segundo a Microsoft, a GPL é anti-americana, “viral” e capaz de tornar todos os produtos de uma empresa “domínio público”. Não me surpreenderia se dissessem que ela também engorda e causa câncer em ratos de laboratório.”*²³

O impacto da associação do software livre (e da GPL) a algo “não-americano” ou “anti-americano” foi algo tão forte que provocou uma resposta do presidente da Free Software Foundation, Richard Stallman. Em sua resposta, ele retrabalha a declaração publicada de Allchin como um comentário à GPL para, em seguida, argumentar que a GPL está de acordo com o *american way* e baseada nos valores daqueles que lutaram pela independência dos EUA. Defender a GPL seria um ato de luta pela liberdade. E esta seria o cerne dos valores e dos ideais do movimento software livre.

Na história mais recente dos Estados Unidos, a palavra *unamerican* lembra o *House Committee on Un-American Activities* (HUAC), comissão instaurada no parlamento estadunidense que se notabilizou pelas investigações de atividades e propaganda comunista entre o final dos anos 1940 e início de 1950. Foi por sua atuação nessa comissão que o senador Joseph McCarthy inspirou o nome McCarthyismo, que é entendido como o período na história dos Estados Unidos marcado pela perseguição político-cultural, principalmente contra artistas, sob a alegação de ligações com o comunismo e com a União Soviética.

“O Movimento Open Source, que foi lançado em 1998, tem o objetivo de desenvolver softwares poderosos e confiáveis e uma tecnologia avançada convidando o público a colaborar com o desenvolvimento do software. Muitos desenvolvedores que participam desse movimento usam a GNU GPL, e são bem-vindos para utilizá-la. Mas as idéias e a lógica da GPL não podem ser encontradas no Movimento Open Source. Elas derivam dos objetivos e valores mais profundos do Movimento de Software Livre.

O Movimento de Software Livre foi fundado em 1984, mas sua inspiração vem dos ideais de 1776: liberdade, comunidade e cooperação voluntária. Isto é o que leva à

23 “Microsoft versus GPL”, em Revista do Linux. Ed.21. Visualizado em 14/12/2004.
<http://www.revistadolinux.com.br/ed/021/assinantes/opiniaio.php3>

livre empresa, à liberdade de opinião e à liberdade de software. Assim como em "livre empresa" e "livre opinião", o "livre" de "software livre" se refere à liberdade, e não a preço; especificamente, isso quer dizer que você tem a liberdade de estudar, mudar e redistribuir o software que utilizar. Essas liberdades permitem que cidadãos ajudem a si mesmos e uns aos outros, e dessa forma participem de uma comunidade. Isto estabelece um contraste com o software proprietário mais comum, que mantém os usuários indefesos e divididos: o funcionamento interno é secreto, e você está proibido de compartilhar o programa com seu vizinho. Um software poderoso e confiável e uma tecnologia avançada são subprodutos úteis da liberdade, mas a liberdade de ter uma comunidade é tão importante quanto."²⁴

Vale aqui retomar a noção de Político empregada por Guimarães para discutir as observações de Stallman. Ao afirmar que o movimento software livre representa sim os valores do *american way*, Stallman rediscute e ressignifica *american way*. Ao fazê-lo, procura significar a expressão como algo coerente com os princípios do software livre, enunciados na GPL, ao mesmo tempo em que trata o software proprietário como algo que mantém seus “usuários indefesos e divididos”, ressignificando, por oposição, também esse termo, como algo “não americano”.

Mais adiante, no mesmo texto, Stallman prossegue seu trabalho de ressignificação do *american way*. Em sua argumentação, em defesa do software livre e da GPL, ele procura explicar o sentido dos direitos e deveres estabelecidos por seu movimento e deixar claro o funcionamento de sua licença. Isso é tão importante quanto mostrar que esses princípios estão de acordo com o que é imaginado como “ser americano”:

“Não pudemos estabelecer uma comunidade de liberdade na terra do software proprietário onde cada programa tem seu senhor. Tivemos de construir uma nova terra no ciberespaço - o sistema operacional GNU de software livre, que começamos a escrever em 1984. Em 1991, quando o GNU estava quase terminado, o kernel Linux escrito por Linus Torvalds preencheu a última lacuna; em pouco tempo o sistema GNU/Linux livre estava disponível. Hoje, milhões de usuários utilizam o GNU/Linux e desfrutam dos benefícios de liberdade e comunidade.

(...)

A Microsoft utiliza uma estratégia anticompetitiva chamada "abraçar e estender". Isto significa que eles começam com a tecnologia que outros estão utilizando, adicionam uma pequena informação adicional que é secreta, de modo que ninguém mais pode

24 Stallman, Richard. “A GNU GPL e o Modo Americano de Viver”. Visualizado em 13/12/2004
<http://gnuweb.kookel.org/philosophy/gpl-american-way.pt.html>

imitá-la, e em seguida usam essa informação secreta de forma que apenas o software da Microsoft possa se comunicar com outro software Microsoft. Em alguns casos, isso torna difícil que você utilize um programa não-Microsoft quando outros com os quais você trabalha usam um programa Microsoft. Em outros casos, isso torna difícil para você usar um programa não-Microsoft para o trabalho A se você usa um programa Microsoft para o trabalho B. De qualquer modo, "abraçar e estender" amplia o efeito do poder de mercado da Microsoft.

Nenhuma licença pode impedir a Microsoft de praticar "abraçar e estender" se ela estiver determinada a fazê-lo às nossas custas. Se eles escrevem seu próprio programa do zero, e não utilizarem nada do nosso código, a licença sobre nosso código não os afetará. Mas uma reescritura total custa muito caro e é muito difícil, e mesmo a Microsoft não consegue fazer isso o tempo todo. Daí essa campanha deles para nos convencer a abandonar a licença que protege nossa comunidade, a licença que não os deixará dizer: "O que é seu é meu, e o que é meu é meu." Eles querem que nós os deixemos pegar o que quiserem, sem nunca revolver nada. Eles querem que abandonemos nossas defesas.

Mas ser indefeso não é o American Way. Na terra do bravo e do livre, defendemos nossa liberdade com o GNU GPL”

Nesse trabalho de resignificação do *american way*, Stallman procura demonstrar como o uso de softwares proprietários se assemelha a um aprisionamento, a uma dependência. Ao mesmo tempo, procura colocar a GPL e o software livre como os instrumentos de combate com o qual foi construída a “nossa terra no ciberespaço”. É possível desenhar o seguinte diagrama:

Microsoft = software proprietário = dependência = Ingleses

GNU GPL = software livre = liberdade = Patriotas²⁵

Esse tipo de comparação, usando elementos da história da independência dos Estados Unidos, já foi repetida outras vezes, por outros autores. Em dezembro de 2002, perguntado sobre os questionamentos legais da Microsoft ao software livre, Matthew Szulik, executivo-chefe da Red Hat, empresa que presta serviços usando GNU/Linux, usou de analogia semelhante, aqui reproduzida como discurso indireto em uma reportagem:

“Likening the Linux rising to the Boston Tea Party, in which the original Patriots resisted taxation without representation, the Red Hat CEO admonished Microsoft for crushing competition.”²⁶

25 Patriotas ou *Patriots* foram os estadunidenses que lutaram contra o domínio britânico

26 Rooney, Paula. “Red Hat CEO 'Scared' About Microsoft's Legal Tactics To Crush Linux” em *CRN*. Publicado em 03/12/2002. Visualizado em 14/12/2004

A Red Hat é a principal empresa estadunidense que trabalha com software livre. A adaptação do GNU/Linux que criou é uma das mais usadas nos EUA e em todo mundo. Seus executivos não são conhecidos por serem radicais no discurso em defesa da liberdade no software livre, pelo contrário, são ligados aos setores mais moderados (o movimento código aberto). O evento a que ele alude, a Boston Tea Party, ocorreu em 1773, três anos antes da independência dos EUA e foi um marco da resistência organizada e pacífica ao domínio econômico da Inglaterra. O acontecimento, no entanto, foi chave para o processo que levou à separação política das trinta colônias norte-americanas do Império Britânico.

Note-se que o representante do um setor mais conservador no escopo do movimento software livre exemplifica a partir de um fato do passado político que lembra uma resistência pacífica enquanto, por sua vez, um representante de um setor mais radical, faz lembrar o momento logo após a guerra da Independência. Cada um ancora o seu discurso em certos fatos da memória política, e não outros, que falam à sua posição com relação ao enfrentamento entre as partes (software livre Vs. software proprietário), mas também à posição que cada um ocupa no próprio movimento pelo software livre.

Nas licenças, as diferenças

Para esta análise, será feito o uso de duas licenças tomadas como modelo de uma licença livre e uma licença proprietária: a GPL, em sua versão oficial traduzida para o português, e a EULA do Microsoft Windows 95. A GPL é apenas uma das várias licenças livres existentes. Contudo, é ela que registra a maior parte dos softwares sistema operacional GNU/Linux. Para esta análise, será utilizada a CC GNU-GPL²⁷, fruto de um projeto que reuniu: o Instituto Nacional de Tecnologia da Informação (ITI), autarquia da Casa Civil do governo brasileiro; a Free Software Foundation (FSF); o autor da versão original em inglês da GPL, Richard Stallman; advogados brasileiros da Fundação Getúlio

http://www.crn.com/sections/breakingnews/dailyarchives.jhtml;jsessionid=DGZPQX2WLYB1OQSNDBCCKHSCJUM EKJVN?articleId=18822143&_requestid=669262

27 <http://www.softwarelivre.gov.br/Licencas/LicencaCcGplBr/view>

Vargas (FGV); e membros da Creative Commons, entidade sediada nos Estados Unidos e que busca dar um escopo legal para uma flexibilização dos direitos autorais. As licenças da Creative Commons, criadas para obras artísticas e jornalísticas, baseiam-se nos princípios das licenças livres (liberdade para copiar, executar, distribuir e alterar) e estão sempre disponíveis em três formatos: um resumido, para leitura rápida, escrita em “linguagem simplificada” - no original em inglês “*human-readable commons deed*” -; outra a “parte jurídica” - “*lawyer-readable GNU GPL*”, no original em inglês -; e outra um “código de leitura pela máquina” - “*machine-readable digital code*”, no original em inglês – um código em linguagem html, lido pelos navegadores da internet e disponibilizado para que seja inserido nas páginas daqueles que licenciam seu softwares²⁸. Em geral, os programadores que licenciam seus softwares pela GPL apenas anexam a seu código um arquivo texto contendo o original dessa licença em inglês. Para efeito desta análise, será utilizada a versão em português da “parte jurídica” da licença²⁹, que é uma tradução da mais recente versão em inglês, modificada pela Free Software Foundation em 1991³⁰.

No mundo proprietário, a diversidade das versões das licenças é maior. Como esses softwares são de propriedade de empresas, estas em geral tem um departamento jurídico responsável por definir variações de licenciamento pra cada produto. Será utilizada a licença do Windows 95³¹ por ser ainda a plataforma padrão para boa parte dos computadores do mundo. Entretanto, a licença do Windows XP, a mais nova versão do sistema operacional da Microsoft também foi examinada. Em comparação com a licença do Windows 95 não foram encontradas diferenças substanciais, apenas restrições específicas em relação a certos usos, como a restrição a que outros computadores se conectem ao computador onde o sistema operacional foi instalado ou a necessidade de ativação do sistema junto ao fabricante em

28 A disponibilização da licença em “três versões” é uma prática da Creative Commons que, internacionalmente, ocupa-se em licenciar prioritariamente audiovisuais (textos, imagens, vídeos). A existência em “três versões” da GPL no Brasil, caso único no mundo, deve-se ao fato de a Creative Commons ter participado do processo de adequação à leis brasileiras.

29 Anexo a. Retirado de <http://creativecommons.org/licenses/GPL/2.0/legalcode.pt>

30 <http://www.gnu.org/licenses/gpl.txt>

31 Anexo b. Retirado de <http://www.cipsga.org.br/sections.php?op=viewarticle&artid=278>

determinado prazo. Contudo, essas diferenças não são relevantes ao propósito desta análise.

As diferenças entre a GPL e a EULA começam pela própria linguagem empregada. Ambas são instrumentos jurídicos, textos tidos como de leitura difícil, não muito agradável. Mas o texto da GPL distancia-se desse modelo. Nele, quem fala não é uma entidade jurídica, sempre na terceira pessoa, mas o coletivo. A terceira pessoa do plural é sempre usada – “queremos proteger...”, “queremos evitar...” - seja literalmente ou de forma implícita. Antes de qualquer norma, ela coloca explicações, justifica as restrições que estabelece. “Para proteger seus direitos, necessitamos fazer restrições que proíbem que alguém negue esses direitos a você ou que solicite que você renuncie a eles”, diz. Com isso, quebra-se o efeito de impessoalidade do texto jurídico comum.

Mas ambos, tanto a GPL como a EULA são contratos jurídicos de licenciamento. Ambos regulam a concessão de direito de uso a partir de uma mesma via legal. No entanto a GPL, colocando-se como uma negação às restrições das licenças proprietárias, parece negar também sua característica jurídica. Em nenhum momento, por exemplo, a palavra contrato é citada como referência ao próprio documento.

a. A EULA

Já a EULA enfatiza seu caráter de um instrumento legal. Os itens são dispostos em cláusulas e a primeira frase do texto é um aviso em letras garrafais: “**IMPORTANTE – LEIA COM ATENÇÃO**”. O recurso à caixa alta é empregado diversas vezes no texto, para marcar o início de cada item/cláusula ou para marcar certas palavras. Essas palavras são:

EULA – SOFTWARE – COMPUTADOR

A palavra “**PRODUTO**” também é grafada em caixa alta logo no primeiro parágrafo mas, como é enunciada como sinônimo de “**SOFTWARE**”, não é mais utilizada com essa referência. Também são

referidos com caracteres especiais as duas partes do contrato, aquele que adquire a licença, referido como “V.Sa”³², e empresa que concede a licença, referida como “Fabricante”.

O nome da empresa proprietária dos direitos sobre o sistema, a Microsoft, é referido em algumas situações de maneira ambígua. À exceção do título do contrato, grafado em caixa alta (“CONTRATO DE LICENÇA DE USUÁRIO FINAL PARA SOFTWARE MICROSOFT”), na primeira vez em que o nome da empresa é referido é para qualificar, marcar o produto: “...o(s) programa(s) de computador Microsoft.”. Na segunda vez, ela aparece como entidade distinta da figura do “Fabricante”. A formulação da frase é: “Caso V.Sa. não esteja de acordo com os termos deste EULA, o Fabricante e a Microsoft não licenciarão o SOFTWARE para V.Sa.”. Assim, ela aparece de duas formas: como marca que acompanha o nome do produto e como licenciadora do software, não se assimilando ao “Fabricante”.

Outra situação em que o nome da empresa aparece não apenas como nome auxiliar da marca do produto que está licenciado é no sub-item “Rescisão”, do item “DESCRIÇÃO DE OUTROS DIREITOS E LIMITAÇÕES”. O texto do sub-item é o seguinte: “Sem prejuízo de quaisquer outros direitos, a Microsoft poderá rescindir este EULA caso V.Sa. não cumpra com seus termos e condições. Neste caso, V.Sa. deverá destruir todas as cópias do SOFTWARE e de seus componentes.” Aqui, a empresa aparece como entidade ativa, com poderes para vetar a continuidade do uso do software se as condições estabelecidas não forem cumpridas.

Na última vez em que nome da empresa é referido não somente como assemelhado ao produto, é para isentá-la da obrigação de oferecer algum apoio ao usuário. Nesse momento, ela novamente se distancia da figura do “Fabricante” e é a única oportunidade em que é enunciado o nome completo da empresa: “Microsoft Corporation”. O trecho seguinte refere-se ao item de número 6, “SUPORTE AO PRODUTO”:

³² Na versão mais recente do produto, na licença do Windows XP, o pronomes de tratamento para o comprador da licença é alterado para “você”.

“O suporte ao produto para o SOFTWARE não é prestado pela Microsoft Corporation ou suas subsidiárias. Para obter informações quanto ao suporte ao produto, favor consultar o número de suporte do Fabricante indicado na documentação do COMPUTADOR. Caso V.Sa. tenha quaisquer dúvidas relativas a este EULA ou deseje contatar o Fabricante por qualquer outra razão, utilize o endereço indicado na documentação do COMPUTADOR.”

Cabem aqui algumas observações com relação ao efeito dessa divisão entre “Microsoft” e “Fabricante”, esse jogo em que uma das partes do contrato ora é descrita como um ora como outro. Ao assumir uma identidade distinta da figura do “Fabricante”, no contrato, a empresa torna-se parte dele de uma maneira indireta. Ao mesmo tempo em que surge como detentora de uma propriedade, que não é vendida mas sim licenciada³³, ela desaparece como entidade com a qual o contratante está realizando uma operação de licenciamento. As partes enunciadas como contratantes são “V.Sa.” e o “Fabricante”. Se “V.Sa.” não concordar com a EULA não será à Microsoft a quem deverá recorrer, mas sim ao “Fabricante”, pois está posto logo no início da licença: “Neste caso, V.Sa. não poderá utilizar ou copiar o SOFTWARE e deverá contatar imediatamente o Fabricante para obter instruções sobre como devolver o(s) produto(s) não utilizados e para receber um reembolso.”

Com isso, tem-se um acordo que se estabelece legalmente entre duas partes, como enunciado pelo contrato, mas em que uma dessas partes pode se dividir em duas, “Fabricante” e “Microsoft”. Para a última, cabe: o papel de romper o acordo se a parte licenciada infringir alguma cláusula; e se isentar no momento de oferecer alguma ajuda a essa parte licenciada. Para a primeira, esse ente sem rosto, nomeado genericamente como “Fabricante”, cabem algumas poucas responsabilidades como atender o licenciado quando esse procurar por suporte.

Se pensarmos o político aqui como já definido acima, é possível dizer que o que a empresa faz é estabelecer uma divisão do real em que condições desiguais são normatizadas. Para que o licenciado, aquele que não detém aquele produto, possa usufruir dele será preciso que pague uma determinada taxa, o custo para obter o direito de uso. Porém, como não é a propriedade daquele código que está sendo

³³ “Este SOFTWARE é protegido pelas leis de copyright e tratados internacionais, bem como por outras legislações e tratados sobre propriedade intelectual. O SOFTWARE é licenciado e não vendido.”, diz o preâmbulo da primeira seção do contrato, intitulada: “LICENÇA DO SOFTWARE”.

adquirida apenas o uso é concedido, o proprietário estabelecerá também certas restrições para esse uso, condição essa que, se violada, implicará em quebra do contrato. Os licenciadores colocam condições bastante claras, presentes no contrato: aceitar ou desistir, apagar e devolver o produto.

A inclusão desse tipo de pré-condição para o uso, “aceite ou desista” é em si uma demonstração dessa desigualdade entre as partes, ela só pode estar presente porque uma das partes estabeleceu unilateralmente as condições a serem aceitas em sua totalidade. E essas condições unilaterais só podem existir porque o contrato está sendo estabelecido entre partes não-iguais, entre proprietários, donos, possuidores e meros usuários licenciados.

Para que se faça uso do software proprietário é preciso aceitar a licença e este ato, a aceitação do contrato, é um ato de reconhecimento da distinção entre licenciador e licenciado, que reconhece a existência da figura do proprietário. Mesmo que a EULA licencie um software para o qual o detentor de direitos não estabeleça uma taxa para o uso naquele momento – o que pode ser o caso do sistema operacional Windows oferecido de graça a escolas, por exemplo – o usuário precisa concordar que o direito de uso é algo concedido por outro e pode ser revogado no futuro.

Esse “compromisso” entre as partes é estabelecido, no caso do Windows, no momento da instalação do software. Mesmo que a cópia disponível ao usuário tenha sido adquirida ilegalmente, a licença implica no reconhecimento de uma infração legal. Ela faz o que diz, estabelece uma propriedade, mesmo quando o usuário viola os termos por ela estabelecidos, pois este torna-se passível de punição legal, já que a licença existe.

b. A GPL

Isso ocorre de maneira distinta com a GPL. Como não há restrições à execução e à cópia do programa, não é requerido ao usuário que aceite a licença para isso. Porém, este precisa concordar com os termos postos pela GPL se deseja distribuir ou modificar o software. Mas, a proposta de adesão a

essas condições nunca é formulada como uma condição imposta, não negociável. O item de número 5, da seção “TERMOS E CONDIÇÕES PARA CÓPIA, DISTRIBUIÇÃO E MODIFICAÇÃO”, que pede a concordância com os termos da licença não se utiliza da formulação “você deverá”, como na EULA. Pelo contrário, a não adesão aos termos aparece como um direito. A primeira frase estabelece esse direito. A segunda, mostra como a aceitação dos termos é o único meio para que o direito à redistribuição e à alteração (direitos inacessíveis pela licença proprietária) do software sejam concedidos. Apenas na terceira frase a legislação é citada e, por consequência, aquilo tudo que essa palavra refere na memória discursiva, instituições como o Direito, a Justiça e as punições que cabem a estes estabelecer. A formulação é a seguinte:

“Você não é obrigado a aceitar esta Licença, uma vez que você não a assinou. Porém, nada mais concede a você permissão para modificar ou distribuir o Programa ou respectivas obras derivativas. Tais atos são proibidos por lei se você não aceitar esta Licença. Conseqüentemente, ao modificar ou distribuir o Programa (ou qualquer obra baseada no Programa), você estará manifestando sua aceitação desta Licença para fazê-lo, bem como de todos os seus termos e condições para copiar, distribuir ou modificar o Programa ou obras nele baseadas.”

A GPL é uma licença que funciona em outro nível. Mesmo que o usuário não a aceite, ele não será impedido de utilizar o software. O que ela estabelece não é algo que se restringe a um contrato entre licenciador e licenciado. Ela fala de uma terceira pessoa que não está no contrato, ela coloca restrições à relação do licenciado (licenciante naquele momento) com os novos licenciados que surgirão no futuro. Impede que o licenciador assuma uma condição diferente do licenciado, de modo que possa estabelecer a outros restrições maiores dos que as de que foi objeto.

A GPL pode ser dividida em duas partes: uma “Introdução”, que antecede as cláusulas da licença, e os termos propriamente ditos, em que figuram: os itens a serem observados; uma “EXCLUSÃO DE GARANTIA”, escrita em letras maiúsculas; e a sub-seção “FINAL DOS TERMOS E CONDIÇÕES”, em que o licenciado é ensinado a aplicar a licença nos softwares a que vier a modificar. A sub-seção “EXCLUSÃO DE GARANTIA” é o único momento em que as letras

maiúsculas são usadas no texto mas, mesmo assim, o que ela estabelece, a exclusão da garantia, é algo que é objeto de justificação, de explicações³⁴.

A “Introdução” é o trecho do texto que soa menos parecido com um contrato jurídico. Ela lembra mais uma pequena dissertação, com uma justificativa para os direitos e deveres estabelecidos pela licença. Desde o início ela coloca-se como uma alternativa, como algo que não pertence ao mundo jurídico dos contratos tradicionais para o uso de programas de computador. Diz o primeiro parágrafo:

“As licenças da maioria dos softwares são elaboradas para suprimir sua liberdade de compartilhá-los e modificá-los. A Licença Pública Geral do GNU, ao contrário, visa garantir sua liberdade de compartilhar e modificar softwares livres para assegurar que o software seja livre para todos os seus usuários. (...) Você também poderá aplicá-la aos seus programas.”

A licença coloca a si mesma como um instrumento para a garantia de direitos, direitos estes que vão além do contrato a ser estabelecido, que extrapolam o escopo de um acordo jurídico. Ela tem um fim que não se esgota na relação entre o licenciador e o licenciado naquele momento, deve ser usada como instrumento para a garantia da liberdade de modificação e alteração do software dos futuros licenciados. Não só aquele licenciado naquele momento, mas a licença foi criada para ser usada exemplarmente por outros desenvolvedores em suas criações.

Nesse sentido, ela se coloca como meio para que desiguais (aqueles que não são proprietários do software) possam se igualar aos “senhores” dos programas – para usar a metáfora estabelecida por Stallman no texto analisado no item 2 deste trabalho, ou seja, aqueles capazes legalmente de evitar a transformação dos softwares de que são proprietários- e também participem da comunidade capaz de distribuir, melhorar e adaptar os programas.

Logo após esse parágrafo inicial, já comentado, o que se segue é uma explicação justificada do

³⁴ *“11. COMO O PROGRAMA É LICENCIADO SEM CUSTO, NÃO HÁ NENHUMA GARANTIA PARA O PROGRAMA, NO LIMITE PERMITIDO PELA LEI APLICÁVEL. EXCETO QUANDO DE OUTRA FORMA ESTABELECIDO POR ESCRITO, OS TITULARES DOS DIREITOS AUTORAIS E/OU OUTRAS PARTES, FORNECEM O PROGRAMA “NO ESTADO EM QUE SE ENCONTRA”, SEM NENHUMA GARANTIA DE QUALQUER TIPO, TANTO EXPRESSA COMO IMPLÍCITA, INCLUINDO, DENTRE OUTRAS, AS GARANTIAS IMPLÍCITAS DE COMERCIALIZIDADE E ADEQUAÇÃO A UMA FINALIDADE ESPECÍFICA. O RISCO INTEGRAL QUANTO À QUALIDADE E DESEMPENHO DO PROGRAMA É ASSUMIDO POR VOCÊ. CASO O PROGRAMA CONTENHA DEFEITOS, VOCÊ ARCARÁ COM OS CUSTOS DE TODOS OS SERVIÇOS, REPAROS OU CORREÇÕES NECESSÁRIAS.”*

que significa a licença e seu funcionamento. Desde logo, procura-se resolver uma controvérsia sobre o sentido da palavra *free* do original em inglês. É dito, então, que a palavra *free* é usada como sinônimo de liberdade e não como gratuidade inerente ao produto. As implicações e a controvérsia sobre esse termo serão examinadas aqui em outro momento.

No terceiro parágrafo, acontece uma inversão bastante interessante. Em geral, um contrato de licença é estabelecido tendo-se em vista o direito do proprietário do software, que concede alguns benefícios ao licenciado em troca de uma compensação financeira – e o licenciado também adquire algumas garantias. No caso da GPL, nesse trecho de introdução da licença, o foco são os direitos do licenciado e que, como já dito, vão além do produto software em questão.

“Para proteger seus direitos, necessitamos fazer restrições que proíbem que alguém negue esses direitos a você ou que solicite que você renuncie a eles. Essas restrições se traduzem em determinadas responsabilidades que você deverá assumir, se for distribuir cópias do software ou modificá-lo.

(...)

Protegemos seus direitos através de dois passos: (1) estabelecendo direitos autorais sobre o software e (2) concedendo a você esta licença, que dá permissão legal para copiar, distribuir e/ou modificar o software.”

A licença se coloca mediando não apenas a relação entre licenciador e licenciado, mas a relação do licenciado com o mundo externo. Ela está intercedendo junto ao desigual de modo a garantir a ele uma igualdade que pode ser ameaçada. Ao fazer isso, ela afirma tanto a união entre licenciador e licenciado como o direito de ambos de realizarem operações que só os “donos” dos programas seriam capazes (alterar, modificar e distribuir). Autor e usuário se equivalem, tem os mesmos direitos sobre o software desde que garantam esses direitos também a outros. Aqui, não é importante o que ela faz ou não faz realmente, se ela efetivamente consegue garantir os direitos do licenciado frente ao exterior, mas o que ela diz fazer, algo que pode ser verificado na materialidade do texto.

Essa conjunção licenciador-licenciado está marcada em outros trechos do texto. O primeiro parágrafo da segunda parte, “TERMOS E CONDIÇÕES PARA CÓPIA, DISTRIBUIÇÃO E

MODIFICAÇÃO”, serve para, como é usual nos contratos, enunciar as partes contratantes:

“0. Esta Licença se aplica a qualquer programa ou outra obra que contenha um aviso inserido pelo respectivo titular dos direitos autorais, informando que a referida obra pode ser distribuída em conformidade com os termos desta Licença Pública Geral. O termo "Programa", utilizado abaixo, refere-se a qualquer programa ou obra, e o termo "obras baseadas no Programa" significa tanto o Programa, como qualquer obra derivada nos termos da legislação de direitos autorais: isto é, uma obra contendo o Programa ou uma parte dele, tanto de forma idêntica como com modificações, e/ou traduzida para outra linguagem. (Doravante, o termo "modificação" inclui também, sem reservas, a tradução). Cada licenciado, doravante, será denominado "você".”

Embora exista a figura do “titular dos direitos autorais”, este não é marcado com um termo específico a ser estabelecido como uma das partes. Só há dois entes sublinhados: “você”, o licenciado (grafado sem letra maiúscula); e o software objeto de licenciamento, denominado “Programa” (iniciado por letra maiúscula).

Mais restrições

Mas acho interessante também ressaltar certas diferenças entre as licenças que marcam essa situação de desigualdade entre proprietário e licenciado. A licença proprietária coloca condições que vão além do mero uso do software. As restrições colocadas pela EULA são tantas, em especial em sua redação para o sistema operacional mais recente, o Windows XP, que é provável que muitos usuários violem seus termos sem saber, o que coloca, mesmo aqueles que adquirem a licença – e não usam software pirata – na ilegalidade. Quem compra uma caixinha com Windows, ou adquire um computador com o sistema pré-instalado, tem até trinta dias para “ativar” o produto. Isso significa enviar à Microsoft, por telefone ou pela internet, algumas informações sobre o usuário.

Logo quando o software é instalado, é pedido que o usuário aceite as condições da EULA do Windows XP. Instalar o sistema operacional ou ativá-lo significa concordar com seus termos e um deles diz: “Você concorda que a MS, a Microsoft Corporation e suas afiliadas podem coletar e usar

informações técnicas recolhidas de qualquer forma como parte do suporte oferecido a você, se houver algum, relacionado ao software.” Não é dito o que pode e o que não pode ser considerado informação técnica.

A mesma cláusula ainda estipula que essa informação pode ser usada pela Microsoft ou suas afiliadas “somente para a melhoria dos produtos ou para oferecer serviços e tecnologias adaptados para você”. Ou seja, a empresa exige recolher dos usuários informações que serão úteis para a melhoria de seus produtos – e, portanto, proporcionarão à empresa melhorar sua participação no mercado. Mas ela não oferece nada em troca, ela obtém informações do licenciado que serão utilizadas para acentuar a distância entre as duas partes, a proprietária e a não-proprietária..

E há ainda as limitações para a instalação. Enquanto os software livres podem ser instalados em quantos computadores e em quantas máquinas o usuário quiser, a EULA, na sua versão típica, para o usuário doméstico, limita a instalação a apenas uma máquina. Cada pacote de software deve se ligar a um único computador. Se o computador for comprado com o sistema operacional já instalado, este só poderá ser vendido acompanhado em conjunto com a máquina.

As limitações atingem também os periféricos que poderão ser instalados e os computadores em rede que poderão fazer uso de serviços do sistema operacional. Algumas versões da EULA do Windows limitam o número de processadores (chips centrais) a serem usados no computador que o software está instalado. A EULA do Windows XP Home Edition limita a cinco o número de aparelhos que podem estar ligados ao computador.

As limitações impostas pelas licenças proprietárias não existem por acaso, assim como as liberdades do software livre. Elas resumem diferentes modelos de negócio. No modelo proprietário, uma empresa desenvolve um produto, com um certo gasto, e consegue multiplicar ao infinito seus lucros, pois vende cópias de algo que pode ser reproduzido, o código. O modelo de negócio livre é outro. Afinal, o programa foi produzido cooperativamente, por uma comunidade e deve permanecer

livre para todos, podendo ser copiado sem restrições.

Conclusão

Lançar um olhar atento para as características discursivas das licenças de software nos ajuda a entender como, na regulação e na defesa (ou ataque) de um sistema de funcionamento de um objeto técnico, o software, podem-se entrever concepções ideológicas sobre como devem se estabelecer certas relações humanas. Eivado de política, esse embate discursivo opõem proprietários, donos de direitos exclusivos, e “coletivistas”, autores que abdicam de seus direitos exclusivos de modificação, venda e distribuição, para torná-los direitos de todos aqueles que também aceitam contribuir com seu trabalho³⁵.

Algumas marcas encontradas no discurso dever ser retomadas aqui. A mais importante delas parece ser a tentativa de nivelamento, de estabelecimento de igualdade colocada pela GPL. Ao permitir o uso irrestrito, estabelecendo uma autorização prévia e extensiva a qualquer usuário, ela coloca o produto software como um bem comum.

Já ao permitir alterações, modificações e a livre redistribuição, essa licença denuncia a existência de uma desigualdade fundante, entre os que têm e os que não têm. Com a condição de que não haja tentativas de reestabelecimento dessa desigualdade, de apropriação daquilo que ela propõem que seja de todos, ela postula o direito à modificação do código, em última instância, ao trabalho, seja permitido a todos.

Penso que é principalmente ao oferecer a liberdade para a alteração e distribuição do código a qualquer um que a licença livre afirma que a desigualdade não é condição necessária à vida social. Ao fazer isso, a licença estabelece que não é necessário que usuários distingam-se dos produtores, que haja

³⁵ Alguns autores já buscam classificar o software livre não como um bem comum, mas como “conhecimento livre”. Saraiva, por exemplo, afirma que os bens comuns são compartilhados por todos, como as praças ou a água. O conhecimento, por ser intangível e multiplicável infinitamente (se tenho uma idéia e passo-a a alguém não deixo de tê-la e outro também a terá, assim como um software pode ser copiado infinitamente). Por isso, pede por uma distinção entre bens comuns e conhecimento livre. Saraiva, Diego e outros. “El conocimiento: ¿libre, común, privado o disperso?”, 10/01/2005. <http://bo.unsa.edu.ar/docacad/softwarelibre/articulos/libreocomun/>

sujeitos em condições desiguais com relação à posse do meio de produzir mercadorias.

Não é à toa que a licença livre, ou o movimento social que a utiliza, são classificados por aqueles que pertencem à esfera da norma estabelecida, da desigualdade que não deve se tornar aparente. Não só a denúncia da desigualdade como a tentativa de se construir uma outra ordem, são ligadas à anormalidade, às patologias, classificados como “vírus” ou “câncer”. Também não é gratuito que as reações à essas afirmações tenham sido inflamadas e partido de diversos lugares. Um ponto central foi tocado, a existência da desigualdade foi negada e os desiguais foram tratados como não pertencentes a esse mundo.

Da mesma forma, o adjetivo “*unamerican*”³⁶ foi amplamente replicado como tendo sido utilizado pela Microsoft para classificar o movimento software livre. Uma declaração como esta funcionou como uma afirmação de não pertencimento, de exclusão do que seriam os valores essenciais de uma nação. O próprio presidente da Free Software Foundation, Richard Stallman, tratou de responder a essa afirmação e o fez procurando estabelecer um certo sentido do que é ser americano. Para fazer isso, buscou a memória da trajetória política de seu país, afirmou certos valores que consistiriam na essência histórica fundadora da nação.

No Brasil, o termo *unamerican* recebeu duas traduções: “não-americano”³⁷ e “anti-americano”. Existe uma divergência de sentido entre os dois termos. Enquanto o primeiro opera significando não integrar aquela nação, seja culturalmente, seja socialmente, o termo “anti-americano” lembra oposição, contrariedade, não estabelece apenas uma diferença, mas uma antítese.

Mais de uma vez Stallman já foi tachado de comunista³⁸ e ele é conhecido pela sua militância política em defesa dos direitos civis e outras causas sociais. Por isso respondeu, enfaticamente, a essa

36 O' Reilly Tim,. “Is Open Source Un-American?” em ONLamp.com. Publicado em 08/03/2001 e visualizado em 14/12/2004. <http://www.onlamp.com/pub/a/onlamp/2001/03/08/unamerican.html>

37 “A vez do desktop”, em *Revista do Linux* . Visualizado em 14/12/2004. <http://www.revistadolinux.com.br/ed/019/assinantes/capa.php3>

38 “Interview: Richard M. Stallman“. Visualizado em 14/12/2004 <http://gnu.open-mirror.com/philosophy/luispo-rms-interview.html>

segunda concepção do termo. Ao ressignificar o “*american way*” como os valores que levaram à independência dos EUA frente ao domínio britânico, Stallman agiu sobre a memória do sentido de ser americano, reforçando uma certa passagem da história em lugar daquela que opôs, durante a Guerra Fria, o sistema comunista e o sistema capitalista, e que também é lembrada pelo termo “*unamerican*”.

Cabe aqui, também, discutir um pouco sobre o caráter performativo dessas licenças. Embora a GPL diga existir para garantir liberdades, estas não se referem ao licenciado que, naquele momento, tem a licença em suas mãos (ou está usando um software licenciado pela GPL). Esse licenciado, se a licença simplesmente não existisse, já teria garantida a liberdade de alteração e distribuição pois um software não registrado não tem dono, pode ser apropriado por qualquer um. A GPL existe para garantir que aquele software é um bem coletivo e que nunca pode tornar-se de propriedade exclusiva.

Enquanto a EULA está lá para marcar o usuário como possível infrator de leis (se não tiver adquirido a cópia do software legalmente), a GPL existe para impedi-lo de se alçar à condição de “senhor”, para evitar que seja restabelecida a distinção entre usuários e donos exclusivos dos direitos autorais. Ela se coloca como uma negação das licenças proprietárias, respondendo à existência da desigualdade.

Concordar com a GPL (aceitá-la) significa se comprometer a nunca restabelecer essa desigualdade entre desenvolvedores e usuários, nunca impedir que este atinja a condição daquele.

Capítulo. 3 GNU/Linux ou Linux? Software livre ou código aberto?

Na comunidade software livre, existe uma constante disputa sobre os nomes e as palavras utilizadas. O próprio nome da comunidade é controverso. Enquanto uns preferem software livre outros falam em código aberto. Alguns, nesse caso não muitos, dizem que tanto faz e que é possível até mesmo somar os nomes³⁹. O nome do principal sistema operacional livre também é objeto de controvérsias, e aqui sim muitos afirmam que são sinônimos. Enquanto alguns se referem a ele simplesmente como Linux, outros fazem questão de dizer GNU/Linux⁴⁰. Essa discussão, às vezes, torna tudo muito mais confuso para quem não participa do debate. Seria muito mais simples se houvesse algum tipo de uniformização nesse vocabulário, sugerem alguns, o leigo poderia tomar parte dele muito mais facilmente.

Mas optar por um nome ou por outro não é algo trivial. Dizer é se colocar no mundo, é assumir

39 Perguntado do motivo do uso da sigla SLCA para designar software livre de código aberto, o diretor de uma ONG brasileira responsável por programas de inclusão digital respondeu da seguinte maneira: “É isso mesmo, Rafael. Não há deslize ideológico em chamar de soft livre e de código aberto - é apenas para assegurar que fique claro que não só a distribuição do código-objeto é livre, mas este tem que ser sempre acompanhado do código-fonte.”

40 É freqüente que, em listas de discussão sobre software livre, alguns membros – em geral os mais novos – publiquem mensagens como à seguir. Em geral, eles, já cientes das diferenças técnicas, consideram que o termo Linux deveria absorver definitivamente o termo GNU. A mensagem abaixo foi coletada em uma lista de discussão venezuelana (país cujo governo federal, nos últimos meses, tem aprofundado suas relações com o software livre) e cujo tema definidor é “onde o novato é respeitado:

“Hola a todos.

¿No les parece que es un poco tonto insistir en decir GNU-Linux en lugar de solo decir Linux?

Es como si dijéramos Microshaft Winblows, cuando todos decimos solo Winblows; o como si dijéramos M\$ D.O.S., cuando en realidad solo decimos D.O.S.; o como si dijéramos MApple MAC-OS, cuando solo decimos MAC-OS...

Ya se que me van a decir que deben aclarar que Linux no solo lo hizo Linus Torvals, etc, etc y todo lo demas tambien, pero ¿no les parece que es redundante, como si dijéran esta cayendo lluvia mojada del cielo?.

No lo digo con la intencion de ofender, solo lo comentaba.

Mata ne.

NOTA: Todas las marcas aqui mencionadas son propiedad de los chivos que cobran por ellas ;)”

posição. Afinal, há alguma diferença entre falar Linux ou GNU/Linux? Ou entre se dizer um adepto do movimento pelo software livre ou do movimento de código aberto? Há sim. A opção por um nome ou por outro marca, para além de possíveis diferenças técnicas, a posição discursiva ocupada pelo sujeito em relação à história do movimento.

De acordo com Orlandi, “ao falarmos nos filiamos a redes de sentidos mas não aprendemos como fazê-lo, ficando ao sabor da ideologia e do inconsciente. Por que somos afetados por certos sentidos e não outros? Fica por conta da história e do acaso, do jogo da língua e do equívoco que constitui nossa relação com eles. Mas certamente o fazemos determinados por nossa relação com a língua e com a história, por nossa experiência simbólica e de mundo, através da ideologia.”⁴¹

Para os fins deste trabalho, está sendo considerado como o “movimento software livre” uma gama ampla de grupos sociais que advogam as quatro liberdades essenciais para os usuários dos softwares que produzem/incrementam/distribuem: execução, alteração, distribuição e cópia. Nesse sentido, o movimento código aberto é um desses grupos. Alguns pesquisadores, para designar o movimento de forma mais ampla, tem usado o termo “Free/Libre/Open Source Software (FLOSS)”⁴². Como, no Brasil, o termo “código aberto” é pouco utilizado – como veremos mais tarde – a opção aqui é pelo termo “software livre”.

Duas correntes. A mesma luta?

Há alguns momentos que são fundantes na história do movimento software livre. É possível dizer que o marco inicial se dá em março de 1985, quando Richard Stallman escreve o Manifesto GNU, documento que desenha os princípios do copyleft⁴³, que dará base para as regras descritas na GPL e que

41 Orlandi, 2003. p.34

42 Um desses grupos integra o Stanford Institute for Economic Policy Research (SIEPR)

43 Copyleft é um termo criado para se opor ao copyright e foi criado por Richard Stallman. Segundo ele, a idéia veio de um colega que grafou: “Copyleft, all rights reversed”, fazendo um trocadilho com o termo e com a frase “all rights reserved” que acompanha o copyright. O termo também é interpretado como uma alusão ao espectro da esquerda na política.

é um convite para que outros programadores se unam ao esforço da Free Software Foundation de produzir um sistema operacional livre. Outro ano importante é 1991, quando, em setembro, Linus Torvalds lança a primeira versão do kernel Linux, que completou o sistema projetado pela FSF antes de ela mesma o fazer. Embora siga os princípios da GPL, o Linux significou, na prática, o surgimento de uma nova corrente de força dentro do movimento, que culminará com o surgimento do movimento código aberto (*open source*), em 1998. Neste ano, Eric Raymond publicou o artigo “Goodbye, “free software”; hello, “open source”” e fundou, com Bruce Perens, a Open Source Initiative⁴⁴

É interessante acompanhar a história para entender algumas questões. Ninguém nega que tudo saiu das mãos e da cabeça do guru Richard Stallman que, ainda na década de 1980, delineou os princípios éticos. Na época, Stallman, fundador da Free Software Foundation (FSF, Fundação do Software Livre, em inglês), estabeleceu as quatro liberdades que fundamentam o movimento: o software deve ser livre para ser modificado, executado, copiado e distribuído. O documento por excelência que marca a luta por essas liberdades é a GPL.

Sem dúvida, Stallman continua sendo o grande filósofo do movimento. No entanto, a partir de 1991, ele se vê obrigado a dividir o palco com uma jovem estrela da Finlândia, Linus Torvalds. Carismático, empreendedor e sabendo usar melhor a internet, ele conseguiu dar solução a um problema que a FSF se dedicava há anos, construir um kernel que suportasse um sistema operacional alternativo. O kernel é uma parte central do sistema, responsável pela configuração e gerenciamento dos dispositivos (teclado, mouse, monitor etc). A FSF já tinha todo o resto da estrutura do sistema pronta e trabalhava no desenvolvimento de seu kernel. Linus foi mais rápido e, mantendo a filosofia livre, adotou soluções tecnicamente mais eficientes, criando o Linux, essa parte essencial do sistema.

O método de desenvolvimento adotado por Linus está em *A Catedral e o Bazar*, livro escrito por Eric Raymond, em 1997. A obra é também uma alfinetada em Stallman, acusado de adotar uma

⁴⁴ Raymond, Eric. “Goodbye, “free software”; hello, “open source”” Visualizado em 27/12/2004 em <http://www.catb.org/~esr/open-source.html>

postura centralizadora de desenvolvimento. A crítica de Raymond aparentemente é voltada ao modelo de desenvolvimento proprietário, mas também refere-se ao desenvolvimento GNU, dizendo que esses códigos são como se fossem catedrais, monumentos sólidos, construídos a partir de um grande planejamento central. Já o desenvolvimento adotado por Linus seria como um bazar, com uma dinâmica altamente descentralizada. Diz Raymond: “De fato, eu penso que a engenhosidade do Linus e a maior parte do que desenvolveu não foram a construção do kernel do Linux em si, mas sim a sua invenção do modelo de desenvolvimento do Linux. Quando eu expressei esta opinião na sua presença uma vez, ele sorriu e calmamente repetiu algo que freqüentemente diz: “Sou basicamente uma pessoa muito preguiçosa que gosta de ganhar crédito por coisas que outras pessoas realmente fazem.” Preguiçoso como uma raposa. Ou, como Robert Heinlein teria dito, muito preguiçoso para falhar.”⁴⁵.

Mas há mais na fala de Raymond com relação ao modelo Linux do que o elogio da técnica - embora o sucesso desta seja inegável. Stallman sempre foi uma figura politicamente muito atuante, não apenas no campo da informática. Mais velho, tendo vivido toda a experiência da luta pelos direitos civis nos EUA, Stallman carrega em sua fala críticas não muito ao gosto das empresas. Em seu site pessoal, por exemplo, ao lado de artigos em favor do software livre, encontram-se também ensaios políticos sobre temas como a invasão estadunidense ao Iraque e o muro de Israel na Palestina. Raymond, por sua vez, é um ardoroso defensor da liberalização do uso de armas, tema usualmente mais ligado às bandeiras da direita liberal.

Linus, por sua vez, além de ser politicamente mais moderado e pragmático, conseguiu criar uma identidade maior com a nova geração de programadores abaixo dos 40 anos, da qual Raymond faz parte. Essa geração, segundo Sam Williams, autor do livro *Free as in Freedom*, é mais energética e ambiciosa. Diz ele: “With Stallman representing the older, wiser contingent of ITS/Unix hackers and Torvalds representing the younger, more energetic crop of Linux hackers, the pairing indicated a

45 Raymond, Eric. A Cathedral e o Bazar. Versão traduzida para o português por Erik Kohler.
<http://www.geocities.com/CollegePark/Union/3590/pt-cathedral-bazaar.html>

symbolic show of unity that could only be beneficial, especially to ambitious younger (i.e., below 40) hackers such as Raymond.”⁴⁶

Desde a ascensão do trabalho de Linus, boa parte do tempo de Stallman tem sido gasta em pedidos para que todos refiram-se ao conjunto do software como GNU/Linux e não apenas Linux. Diz apenas querer que seu trabalho, e de toda FSF, seja reconhecido já que, sem ele, não teria sido possível a existência do Linux

Se o discurso politizado e a integridade radical de Stallman nunca foram de fácil digestão para os programadores da nova geração, ambos são ainda mais indigestos para os empresários. Raymond teve um papel decisivo na criação da alternativa mais ao gosto do paladar corporativo. Em *A Catedral e o Bazar*, ele descreveu um processo de produção inovador e descentralizado, em que as alterações no software são rapidamente entregues à comunidade. Esta, testando e avaliando o produto, estabeleceria uma espécie de seleção natural em que as melhorias sobrevivem e as soluções falhas são logo identificadas⁴⁷. A descrição encantou os executivos da Netscape, dona de navegador de internet que havia sido destruído pela ofensiva agressiva - e anti-competitiva, segundo os próprios tribunais dos EUA - da Microsoft e seu Internet Explorer. Em 1998, Raymond foi a peça chave no processo de convencimento dos executivos da Netscape para que liberassem o código sob uma licença livre, para que a comunidade continuasse seu desenvolvimento.

O prestígio adquirido por Raymond, somado ao do carismático Linus, foram essenciais para que o movimento de código aberto pudesse se estabelecer. Frequentemente, Stallman procura, chegando a ser insistente, deixar claro que o *free* de *free software* (do termo original em inglês), não significa grátis mas livre. A confusão entre livre e grátis tornou-se a justificativa perfeita para que surgisse o termo código aberto.

46 Williams, Sam. *Free as in Freedom: Richard Stallman's Crusade for Free Software*. EUA, março, 2002. Disponível em <http://www.oreilly.com/openbook/freedom/>

47 Idem. “Analyzing the success of the Torvalds approach, Raymond issued a quick analysis: using the Internet as his "petri dish" and the harsh scrutiny of the hacker community as a form of natural selection, Torvalds had created an evolutionary model free of central planning”

Não há diferenças substanciais entre o que os termos software livre e código aberto pretendem definir. Ambos estabelecem praticamente os mesmos parâmetros que uma licença de software deve conter para ser considerada livre ou aberta. Ambas estabelecem, na prática, que o software deve respeitar aquelas quatro liberdades básicas que a FSF estabeleceu. Mas os defensores do termo código aberto afirmam que o termo fez com que os empresários percebessem que o software livre também pode ser comercializado. Teriam sido mudanças “pragmáticas” e não “ideológicas”.

O irônico é que o co-fundador da Open Source Initiative junto com Eric Raymond, veio de um dos setores tidos como dos mais ideológicos do movimento. Bruce Perens é um dos líderes da distribuição Debian, classificada pelo próprio Stallman como uma das que mais se aproxima dos ideais da Free Software Foundation.

Cabe aqui uma pequena explicação sobre o que significa uma distribuição. Como o código do GNU/Linux é livre, ou seja, pode ser modificado e adaptado por qualquer um, os pacotes de software que obedecem a certos padrões e tem certas linhas de desenvolvimento são agrupados nas chamadas distribuições. Em geral, são as empresas que comercializam esses softwares que os agrupam, fazendo com que funcionem a partir de certas regras técnicas e vendendo-os, em caixinhas, para seus clientes. No entanto, existem também as chamadas distribuições da comunidade. Grupos de usuários e programadores empacotam vários programas de modo a que formem um sistema completo, integrando o sistema operacional com diversas ferramentas de desenvolvimento, de escritório, jogos e outros. A distribuição Debian, cuja Definição Debian de Software Livre teve sua redação final feita por Perens, é construída exclusivamente com softwares considerados livres. Ela tem, inclusive, um contrato social⁴⁸.

A definição de Código Aberto⁴⁹ usada pela Open Source Initiative é a Definição Debian de Software Livre, com a mesma formulação, apenas com a omissão das referências ao Debian. No

48 As regras do Contrato Social Debian são: “1. Debian será 100% livre; 2. Vamos retribuir à comunidade software livre; 3. Não esconderemos problemas; 4. Nossa prioridade são os usuários e o software livre; 5. Programas que não atendem nossos padrões de software livre [serão disponibilizados em outras áreas assim identificadas]”. Em http://www.br.debian.org/social_contract

49 Anexo c. Retirado de <http://www.opensource.org/docs/definition.php> em 27/12/2004.

entanto, a definição de código aberto conta também, em cada item, com uma explicação, uma justificativa de sua existência, o que não existe na definição Debian.

Diz o item 3, com sua justificativa:

“3. Trabalhos Derivados

A licença deve permitir modificações e trabalhos derivados, e devem permitir que estes sejam distribuídos sob a mesma licença que o trabalho original.

Fundamentação: A simples habilidade de ver o código fonte não é suficiente para apoiar a revisão independente e a rápida seleção evolutiva. Para que a rápida evolução se concretize, as pessoas devem ser capazes de realizar experimentos e distribuir modificações.”

Aqui há a menção clara ao “achado” de Raymond, a seleção evolutiva decorrente do modo de desenvolvimento de Linus Torvalds. O fim do item 3, explicitado pela fundamentação, é permitir a continuidade do método de trabalho, baseado na revisão dos pares e no encaminhamento de soluções autônomas, sem a necessidade de autorização do autor anterior. A possibilidade de alteração e distribuição da versão modificada já era algo permitido e incentivado pela GPL, porém, com outros fins, que não a melhoria técnica. Não se trata de abdicar do controle, da autoria, da propriedade em nome “progresso”, em nome da melhoria do software e da correção de erros. O que existe é uma noção de autoria coletiva, direitos coletivos e, portanto, bem coletivo, comunitário. Vejamos um trecho do sub-item c do item 2 da GPL, que fala sobre a liberdade para a modificação:

“Portanto, esta cláusula não tem a intenção de afirmar direitos ou contestar os seus direitos sobre uma obra escrita inteiramente por você; a intenção é, antes, de exercer o direito de controlar a distribuição de obras derivadas ou obras coletivas baseadas no Programa.”

Em fevereiro de 1999, Bruce Perens, alegando divergências éticas e pessoais com Eric Raymond, abandonou a Open Source Initiative e retornou à comunidade Debian, de quem havia se distanciado. O fez por meio de um email enviado à lista de discussão dos desenvolvedores Debian intitulado “It's Time to Talk About Free Software Again”. No trecho da mensagem reproduzido abaixo, ele afirma que *open source* e *free software* significam a mesma coisa mas que a OSI não tem enfatizado

a importância da liberdade, o que considera um erro. Na mesma mensagem, ele refere-se ao sistema operacional como Linux e não GNU/Linux, como a própria comunidade refere-se em seu website – e como consta no Contrato Social Debian.

*“Most hackers know that Free Software and Open Source are just two words for the same thing. Unfortunately, though, Open Source has de-emphasized the importance of the freedoms involved in Free Software. It's time for us to fix that. We must make it clear to the world that those freedoms are still important, and that software such as Linux would not be around without them.”*⁵⁰

Perens certamente foi um dos sujeitos que mais tentou conciliar os ditos propósitos pragmáticos da OSI com o debate sobre a liberdade patrocinado pela FSF. Em 2001, logo após as declarações do executivo da Microsoft, Craig Mundie, que criticou o caráter “viral” da GPL, Perens escreveu uma carta assinada conjuntamente por dez membros da comunidade software livre, incluindo Perens, Torvalds, Raymond e Stallman.

O documento, intitulado “Free Software Leaders Stand Together”, usa ao mesmo tempo, e com muita habilidade, o termo *free software* e *open software*. Todas referências aos sistemas são feitas como GNU/Linux e, junto da menção à crescente adoção do modelo *open source* pelas empresas está o alerta que é repetido com frequência por Stallman: “Remember that *Free* refers to liberty, not price”.

Na carta de Perens, há trechos com argumentação muito semelhante à desenvolvidas por Stallman em “The GNU GPL and the American Way”⁵¹. Segue um trecho da carta:

50 <http://lists.debian.org/debian-devel/1999/02/msg01641.html>

51 Microsoft surely would like to have the benefit of our code without the responsibilities. But it has another, more specific purpose in attacking the GNU GPL. Microsoft is known generally for imitation rather than innovation. When Microsoft does something new, its purpose is strategic--not to improve computing for its users, but to close off alternatives for them. Microsoft uses an anticompetitive strategy called "embrace and extend". This means they start with the technology others are using, add a minor wrinkle which is secret so that nobody else can imitate it, then use that secret wrinkle so that only Microsoft software can communicate with other Microsoft software. In some cases, this makes it hard for you to use a non-Microsoft program when others you work with use a Microsoft program. In other cases, this makes it hard for you to use a non-Microsoft program for job A if you use a Microsoft program for job B. Either way, "embrace and extend" magnifies the effect of Microsoft's market power.

No license can stop Microsoft from practicing "embrace and extend" if they are determined to do so at all costs. If they write their own program from scratch, and use none of our code, the license on our code does not affect them. But a total rewrite is costly and hard, and even Microsoft can't do it all the time. Hence their campaign to persuade us to abandon the license that protects our community, the license that won't let them say, "What's yours is mine, and what's mine is mine." They want us to let them take whatever they want, without ever giving anything back. They want us to abandon our defenses. Em <http://gnuweb.kookel.org/ftp/www.gnu.org/philosophy/gpl-american-way.html>

“It's the share and share alike feature of the GPL that intimidates Microsoft, because it defeats their Embrace and Extend strategy. Microsoft tries to retain control of the market by taking the result of open projects and standards, and adding incompatible Microsoft-only features in closed-source. Adding an incompatible feature to a server, for example, then requires a similarly-incompatible client, which forces users to "upgrade". Microsoft uses this deliberate-incompatibility strategy to force its way through the marketplace. But if Microsoft were to attempt to "embrace and extend" GPL software, they would be required to make each incompatible "enhancement" public and available to its competitors. Thus, the GPL threatens the strategy that Microsoft uses to maintain its monopoly. ”

Em ambas as formulações a Microsoft é descrita como uma empresa que deseja “controlar”, seja o mercado, sejam os usuários. Embora na carta a GPL seja retratada de maneira mais ativa (“*GPL defeats*”, “*GPL threatens*”) do que no artigo de Stallman (“*GPL our defense*”), em ambos ela é tida como instrumento de defesa contra a “usurpação” do código promovida pela empresa.

Chama a atenção também, na carta, a ordem das assinaturas, indício das relações de poder e prestígio. Em primeiro lugar, Perens, quem tomou a iniciativa e articulou o grupo. Em seguida, Stallman, seguido por Raymond e, só depois, Torvalds. Os quatro e mais seis “líderes”, entre chefes de projetos importantes e empresários do novo modelo. Todos contra o inimigo comum, a maior defensora do modelo proprietário e dos direitos autorais enrijecidos.

Richard Stallman diz não ver o grupo do código aberto como o inimigo, adjetivo que ele guarda para o modelo proprietário. "We disagree on the basic principles, but agree more or less on the practical recommendations. So we can and do work together on many specific projects. We don't think of the Open Source movement as an enemy. The enemy is proprietary software."⁵², diz.

O fato é que a OSI, entidade cuja criação foi proposta por Eric Raymond, significou uma polarização de poder com a FSF de Stallman. Como ambas as entidades e o movimento como um todo só cresceram nos últimos anos, a longo prazo, isso não significou que Stallman tenha desaparecido, mas sua personalidade e seus modos de ação são tratados crescentemente de maneira caricata e jocosa.

Perens, por outro lado, na carta que marcou seu retorno à comunidade Debian, afirma que, pelo

52 Stallman, Richard. Free Software. Free Society. p.55

menos no período logo após a OSI, as bandeiras da FSF ficaram enfraquecidas. Ele também reafirma seu papel conciliador.

“One of the unfortunate things about Open Source is that it overshadowed the Free Software Foundation's efforts. This was never fair - although some disapprove of Richard Stallman's rhetoric and disagree with his belief that all software should be free, the Open Source Definition is entirely compatible with the Free Software Foundation's goals, and a schism between the two groups should never have been allowed to develop. I objected to that schism, but was not able to get the two parties together.”

Em seu livro de ensaios, *Free Software, Free Society*, Stallman argumenta que o termo código aberto na verdade confundiu mais do que esclareceu. "The official definition of 'open source software,' as published by the Open Source Initiative, is very close to our definition of free software; however, it is a little looser in some respects, and they have accepted a few licenses that we consider unacceptably restrictive of the users. However, the obvious meaning for the expression 'open source software' is 'You can look at the source code.'"⁵³, escreve. De fato, não basta que um usuário possa ler o código de um programa para que ele seja livre. A liberdade para olhar o código é apenas uma das quatro liberdades fundamentais.

Stallman continua, colocando o dedo na ferida apontando uma despolitização do termo. "The main argument for the term "open source software" is that "free software" makes some people uneasy. That's true: talking about freedom, about ethical issues, about responsibilities as well as convenience, is asking people to think about things they might rather ignore. This can trigger discomfort, and some people may reject the idea for that. It does not follow that society would be better off if we stop talking about these things."⁵⁴

De fato, Stallman parece ter razão quando fala do desconforto que suas reivindicações trazem. Em agosto de 1998, em um evento na Califórnia chamado Open Source Development Day, ele foi convidado a palestrar e recebeu instruções explícitas de que não deveria tocar em pontos que pudessem

53 idem

54 idem

afugentar os executivos das empresas, para quem o evento era dirigido. Relata Stallman, em um debate com Eric Raymond publicado na revista estadunidense Salon.com: “I was asked to keep silent about my views that the others disagree with, but they had no intention of holding back their views on the same issues.”.

Mas as incompatibilidades parecem ser de ambas as partes. A retórica empresarial também é algo que dói nos ouvidos do presidente da Free Software Foundation. Continua Stallman, descrevendo o evento:

“Several long speeches during the day were [pervaded] by the assumption that non-free software that relates somehow to free software constitutes "value added" -- an assumption which is the direct opposite of what I am trying to tell people. I was not supposed to state my side of this issue; I was supposed to talk about another topic. I brought up this issue anyway, during my speech, because I was incensed at how the agenda had been set up to present only the other side..

Raymond, por sua vez, não esconde, que com o termo *open source*/código aberto, procurou calar as evidências ideológicas do movimento liderado por Stallman. Em outra entrevista para a revista Salon.com, poucos meses após a fundação da OSI, diz ele:

“Sure. [After meeting with Netscape] I got together with a bunch of free software hackers and we had our own strategy conference. The issue on the table was how to exploit the Netscape breakthrough. We worked out some strategies and tactics. First conclusion: The name "free software" has to go. The problem is nobody knows what "free" means, and to the extent that they do think they know, it's tied in with a whole bunch of ideology and that crazy guy from Boston, Richard Stallman.”⁵⁵

À declaração de conteúdo forte de Raymond, que acabara de chamar de louco um dos dois maiores representantes do movimento, o repórter mostra-se surpreso, ao que Raymond complementa:

“I love Richard dearly, and we've been friends since the '70s and he's done valuable service to our community, but in the battle we are fighting now, ideology is just a handicap. We need to be making arguments based on economics and development processes and expected return. We do not need to behave like Communards pumping our fists on the barricades. This is a losing strategy. So in order to execute that, we needed a new label, and we brainstormed a bunch of them and the one that we finally came up with is "open source.”

A caracterização da atitude de Stallman como comunista não é uma novidade e é algo repetido

55 http://archive.salon.com/21st/feature/1998/04/cov_14feature2.html

até em tom de brincadeira⁵⁶. Communard é usado por Raymond em alusão ao governo socialista que comandou Paris por menos de três meses, em 1871. Os trabalhadores que tomaram o poder, na ocasião, também ficaram conhecidos por terem deixado intactos bilhões de francos do Banco Nacional da França, dinheiro que depois foi utilizado para financiar o exército que derrotou a Comuna. Longe de dar um exemplo fortuito, Raymond está lembrando a todos de um momento em que a hesitação em adotar uma postura “pragmática” acabou condenando todo o movimento.

Stallman, por sua vez, não nega sua inclinação ideológica à esquerda, mas procura isentar o movimento software livre de qualquer filiação a correntes político-partidárias. Diz ele em resposta à pergunta “O software livre está mudando o relacionamento entre a direita e a esquerda?”, feita por uma dupla de jornalistas:

“Pertencendo à esquerda, eu gostaria de dizer que a idéia é da esquerda, mas nos EUA a maioria daqueles que está interessado em software livre estão na direita, e são liberais. Eu não concordo com eles, acho que nós devemos cuidar dos pobres, dos doentes, e não deixar as pessoas morrerem de fome.”⁵⁷

Recusando-se a reconhecer que o movimento identifica-se com o espectro ideológico da esquerda, Stallman assume o sucesso do movimento *open source* nos EUA para mostrar, significando, o que entende por direita e esquerda: “Eu não concordo com eles [direita e liberais], acho que devemos cuidar dos pobres [eles, da direita, não acham], dos doentes [eles não acham] e não deixar as pessoas morrerem de fome [eles não acham]”. O recado é direto para Raymond, militante do Libertarian Party⁵⁸, dos EUA, e que com frequência manifesta-se contrariamente a qualquer regulação governamental sobre a economia e em assuntos sociais.

Raymond, por outro lado, recusa a classificação de “direitista”, dizendo achar “ambos os

56 No dia 1o de abril de 2004, o site NewsForge, bastante visitado pela comunidade da Tecnologia da Informação, publicou, como piada, uma falsa notícia que afirmava que frases de incitação ao comunismo teriam sido encontradas em um software desenvolvido por Richard Stallman. <http://trends.newsforge.com/article.pl?sid=04/03/31/1755246>

57 http://www.geocities.com/CollegePark/Union/3590/direita_esquerda.html

58 O Libertarian Party descreve assim seus compromissos: “The Libertarian Party is committed to America's heritage of freedom: individual liberty and personal responsibility, a free-market economy of abundance and prosperity; a foreign policy of non-intervention, peace, and free trade.” <http://www.lp.org/>

campos do espectro igualmente repugnantes”⁵⁹. Vale destacar aqui dois motivos em especial, entre os dez que ele elenca, de sua aversão à esquerda:

“- Communism. I haven't forgiven the Left for sucking up to the monstrous evil that was the Soviet Union. And I never will.

- Socialism. Liberals have never met a tax, a government intervention, or a forcible redistribution of wealth they didn't like. Their economic program is Communism without the guts to admit it.

- Defining deviancy down. Liberals are in such a desperate rush to embrace the 'victimized by society' and speak the language of compassion that they've forgotten how to condemn harmful, self-destructive and other-destructive behavior.”

O primeiro, em especial, e o segundo ponto, em particular, são relevantes por serem a manifestação de Raymond sobre uma característica associada com frequência ao movimento software livre e mais especificamente à figura de Stallman. Mas o último ponto, combinado com o segundo, é importante por dar conta de sua visão sobre uma questão a partir da qual a GPL, como já vimos, coloca-se no jogo político: a divisão entre os que tem e os que não tem (riqueza, meios de produção ou o código-fonte). Como vimos no segundo capítulo, a GPL, no jogo metafórico da linguagem e de seu deslizamento, não significa apenas o acesso e a possibilidade de alteração do código de um programa. Ela, aplicada, é a restauração da igualdade entre produtores e usuários. O próprio nome livre coloca-se em oposição a proprietário/propriedade.

Para que a mensagem que Raymond quer passar para a comunidade de empresários possa funcionar, levar à frente um discurso sobre a desigualdade e sobre os que têm e os que não têm não parece ser adequado. Não se trata apenas de uma lógica utilitária conscientemente empregada por ele – embora exista a clara noção de que o que Stallman fala incomoda. Raymond quer “vender” as idéias do *open source*. Diz ele, continuando o debate que teve com Stallman publicado na Salon.com:

⁵⁹ Esses comentários de Raymond foram feitos em seu blog, “Armed and Dangerous” (<http://www.ibiblio.org/esrblog/>). A formulação completa é: “I'm not a conservative or right-winger myself, but a radical libertarian who finds both ends of the conventional spectrum about equally repugnant. My tradition is the free-market classical liberalism of Locke and Hayek. I utterly reject both the Marxist program and the reactionary cultural conservatism of Edmund Burke, Russell Kirk, and (today) the Religious Right. Conservatism is defined by a desire to preserve society's existing power relationships; given a choice, I prefer subverting them to preserving them.”

“When the purpose of the event is to sell our ideas to the trade press and business, there are times when the speeches of people you disagree with are functionally helpful and yours are not. Therefore, if I am trying to get victory for all of us, I may have to put pressure on you but not on the people who disagree with you -- even if my private views are actually closer to yours.

Mas nem as idéias que Raymond patrocinou com sua Open Source Initiative estão totalmente de acordo com sua visão. Ele defende os princípios *open source* pela eficiência que vê na prática, pela qualidade do software gerado pela “seleção natural” que descreveu em *A Cadeira e o Bazar*. O efeito “prático” tem mais relevância do que os princípios colocados. Continua ele no mesmo debate:

“I’m not being a hypocrite when I say this, because I myself have positions that I keep quiet about for political and marketing reasons. If the Open Source Definition completely reflected my personal convictions it would be a bit different than it is. But I’ve left it alone because it works. The fact that it works, and the consensus around it, is more important than the points on which I differ with it.

(...)

Either open source is a net win for both producers and consumers on pure self-interest grounds or it is not. If it is, you cannot lose; if it is not, you cannot (and should not) win. Either way, the moralizing you do about how things "ought" to be is at best useless, and at worst actively harmful.”

Há ainda outro elemento que distingue a fala de ambos e as relações derivadas do modelo que propõem. Raymond refere-se com frequência à satisfação dos usuários, como vemos acima, mantendo a separação produtores/consumidores. A marca do discurso de Raymond não é a eliminação da desigualdade, é a eficiência técnica de um método que estabelece uma “seleção natural”, pela qual o software “evolui”. Vejamos o parágrafo que explica o que é *open source* na primeira página do website da Open Source Initiative:

*“The **basic idea behind open source** is very simple: When programmers can read, redistribute, and modify the source code for a piece of software, the software evolves. People improve it, people adapt it, people fix bugs. And this can happen at a speed that, if one is used to the slow pace of conventional software development, seems astonishing.*

We in the open source community have learned that this rapid evolutionary process produces better software than the traditional closed model, in which only a very few programmers can see the source and everybody else must blindly use an opaque block of bits.

Open Source Initiative exists to make this case to the commercial world.

*Open source software is an idea whose time has finally come. For twenty years it has been building momentum in the technical cultures that built the Internet and the World Wide Web. Now it's breaking out into the commercial world, and that's changing all the rules. Are you ready?"*⁶⁰

Algumas expressões merecem ser destacadas pois são as marcas desse discurso derivado da idéia de “seleção natural”. Está dito: “o software evolui/the software evolves” - como se fosse uma espécie animal para a qual os programadores são os responsáveis pela seleção. Também: “Quando a comunidade do código aberto aprendeu que esse veloz processo evolucionário”. E mais: “Há 20 anos esse momentum está sendo construído nas culturas técnicas [em oposição a ideológicas ou idealistas] que construíram a Internet...”, ou seja, um fenômeno alheio aos ideólogos, construído em um ambiente pragmático., o contrário do que Raymond diz ser a FSF.

Elemento inerente ao processo evolutivo, a competição, por outro lado, é algo que, se acirrada, não é vista com bons olhos por Stallman – e principalmente como elemento do capitalismo. No Manifesto GNU, que escreveu em 1985, antes da redação da GPL, e como texto-convite, diz ele:

“O paradigma da competição é uma corrida: recompensando o vencedor, nós encorajamos todos a correr mais rápido. Quando o capitalismo realmente funciona deste modo, ele faz um bom trabalho; mas os defensores estão errados em assumir que as coisas sempre funcionam desta forma. Se os corredores se esquecem do porque a recompensa ser oferecida e buscarem vencer, não importa como, eles podem encontrar outras estratégias -- como, por exemplo, atacar os outros corredores. Se os corredores se envolverem em uma luta corpo-a-corpo, todos eles chegarão mais tarde.

Software proprietário e secreto é o equivalente moral aos corredores em uma luta corpo-a-corpo. É triste dizer, mas o único juiz que nós conseguimos não parece se opor às lutas; ele somente as regula (“para cada 10 metros, você pode disparar um tiro”). Ele na verdade deveria encerrar com as lutas, e penalizar os corredores que tentarem lutar.”

Software livre, cerveja grátis e liberdade de expressão.

60 <http://www.opensource.org/>

Está no Manifesto GNU aquilo que o próprio Stallman chama de “descuidos” ou “mal entendidos” em torno do termo “free”. Ao anunciar o projeto de construção do sistema operacional GNU, Stallman explicitou seu plano de construção de um sistema operacional “free”, dizendo, na primeira frase do texto:

“GNU, which stands for Gnu's Not Unix, is the name for the complete Unix-compatible software system which I am writing so that I can give it away free to everyone who can use it. Several other volunteers are helping me. Contributions of time, money, programs and equipment are greatly needed.”⁶¹

Em uma nota escrita em 1993, Stallman busca fechar o sentido para o termo acrescentando a seguinte explicação:

*“The wording here was careless. The intention was that nobody would have to pay for *permission* to use the GNU system. But the words don't make this clear, and people often interpret them as saying that copies of GNU should always be distributed at little or no charge. That was never the intent; later on, the manifesto mentions the possibility of companies providing the service of distribution for a profit. Subsequently I have learned to distinguish carefully between "free" in the sense of freedom and "free" in the sense of price. Free software is software that users have the freedom to distribute and change. Some users may obtain copies at no charge, while others pay to obtain copies--and if the funds help support improving the software, so much the better. The important thing is that everyone who has a copy has the freedom to cooperate with others in using it.*

Com ou sem a cobrança pela “permissão” de uso do software, uma outra frase do mesmo texto mostra como Stallman, sem precisar se a cobrança seria admitida ou restrita, pensava o sistema operacional como um bem comum, coletivo. Neste trecho, ele considera também que houve uma falta de precisão de sua parte:

“Once GNU is written, everyone will be able to obtain good system software free, just like air.”

Para tentar consolidar o sentido do termo que criou, free software, Stallman repete exaustivamente que o termo não deve ser entendido como software grátis. Para isso, ele usa de uma certa analogia, dizendo que a palavra “free” tem o mesmo sentido do que “free speech” (liberdade de

61 Em todo o texto, tenho optado por utilizar as versões, quando disponíveis, em português (em geral traduzidas por membros da comunidade software livre), pois: 1. minha posição, como analista, é de um falante nativo do português brasileiro; 2. interessam-me, em particular, os efeitos de sentido para a comunidade brasileira. Neste trecho em particular, entretanto, opto por usar a versão em inglês para qua a ambiguidade do termo seja mantida.

expressão) ou “*freedom*”. O uso do termo não é gratuito pois, ao falar em liberdade de expressão ou na palavra “*freedom*”, Stallman usa de dois termos muito importantes para a definição dos valores dos EUA. Deste modo, ele não apenas precisa o sentido da sua expressão mas: reafirma certos valores dos EUA, os quais o próprio Stallman também estima; ele identifica-se com os valores de seu país, afastando-se do estereótipo de “comunista”; e dá, à luta pela liberdade do software, o mesmo status que a luta pela liberdade de expressão. Escrever códigos é um dos meios de expressão de um programador.

Mas a mesma analogia (“*free as in freedom*”), também define “*free software*” por oposição. Diz a FSF no documento em que explica o que é software livre e aponta as quatro liberdades essenciais:

“`Free software` is a matter of liberty, not price. To understand the concept, you should think of `free` as in `free speech,` not as in `free beer.`”⁶²

A idéia de “cerveja grátis” deriva de uma frase que ficou famosa como sendo do economista liberal Milton Friedman (é, na verdade, o título de um de seus livros) e que encerra a idéia de que nada é de graça, para tudo há um preço: “There's no such thing as a free lunch”. A frase é uma alusão à prática de certos restaurantes de oferecer comida salgada e de baixo custo de produção gratuitamente para que o cliente pagasse pelas bebidas. Ao dizer “não pense em almoço grátis/cerveja grátis”, a FSF está dizendo que o modelo é economicamente viável, que não há nenhum truque ou esperteza escondido por trás dele. Mas essa frase não é de Friedman e este, ao comentar essa atribuição errônea, nos mostra mais mais um efeito de sentido por ela engendrado:

I have sometimes been associated with the aphorism "There's no such thing as a free lunch," which I did not invent. I wish more attention were paid to one that I did invent, and that I think is particularly appropriate in this city, "Nobody spends somebody else's money as carefully as he spends his own." But all aphorisms are half-truths. One of our favorite family pursuits on long drives is to try to find the opposite of aphorisms. For example, "History never repeats itself," but "There's nothing new under the sun." Or "look before you leap," but "He who hesitates is lost." The opposite of "There's no such thing as a free lunch" is clearly "The best things in life are free."

Friedman não é alguém que está envolvido diretamente no debate sobre o modelo do software livre, mas o parágrafo acima é para nós interessante pela oposição que nos propõe. Diz ele que o

62 <http://www.gnu.org/philosophy/free-sw.html>

aforismo sobre o almoço grátis encontra seu oposto em “as melhores coisas da vida são de graça”, frase que só pode ser dita por alguém “ingênuo”, que não dá valor às coisas materiais. Ora, é dessa forma que os defensores do software livre têm sido retratados, como sonhadores pouco realistas, que propõe algo inviável. O lado oposto, os defensores do software proprietário seriam, então, os que “sabem que não há almoço grátis”, os realistas, os céticos que vivem o dia-a-dia dos negócios.

Não apenas Stallman e a FSF repetem essa analogia entre *free software* e *free speech*. É algo que a comunidade incorporou como uma explicação padrão para dizer que livre não significa grátis⁶³. Mas mesmo repetida repetida por consecutivas vezes, por Stallman e pelos defensores do modelo livre, o significado de *free* como livre encontra resistências, principalmente na imprensa. Vejamos o seguinte trecho de um livro de divulgação científica editado pela revista *Superinteressante*, da editora Abril:

“Qualquer um que já tenha tropeçado em algum texto ou em alguma idéia de Stallman conhece seu mantra: livre (em inglês, free) se refere a liberdade, não a preço. Tive a sorte de encontrá-lo pela primeira vez em julho de 1996 para uma entrevista. Enquanto mascava e tirava doces M&M do pacote com a mão direita e tentava desembaraçar os longos cabelos com a esquerda, Stallman não parava de repetir seu credo. A palavra free na expressão Software Livre (Free Software, em inglês) não significa grátis. Não significa que é proibido cobrar pelo software. Significa apenas que você deve ser livre para executá-lo, copiá-lo, modificá-lo e redistribuir uma versão melhorada. Stallman afirma que a palavra free está sujeita a muita confusão para aqueles cuja língua materna é o inglês, pois tem duplo sentido: refere-se tanto à liberdade quanto à gratuidade. Para mim, porém, um falante nativo de português, tal ambigüidade não traria problema algum, pois em português há duas palavras, uma para cada caso. Por isso traduzimos a palavra free em free speech e em free beer por duas palavras distintas: livre (o discurso) e grátis (a cerveja), respectivamente. Tudo o que era preciso fazer para entender o conceito seria empregar a tradução correta em português para a expressão Free Software, ou seja, Software Livre.

Stallman então começava a pregar que a liberdade, a ética e o desejo de viver uma vida decente foram suas motivações para fundar a Free Software Foundation (Fundação pelo Software Livre) e publicar a GPL. Ele jamais admitiu ter algo contra o mundo dos negócios ou contra vender software para obter lucro. Negava de todas as formas possíveis ser anticomercial. Devo confessar que deixei a entrevista completamente hipnotizado e absolutamente seduzido pela idéia de que o Software Livre era a única forma decente como um ser humano decente poderia se relacionar com o software (e eu repetia sem parar para mim mesmo: como fui burro e ingênuo ao confundir a liberdade, como em free speech, com o preço, como em free beer). De

63 ““Free” is used as in the phrase “free speech” (a right we covet), rather than the phrase “free beer” (always too good to be true) or “free kitten” (which sounds good, but has a high overhead).”, foi outra formulação encontrada. http://news.com.com/2010-1071-954384.html?tag=fd_nc_1

repente, acordei. O ponto de Stallman com a distinção entre free speech e free beer é tão sedutor quanto inócuo. Por uma razão trivial: quando você é livre (como em free speech) para copiar um programa sem pagar nada - e o Software Livre, de acordo com sua definição, deve permitir essa liberdade-, então, o programa se torna gratuito (como em free beer). Se você sempre tem de pagar para fazer uma cópia, estamos de volta ao mundo do software proprietário. Se não, o programa não custa nada. É grátis. E ponto final.”⁶⁴

Cabe aqui descrever e caracterizar o modo como o autor descreve a figura de Stallman e seus argumentos. Essa descrição implicará na forma como as palavras de Stallman são recebidas, construirá um referencial para o leitor sobre quem profere as palavras entre aspas. Alguns pontos:

- Stallman repete mantras
- Stallman tem longos cabelos
- Stallman repete seu credo
- Stallman prega
- Stallman hipnotiza, é hipnótico

O autor, ao descrever Stallman dessa forma, retrata a figura do presidente da FSF como alguém que não trabalha com uma argumentação racional (a mesma do “mundo dos negócios”). Ele se transforma, então, em um personagem religioso, que repete mantras, que prega e hipnotiza. Durante a “hipnose” do repórter, este relata que Stallman: “... jamais admitiu ter algo contra o mundo dos negócios ou contra vender software para obter lucro. Negava de todas as formas possíveis ser anti-comercial”. A conclusão será: software livre é igual a software grátis, algo que não pode ser comercializado.

A oposição software comercial/software livre, em lugar da oposição software proprietário/software livre também pode ser encontrada no texto de Craig Mundie, executivo da Microsoft, datado de 2001, “The Commercial Software Model”, o mesmo que gerou a carta-resposta dos líderes do software livre, organizada por Bruce Perens. Mundie escreveu ainda um outro texto,

64 Gurovitz, Hélio. *Linux*. Coleção Para saber mais. Ed. Abril, 2003.

menos de 20 dias depois, intitulado “Commercial software, sustainable innovation”⁶⁵, em que repetiu o mesmo termo. Em ambos, a oposição é entre código aberto/software comercial.

Ao que parece, ao traçar esses paralelos, esses sujeitos conseguem o efeito de carregar os softwares de código aberto ou livre como algo fora do mercado, fora de um mundo em que o que prevalece são os argumentos práticos e racionais.

Em página oficial na internet, mantida pelo Instituto Nacional de Tecnologia da Informação (ITI), o governo brasileiro, bastante envolvido com o movimento software livre, publica um pequeno glossário que marca a seguinte definição para software comercial:

*“Software comercial é o software desenvolvido por uma empresa com o objetivo de lucrar com sua utilização. Note que 'comercial' e 'proprietário' não são o mesmo. A maioria do software comercial é proprietário mas existe software livre que é comercial, e existe software não-livre não-comercial.”*⁶⁶

Ao criar o termo *open source*, um dos motivos expostos por Eric Raymond foi resolver o problema da ambiguidade do termo livre em inglês. Diz ele, no texto “Goodbye, “free software”; hello, “open source””:

Specifically, we have a problem with the term “free software”, itself, not the concept. I’ve become convinced that the term has to go.

The problem with it is twofold. First, it’s confusing; the term “free” is very ambiguous (something the Free Software Foundation’s propaganda has to wrestle with constantly). Does “free” mean “no money charged?” or does it mean “free to be modified by anyone”, or something else?

Second, the term makes a lot of corporate types nervous. While this does not intrinsically bother me in the least, we now have a pragmatic interest in converting these people rather than thumbing our noses at them. There’s now a chance we can make serious gains in the mainstream business world without compromising our ideals and commitment to technical excellence -- so it’s time to reposition. We need a new and better label.

Mundie, ao estabelecer sua crítica ao modelo livre e à GPL, em nenhum momento usa a expressão “software livre” (“free software”). Todas as vezes que a palavra livre é usada é no sentido de

65 Mundie, Craig “Commercial software, sustainable innovation” Em <http://news.com.com/2010-1071-281466.html?legacy=cnet> . Visualizado em 27/12/2004

66 <http://www.softwarelivre.gov.br/SwLivre/>

“dar de graça”. O *open source*, ao contrário, é citado várias vezes, assim como a Open Source Initiative. Ao que parece, a solução do problema da ambiguidade do termo dada por Raymond, o nome *open source*, apenas proporcionou à Microsoft uma base de comperação com seu programa *Shared Source* (algo como código compartilhado), em que a empresa “abre” partes do código de certos programas de sua propriedade para algumas entidades selecionadas, como governos e empresas. Entre a Microsoft e essas entidades, é estabelecido um rígido contrato de confidencialidade, em que em nenhum momento a empresa abre mão de sua propriedade intelectual. A existência da OSI e do termo *open source*, além disso, permitiu à Microsoft dialogar com um adversário tão comprometido com valores práticos quanto ela. Ao mesmo tempo, possibilitou ao termo “*free software*” (e, por conseqüência, seus representantes e suas lutas) pudesse ser desconsiderado no eixo horizontal de formulações possíveis do dizer, já que havia outro à disposição.⁶⁷

Conclusão

A análise feita até aqui usou com muita freqüência de declarações e textos de dois personagens: Erci Raymond e Richard Stallman. A opção pela leitura e análise do material produzido por ambos acontece por serem os dois os principais “porta-vozes” desse movimento polarizado. Stallman e Raymond são considerados os “filósofos” dessas duas principais correntes e por eles passam as declarações públicas de duas instituições importantes, a FSF e a OSI. Não se pretende, em nenhum momento, trabalhar sobre as “criações” desses dois sujeitos e sim perceber como as instituições que chefiam usam de argumentos baseados em memórias discursivas com diferentes interpretações.

Um pouco do que conseguimos perceber aqui, traçando o percurso desses discursos, foi como a

67 Em Orlandi, 2003, p.32: “Courtine (1984) explicita essa diferença considerando a constituição – o que estamos chamando de interdiscurso – representada por um eixo vertical onde teríamos todos os dizeres já ditos – e esquecidos – em uma estratificação de enunciados que, em seu conjunto, representa o dizível. E teríamos o eixo horizontal – o intradiscurso – que seria o eixo da formulação, isto é, aquilo que estamos dizendo naquele momento dado, em condições dadas.”

criação de uma entidade (a Open Source Initiative) de um termo em particular (*open source*) foi algo de grande impacto na aceitação do movimento por uma parte da sociedade. Para melhor retratar o posicionamento desses termos, o quadro abaixo será interessante:

software livre	código aberto	software proprietário (comercial)
liberdade	eficiência/evolução	realidade/experiência
GNU/Linux	Linux	Windows
FSF	OSI	Microsoft

No discurso, o sentidos desses termos derivam, transformam-se, relacionam-se e ressignificam-se uns aos outros. Dizer de uma forma ou de outra, usar esta ou aquela palavra, implica em uma cadeia de relações de sentidos e de memórias, implica em filiações. O próprio Stallman faz uma estrita “vigilância” nesse sentido⁶⁸, e considera o uso de certos termos como prova de compromisso com a FSF e o software livre. Não se trata apenas do reconhecimento do trabalho da FSF na construção do GNU/Linux, dizer apenas Linux implica em associação com a OSI. Na memória, dizer Linux se mostra como dizer “estou nisso pela eficiência, pelas vantagens técnicas”.

Pelas declarações de Raymond, o objetivo com a OSI era resolver o problema da ambigüidade do termo e criar uma linguagem que não deixasse desconfortável os executivos e as empresas, vistos como potenciais parceiros. Mas o que vemos é que, enquanto há fala, há política. Pela derivação dos sentidos, o novo termo não se esvaziou politicamente, pelo contrário, aproximou-se do “outro lado”, do mundo da propriedade. O quadro acima mostra como, na história, certos sentidos se ligam a outros e para eles derivam. Foi depois que um novo sentido se estabilizou, o *open source*, que alguns empresários decidiram se aventurar nesse novo modelo de negócios.

De fato, o mundo dos negócios, principalmente nos EUA, refere-se quase que exclusivamente

⁶⁸“ Pergunta, seguida de resposta do próprio, feita a Richard M Stallman: “Enquanto o Projeto KDE e a Free Software Foundation algumas vezes tiveram um duro relacionamento, parece que alguns de seus desenvolvedores têm estendido a mão à FSF – por exemplo, incluindo GNU no termo que alguns adotaram para se referir ao desktop GNU/Linux baseado no KDE: KGX (KDE/GNU/linuX). Você acha que pode existir espaço para mais cooperação entre os dois projetos em um futuro próximo?”

RMS: Certamente. (...) Quanto mais eles quiserem cooperar conosco, mais nós podemos cooperar.”

ao *open source*. E não poderia ser diferente, dado que a expressão *free* software traz consigo a denuncia da desigualdade de que existem os que têm e os que não tem. Não será esse discurso que fará sentido para os empresários.

Mas há mais, as razões do sucesso do termo *open source* podem ser encontradas também na sua proximidade com a idéia de seleção natural e de ambiente competitivo. Para os que têm, para os donos dos meios de produção, a desigualdade é fruto da diferença de performance daqueles que são mais eficientes e dos que são inaptos. Os “melhores” sobrevivem, enriquecem e são recompensados. Os “inaptos” morrem e, com isso, o sistema, em si, melhora. Sem estar de acordo com essa filosofia de desenvolvimento, o termo *open source* estaria fadado ao fracasso e não faria sentido. Não é possível simplesmente desassociar-se de um campo, será necessário assimilar-se ao outro, dominante, dando a impressão de que o que havia de ideológico se calou, quando apenas transmutou-se, deu margem a derivações.

Também, ao existir, o termo código aberto contribuiu com uma marginalização da expressão software livre, fazendo com que ela fosse cada vez mais associada com “idealismo”, “religião”, “credo”, “radicalidade”. Se é sob o termo “código aberto” que se abrigam aqueles do movimento que privilegiam a técnica e a eficiência, os outros são os “ideólogos”, os “comunistas” - e os menos “eficientes”.

Não parece ser correto atribuir a adesão dos empresários ao software livre como algo exclusivamente ligado à criação de um novo termo como “código aberto”. Desde meados da década de 1990 era crescente o número de empresas que passou a trabalhar com software livres. O que o termo e a instituição que o corroborou (a OSI) parecem ter permitido foi a ligação de suas iniciativas empresariais a uma instituição (a OSI) e um termo (o “código aberto”) em que a não distinção entre usuário e produtor aparecesse de forma neutralizada (até mesmo recusável ou passível de ser esquecida).

Mas não seria apenas no exame dos termos *free* e *open* que conseguiríamos identificar esses

fatores. O sentido não se restringe a esses termos, eles têm uma história, são ditos por certas pessoas, estão no mundo. E é na história que eles significam-se, são reformulados e associados a idéias que vão além do significado imediato daquelas palavras.

Capítulo 4. Software Livre na periferia do sistema

Nascido nos EUA e “completado” por um finlandês, o sistema operacional GNU/Linux e sua filosofia de desenvolvimento passam, a partir do fim de década de 1990, a atrair a atenção também nos países periféricos. No Brasil, os primeiros grupos de usuários surgem nesse mesmo período, alguns deles ligados a universidades, outros formados por técnicos e jovens aficionados por informática. A primeira distribuição brasileira de Linux, a Conectiva, produzida numa empresa sediada em Curitiba, foi lançada em 1997.

No entanto, uma das primeiras iniciativas institucionais pela adoção do software livre em larga escala (embora sem ter alcançado êxito) vem de um político. Em dezembro de 1999⁶⁹, o deputado Walter Pinheiro, do Partido dos Trabalhadores, enviou ao Congresso Nacional um projeto de Lei que pedia a adoção preferencial, por parte do poder público, de softwares com licenciamento livre. O projeto até hoje não foi votado na Câmara.

De projetos como esses, que marcam uma intervenção clara do poder público na adoção de certos padrões de relação de propriedade do produto software e, por consequência, de determinadas tecnologias⁷⁰, derivam debates muito interessantes, principalmente se considerarmos que a licença livre institui uma nova relação entre usuário e produtor (ou dono) do software, em que ao primeiro é permitido igualar-se ao segundo, ou seja, em que a diferença entre o simples usuário e o desenvolvedor do software é nublada e pode ser anulada.

⁶⁹ http://www.camara.gov.br/Internet/sileg/Prop_Detalhe.asp?id=17879

⁷⁰ Como já pudemos ver, embora o que esteja sendo adotado preferencialmente seja um determinado tipo de licenciamento, isso implica também na adoção de certos padrões tecnológicos, já que há toda uma linha de softwares que funcionam em conjunto e para isso tem um sistema de licenciamento compatível.

Como vimos, há uma contradição entre a licença livre a licença proprietária, em que os proponentes da primeira são objeto de uma tentativa de silenciamento pelos defensores da segunda. A licença livre aponta a condição desigual entre aqueles que podem e os que não podem alterar, distribuir e copiar o software, postulando uma nova norma. Como se posicionam, então, nesse debate, sujeitos que vivem em sociedades em que diferença social e de classe é bastante acentuada? Como imaginam o papel dessa tecnologia, que traz em si uma nova relação de propriedade, e como a relacionam com a inserção periférica e dependente de seus países?

Sob esse ponto de vista, será interessante, então, analisar três documentos/debates em especial: o já referido Projeto de Lei do deputado brasileiro Walter Pinheiro; o projeto de caráter semelhante do deputado peruano Edgard Villanueva Nuñez, que foi seguido de carta de representante da Microsoft contestando o projeto e de tréplica do deputado; e o debate marcado pelo início do governo Lula no Brasil, que iniciou uma série de ações para a o uso preferencial do software livre. Um dos momentos mais intensos desse debate foi quando de uma ação judicial da Microsoft contra o presidente do Instituto Nacional de Tecnologia da Informação, Sérgio Amadeu, acusado de difamar a empresa.

O projeto de Pinheiro e os apensados

Apresentado em dezembro de 1999, o projeto de Lei do deputado Walter Pinheiro encontra-se em tramitação na Câmara Federal. A ele, foram apensados mais cinco outros projetos, tidos como de teor semelhante. Um foi apresentado em 2000, pelo deputado Werner Wanderer (PFL)⁷¹, outro em 2001 pelo deputado Luiz Bittencourt (PMDB)⁷², outro em 2002 pelo deputado Sérgio Miranda (PC do B)⁷³, outro em 2003 pelo deputado Coronel Alves (PL)⁷⁴ e o último em 2004, pelo deputado Luiz Couto

71 http://www.camara.gov.br/Internet/sileg/Prop_Detalhe.asp?id=19028

72 http://www.camara.gov.br/Internet/sileg/Prop_Detalhe.asp?id=26688

73 http://www.camara.gov.br/Internet/sileg/Prop_Detalhe.asp?id=64532

74 http://www.camara.gov.br/Internet/sileg/Prop_Detalhe.asp?id=136056

(PT)⁷⁵.

Boa parte da justificativa presente no projeto do deputado Walter Pinheiro é apenas uma tradução fiel de trechos do texto “15 Years of Free Software”, escrito por Richard Stallman para a revista *Linux Today* em março de 1999⁷⁶. Trata-se de um pequeno histórico do projeto GNU, em que Stallman afirma sua motivação ao iniciar o projeto⁷⁷, como este foi completado com o kernel Linux e como o conjunto já seria utilizado por milhões de pessoas em todo mundo.

No trecho final da justificativa do deputado é adicionado o seguinte trecho, que não existe no texto de Stallman:

“Este programa [o GNU/Linux] é apenas um exemplo de como a liberdade na alteração, distribuição e utilização de programas de computador pode transformar ainda mais rapidamente, e de maneira mais democrática, o perfil do desenvolvimento social e tecnológico no mundo.

O Estado, como ente fomentador do desenvolvimento tecnológico e da democratização do acesso a novas tecnologias para a sociedade, não pode se furtar a sua responsabilidade de priorizar a utilização de programas abertos ou os “free software/open source”. E se as pequenas, médias e grandes empresas multinacionais já estão adotando programas abertos, evitando assim o pagamento de centenas de milhões de dólares em licenciamento de programas, porque deveria o Estado, com uma infinidade de causas sociais carentes de recursos, continuar comprando, e caro, os programas de mercado.”

Logo na primeira frase, o modelo livre de licenciamento, que gerou o GNU/Linux, é descrito como algo que pode “transformar ainda mais rapidamente”, ou seja, acelerar uma transformação já em curso de “o perfil do desenvolvimento social e tecnológico no mundo”. À rápida transformação no “perfil”, adiciona-se uma característica, seria “mais democrática”. A licença livre, que permite que usuários alterem os programas, poderia acelerar a alteração (“transformar” o “perfil”) da rota de desenvolvimento social e tecnológico.

75 http://www.camara.gov.br/Internet/sileg/Prop_Detalhe.asp?id=159126

76 Stallman, Richard. “15 years of Free Software”, *Linux Today*, 17/03/1999, http://linuxtoday.com/news_story.php3?ltsn=1999-03-17-003-10-NW-LF

77 “Em 1984 era impossível usar um computador moderno sem a instalação de um sistema operacional proprietário, fornecido mediante licenças restritivas de amplo espectro. Ninguém tinha permissão para compartilhar programas (software) livremente com outros usuários de computador, e dificilmente alguém poderia mudar os programas para satisfazer as suas necessidades operacionais específicas”

Em seguida, o Estado é colocado em um papel ativo no “desenvolvimento tecnológico” e na “democratização do acesso”. Ao mesmo tempo, para sua ação é estabelecido um referencial externo, internacional, as “grandes médias e pequenas empresas multinacionais”, que já estariam em processo de adoção de “programas abertos”, mas com o objetivo de reduzirem custos. O Estado, “carente de recursos”, deveria fazer o mesmo.

Ao justificar a Lei, o projeto, então, caminha em duas direções: a primeira, a da ênfase na transformação social e tecnológica, rumo a algo “mais democrático”⁷⁸; a segunda, a da justificativa econômica (o Estado tem que economizar, deve priorizar os investimentos sociais), para a qual buscase um exemplo vindo do exterior e do setor empresarial (aquele que racionaliza ao máximo seus gastos). É muito interessante também como, embora seja dito que o GNU/Linux é usado por empresas, ele é colocado fora do “mercado”. O Estado deveria deixar de comprar os “programas de mercado”.

A justificativa econômica é a única que pode ser encontrada também em todos outros cinco projetos de Lei pensados ao de Pinheiro. Os projetos dos deputados Wanderer (de 2000) e Bittencourt (de 2001) apresentam essa como única justificativa. Já o projeto de Alves (de 2003) soma a essa justificativa uma melhor eficiência do sistema livre, em argumentação semelhante à usada, como vimos no capítulo anterior, pelos defensores do open source/código aberto. Já o projeto de Miranda (de 2002) dirige-se à oferta, por parte do governo federal, de serviços na internet que funcionem também em computadores que tenham sistemas livres instalados. O projeto de Couto (de 2004), por sua vez, está voltado ao uso de sistemas livres para a educação e usa de trechos contidos nos projetos de Pinheiro e Bittencourt.

Dos seis projetos, o de Alves é o único que fala em software livre. Os outros usam do termo “programa/sistema aberto”, fazendo referência à livre manipulação e distribuição do código fonte. Os

78 Em outro texto, preparado em 2001, para um Fórum na Câmara dos Deputados, Pinheiro ainda reforça essa argumentação dizendo, em um primeiro momento:

“Gostaríamos que o leitor atentasse ao movimento internacional pelo software livre porque estamos falando de uma revolução na informática e, mais que isso, no acesso ao conhecimento. Estamos falando de programas sem proprietários, programas livres.”

projetos de Pinheiro e Couto falam também em “free software/open source” e “open/free software”. Pela descrição contida no projeto de Wanderer percebe-se que o termo “programa/sistema aberto” não é usado como sinônimo nem de open source nem de software livre. A referência é apenas a um licenciamento que permite ao usuário a leitura do código fonte do programa, sem necessariamente estar permitida a alteração ou redistribuição. Esse projeto não propõe a adoção de uma solução livre em lugar de uma solução proprietária, ele fala na adoção de sistemas abertos em lugar de sistemas “fechados” (aspas contidas no Projeto), ou seja, pede o uso de sistemas que podem ser proprietários mas os quais se possa ler o código-fonte.

Mas o que chama bastante a atenção, em todos os projetos, é a relação que estes estabelecem com o exterior (outros países), com as empresas, com o mercado e, em especial, com a maior empresa de software proprietário, a Microsoft, e seu sistema operacional, o Windows.

Quero destacar o seguinte trecho, contido no projeto do deputado Luiz Bittencourt (de 2001):

“...[GNU/Linux] coloca-se como alternativa viável e extremamente bem sucedida a sistemas operacionais proprietários, tal como o Windows, cujos distribuidores adotam práticas monopolísticas muito agressivas.

Tais programas abertos vêm mostrando sua eficácia em inúmeras aplicações, em especial no setor privado, com a vantagem de poderem ser adaptados e executarem em diversos equipamentos. Grandes multinacionais, a exemplo da Chrysler, Boeing, Sony e Mercedes Benz, adotam em muitos de seus sistemas soluções abertas, reduzindo custos com licenciamento de programas e garantindo uma uniformidade de soluções, independente de plataformas.”

No projeto de Pinheiro, existe a menção ao “sistema operacional proprietário”, mas este não recebe nenhum nome em especial, nenhum “proprietário” é especificado. A expressão, inclusive, está no trecho que é uma reprodução do artigo de Stallman. No projeto de Bittencourt, a figura do software proprietário por excelência ganha um nome (“Windows”) e a empresa que o detém (a Microsoft), recebe uma descrição para seu modo de fazer negócios (“práticas monopolísticas muito agressivas”). Ao mesmo tempo, “grandes multinacionais, a exemplo da Chrysler, Boeing, Sony e Mercedes Benz” dão um exemplo de como fugir dessa empresa e reduzir custos.

A descrição da prática dessa empresa como agressiva em demasia aparece também no projeto de Werner Wanderer, de 2001. No entanto, ela não é citada nominalmente:

“Cabe lembrar, ainda, que a iniciativa [compra de programas de computador mais baratos com funcionalidade equivalente] foi perseguida durante anos pelo Poder Executivo Federal através de um programa normativo de compras governamentais, que acabou preterido em virtude de pressões de marketing de grandes fornecedores de programas de computador.”

A acusação de “práticas monopolísticas” contra os “detentores desse sistema [Windows]” é repetida no texto do projeto de Miranda, que não cita a empresa nominalmente, apenas o produto. O nome da Microsoft vai aparecer no projeto do deputado Alves, que diz: “Cada vez que a Microsoft lança uma nova versão do Windows lá se vão bilhões de dólares no mundo inteiro para pagamento de licenças para operar o sistema.”.

O texto do projeto do deputado Couto, aquele que soma trecho do projeto de Pinheiro e Bittencourt, mantém, do primeiro, apenas o trecho que descreve resumidamente a história do GNU/Linux e reproduz, do segundo, a referência às grandes empresas que já usam sistemas livres, agregando quais seriam os motivos destas:

“Hoje, este sistema operacional é usado por milhões de pessoas, de forma livre, no mundo inteiro. Mais do que isso, há um incontável número de Empresas, entre elas as gigantes multinacionais Mercedes Benz, General Motors, Sony Electronics optaram pelo uso de softwares livres. São três os principais motivos que levaram tais Empresas a essa opção: 1) a liberdade para criar soluções próprias que muitas vezes ficam comprometidas pela dependência e atrelamento a padrões fechados de softwares. 2) a segurança de seus sistemas de informação na produção, organização, gerenciamento e distribuição de informações. 3) o mais importante motivo - a drástica redução de custos. Com a adoção de softwares livres, estas Empresas exoneram-se da obrigação de pagamento de licenças e ainda contam com a vantagem de ter parte desses programas abertos distribuídos gratuitamente.”

Como o projeto de Pinheiro, o de Couto não cita diretamente o Windows nem a Microsoft. No entanto, ele descreve conseqüências nocivas da “hegemonia dos sistemas fechados” e aproxima-se de Pinheiro ao apontar certas vantagens ao desenvolvimento nacional derivada do uso de sistemas abertos:

“Também, há em curso por todo o País um amplo e necessário debate sobre o uso monopolista dos programas de computadores denominados de softwares proprietários.”

A hegemonia dos sistemas fechados causam prejuízos a economia nacional e inibem o desenvolvimento de ciência e tecnologia nacional.

Na área de informática, o ensino público deve repensar esse modelo que transforma as pessoas em “clicadores” de botões, que mais se assemelham a robôs adestrados para a execução de uma única tarefa. O papel da educação pública, por outro lado, é estimular a capacidade criativa das pessoas, objetivando transformá-las em cidadãos pensantes capazes de produzir e reproduzir conhecimento e de fomentar os empreendedores sociais, empresariais e tecnológicos.”

“Clicadores de botões, que mais se assemelham a robôs adestrados” são os usuários do Windows. Com eles, não se conseguiria o “desenvolvimento”. Seria preciso, então, criar “cidadãos pensantes capazes de produzir”. Assim, sem deixar de argumentar também em favor da redução dos gastos públicos, Couto retoma o elogio da possível indistinção entre produtores e usuários na argumentação em favor do software livre. Um pouco mais à frente no texto ele ainda diz, reforçando a idéia de mudança social proporcionada pelo software livre:

“Com isso, modifica-se a orientação tradicional de nossa educação na área de informática, também constitui-se como passo importante no sentido de fomentar outras possibilidades de exploração de software e equipamento, ampliando horizontes de conhecimentos e ajudando principalmente a reduzir custos.”

Assim como o projeto de Pinheiro fala em “transformar o desenvolvimento”, o projeto de Couto fala em modificar “a orientação tradicional de nossa educação”. A crítica, aqui, também é contra a Microsoft, que é conhecida na comunidade software livre por oferecer de graça o sistema operacional Windows para escolas e projetos de inclusão digital. Ao oferecer seus programas, a empresa treinaria os beneficiados em sua tecnologia, facilitando a penetração do produto no mercado⁷⁹.

O projeto do Peru

Tornou-se notória na comunidade software livre a discussão que opôs o congressista peruano

⁷⁹ A comparação mais frequente é com um “presente de grego”. Queiroz, Rubens. “Presente de grego”. Site Dicas-I, http://www.dicas-l.unicamp.br/artigos/linux/presente_de_grego.shtml

Edgar Villanueva Nuñez e a Microsoft, quando da apresentação, em 2001, do projeto de autoria de Nuñez que propõe a adoção prioritária de sistemas livres pelo governo peruano. Pouco mais de três meses após apresentação do projeto de Lei, em março de 2001, a empresa enviou uma carta ao congressista peruano questionando o projeto. A réplica de Nuñez, que procurou rebater quase parágrafo por parágrafo a carta da empresa, circulou vigorosamente pela internet, sendo usada como exemplo de boa argumentação por toda a comunidade livre.

No entanto, gostaria de me deter especialmente na análise do texto do projeto da Nuñez⁸⁰ e na carta⁸¹ assinada pelo gerente geral da Microsoft Peru, Juan Alberto González, endereçada ao congressista. Considero que ali encontra-se o centro da discussão que estabeleço aqui de uma maneira geral (a reflexão sobre a Política) e em especial neste capítulo (a reflexão sobre como esse debate se desdobra na periferia).

Em primeiro lugar, quero comentar alguns trechos do projeto. A primeira seção do texto é intitulada “Considerando”, e estabelece certos pressupostos legais para a justificativa (“exposición de motivos”) que virá a seguir. A primeira consideração refere-se ao papel do Estado com relação ao desenvolvimento da ciência e da tecnologia, enfatizando o ativo papel estatal, e recebe a seguinte formulação:

“Que, el artículo 14° de la Constitución establece que es deber del Estado promover el desarrollo científico y tecnológico del país.”

Em seguida, a justificação descreve um contexto histórico (a globalização), estabelece uma relação entre software e conhecimento e aponta para uma situação dependente entre usuário/proprietário mas que também se desdobra entre nacional/estrangeiro. Vejamos os primeiros quatro parágrafos:

“En la época de globalización que vivimos uno de los factores claves para el desarrollo de los países es el dominio de la ciencia y de la tecnología.”

80 <http://www.gnu.org.pe/proley1.html>

81 <http://www.gnu.org.pe/mscarta.html>

Esto es especialmente cierto con relación a la informática que permite mediante programas o “software” procesar información necesaria para las distintas áreas del conocimiento. En la actualidad no existe prácticamente ninguna actividad que no este relacionada de una u otra manera con la informática y obviamente con los programas informáticos.

Esto crea una situación de dependencia en la que nos convertimos en consumidores de tecnología elaborada en otros países. Es más, esta tecnología se encuentra en constante desarrollo, y como consecuencia de ello lo que hoy es novedoso en poco tiempo se convierte en obsoleto; obligándonos así a tener que adquirir la nueva tecnología sino queremos quedar rezagados.

Esta carrera por no quedar atrasados resulta en elevados costos por pago del derecho de uso de esta tecnología informática. Esto es particularmente cierto en la administración pública, como en la actividad privada, en las que por concepto de licencias de “software” se debe abonar elevadas sumas de dinero cada año.”

É especialmente interessante como a função, o funcionamento do software é descrito como “processar informação” e como estabelece-se a imagem de que “praticamente nenhuma atividade” deixa de estar relacionada a ele. Em seguida, a relação que seria apenas de consumo torna-se equivalente a uma situação de “dependência”. Essa “dependência” se mostra ainda mais perigosa por se dar em relação ao exterior, a um outro país.

Ainda no mesmo terceiro parágrafo e depois no quarto, surge a imagem do atraso (“quedar rezagados”). Acompanhar o ritmo de desenvolvimento da tecnologia internacional será algo caro mas, ao mesmo tempo, não acompanhar significa ser deixado para trás. O desenvolvimento tecnológico é descrito como uma corrida (“carrera”), em que os participantes seriam, então, os países.

A argumentação econômica está presente, mas o alto custo dessa “corrida” é derivado da situação de “consumidor” ocupada pelo país. Em um projeto intitulado “Projecto de Ley Software Libre”, o software livre já se anuncia como um meio para romper esse “círculo vicioso”⁸² de dependência externa.

Mesmo usando o termo software livre, e não código ou programa aberto, e citando as quatro liberdades que definem o software livre segundo a Free Software Foundation (FSF), o projeto,

⁸² O sexto parágrafo dessa seção do projeto diz:

“Este panorama hace necesario que el Estado se preocupe de dar alternativas de solución que permitan romper el círculo vicioso de dependencia tecnológica en la que nos encontramos, así como a los problemas que de ello se derivan.”

possivelmente para enfatizar a economia de recursos, repete a dicotomia software livre x software comercial (dicotomia que a FSF repudia pois o software livre também pode ser comercializado). Assim, como no projeto brasileiro de Pinheiro, posiciona os sistemas livres fora do mercado:

“Precisamente, la alternativa inteligente y viable a los programas informáticos o “software” comercial viene a ser el denominado “software libre” denominado así porque se basa en el libre empleo de los programas de computación.”

Para exemplificar iniciativas semelhantes à desse projeto de Lei que instituirá a preferência pelo software livre, o exemplo é internacional e inclui empresas e “organizações”. No entanto, os nomeados são países:

“Cabe reiterar que debido a las ventajas que ofrece que los mismos usuarios creen soluciones propias, estableciendo sus propios patrones de seguridad hace que sean cada vez más el número de empresas y organismos que adopten el sistema de software libre, como por ejemplo el Ministerio de Defensa de Francia y la República Popular China.”

As “soluções próprias” possíveis de serem encontradas pelo usuário do software livre levariam a padrões de segurança únicos. Os exemplos referidos, França e China, países conhecidos por sua oposição/desconfiança com relação ao governo dos Estados Unidos, fazem com que a origem da ameaça contra a segurança seja imaginada não como partindo de indivíduos mas sim partindo do governo dos Estados Unidos. Sendo o software um processador de informações como foi anunciado anteriormente, ele estaria vulnerável ao governo dos EUA se o software livre não for utilizado.

Para contra-argumentar o projeto, a carta enviada pelo representante da Microsoft apóia-se na construção de uma imagem da empresa como parceira dos Estados e defensora de leis que são aplicadas internacionalmente. Ao mesmo tempo, tenta caracterizar o deputado como alguém que não maneja corretamente os conceitos que emprega no texto do projeto. Mas de mais impacto será a ameaça velada que a carta faz, de que essa espécie de atalho para romper o “círculo vicioso” que seria a adoção do software livre, poderá ser interpretada como uma ilegalidade, um burlar de regras internacionais, que teria por consequência a redução dos investimentos externos, significando a condenação definitiva do

Peru ao atraso.

Vejamos o trecho inicial, que faz alusão a um encontro prévio entre o deputado e representantes da empresa:

“(...) Precisamente, fruto de nuestra reunión hoy Usted conoce de nuestros avances a nivel internacional en el diseño de nuevos servicios para el ciudadano, dentro del marco de un Estado modelo que respeta y protege los derechos de autor.

Este accionar, tal como conversamos, es parte de una iniciativa mundial y hoy en día existen diversas experiencias que han permitido colaborar con programas de apoyo al Estado y a la comunidad en la adopción de la tecnología como un elemento estratégico para impactar en la calidad de vida de los ciudadanos.”

De início, o deputado é colocado como alguém que está recebendo informações – que podem ser entendidas como as quais ele não teria tido acesso antes ou as quais ele, no futuro, não poderá negar ter recebido – de que a empresa pode “oferecer serviços para o cidadão” se o Estado for “modelo”, ou seja, “respeitar os direitos autorais”. No mundo todo, é reforçado no parágrafo seguinte, ela agiria assim. Frente a ênfase na ação em nível mundial da empresa, o Estado peruano se apequena. E, ao reforçar a legalidade e a defesa de direitos de propriedade intelectual do software proprietário, marca o software livre como algo mais próximo da ilegalidade.

Ao estabelecer o par legalidade x ilegalidade e ao referenciar ações mundiais, o representante da Microsoft está lembrando da existência das instituições que zelam pelas normas do comércio mundial, como a Organização Mundial do Comércio (OMC). A carta classifica o projeto de Lei de Nuñez como “discriminatório” e “anti-competitivo”. Esses são os termos exatos que definem algumas das violações de normas da OMC, o que pode valer ao país sanções comerciais.

Agigantada frente ao Estado peruano, apoiada nos ombros da OMC, a indústria do software proprietário - “aquela que verdadeiramente faz investimentos importantes” - ameaça então deixar o país, o que causaria desemprego e uma redução no Produto Interno Bruto do país:

“Así, al obligar al Estado a favorecer un modelo de negocios que apoyaría exclusivamente el software de código abierto, el proyecto sólo estaría desalentando a las compañías fabricantes locales e internacionales que son las que verdaderamente

realizan importantes inversiones, crean un significativo número de puestos de empleos directos e indirectos, además de contribuir al PBI vs. un modelo de software de código abierto que tiende a tener un impacto económico cada vez menor debido a que crea principalmente empleos en servicio.”

Embora o projeto do deputado fale exclusivamente em software livre, a carta insiste no termo código aberto e considera o termo livre como um erro.

“El proyecto maneja de manera errónea los conceptos de software de código abierto, que no necesariamente implica que sea software libre o de costo cero, llegando a realizar conclusiones equívocas sobre ahorros para el Estado, sin ningún sustento costo beneficio que valide la posición.

Es equivocado pensar que el Software de Código Abierto es gratuito. Investigaciones realizadas por Gartner Group (importante investigadora del mercado tecnológico reconocida a nivel mundial) han señalado que el costo de adquisición del software (sistema operativo y aplicaciones) se reduce a sólo 8% del total de costos que las empresas e instituciones deben asumir como consecuencia del uso racional y realmente provechoso de la tecnología. El otro 92% lo constituyen: costos de implantación, capacitación, soporte, mantenimiento, administración e inoperatividad.”

Mesmo que a confusão entre os termos livre e grátis não exista no espanhol, assim como não existe no português, o deputado é acusado de ter cometido um erro, de considerar o software livre como software gratuito (“de custo zero”). Para desmentir o deputado, surge uma instituição internacional, o Gartner Group, descrita como “importante investigadora” e reconhecida em “nível mundial”. Agora, quem se torna reduzido é o próprio deputado, frente a uma instituição de maior relevância internacional que ele. E mais, esses números, que mostrariam que o custo das licenças é reduzido, seriam obtidos quando o uso da tecnologia é “racional” e “realmente proveitoso”. Antes mesmo de qualquer contra-argumentação do deputado, que pode questionar esses números, a resposta acaba sendo previamente adjetivada como não-racional.

Dois momentos no Brasil

Retornando esta análise para o contexto brasileiro, quero caracterizar dois momentos do debate

que acho relevantes pois exemplificam o uso de categorias muito semelhantes às que vimos até agora, tanto com relação à defesa do uso das licenças livres como com relação às respectivas tentativas de silenciamento. Em especial quero me deter na análise de dois pontos que considero terem mostrado sua relevância até aqui. O primeiro é o jogo entre a diferenciação ou não-diferenciação entre usuários e produtores, quando é ressaltada a característica da licença livre de permitir ao usuário que altere e redistribua o software. O segundo ponto relaciona-se à exposição de certos sujeitos (países, empresas, indivíduos, iniciativas) que usam ou não usam o software livre. Ao se utilizar desses exemplos, o que estão dizendo os sujeitos sobre o significado do emprego das tecnologias? A partir dessa questão, surgem significados para palavras como “atraso”, “autonomia”, “dependência/independência”, “desenvolvimento” e outros.

Os dois momentos que analisarei têm como contexto a eleição e o exercício do poder por parte do Partido dos Trabalhadores (PT) na esfera executiva. O Projeto Software Livre RS (PSL-RS), coletivo dedicado à promoção do software livre baseado no Rio Grande do Sul, assim como outros PSL's de outros estados, têm entre seus membros diversos militantes do PT. Marcelo Branco, do PSL-RS, foi entrevistado no final de 2002⁸³ e demonstrou grande expectativas com o governo Lula, afirmando ser o novo governo um espaço para a expansão do uso das licenças livres. Para responder às afirmações de Branco, o executivo da Microsoft do Brasil, Eduardo Campos⁸⁴, pediu para ser entrevistado pelo mesmo veículo, argumentando contrariamente às licenças livres tendo como perspectiva as conseqüências para o Brasil.

O segundo momento que quero analisar refere-se a meados de 2004 quando, tendo o governo Lula já iniciado algumas ações em favor do uso do software livre, o presidente do Instituto Nacional de Tecnologia da Informação (ITI), Sérgio Amadeu, recebeu um pedido judicial de explicações (primeira instância para o início de um processo legal) remetido pela Microsoft. A empresa ameaçava processar

83 “Alternativa para a independência digital” Entrevista de Marcelo Branco. *Baguete*, http://www.baguete.com.br/entrevista.php?nome=marcelo_branco

84 “A visão do proprietário”, Entrevista de Eduardo Campos, *Baguete*, http://www.baguete.com.br/entrevista.php?nome=eduardo_campos

Amadeu por difamação pelas declarações dadas por ele à revista *Carta Capital*, quando comparou a atitude da empresa à “prática de traficante”. Ele fazia referência à política da empresa de oferecer gratuitamente seus softwares à escolas e à projetos de inclusão digital. Essa prática levaria à um “aprisionamento” do usuário, que só saberia utilizar a tecnologia proprietária.

Na entrevista concedida por Branco, no final de 2002, destaca-se a ênfase que o entrevistado dá para a relação entre tecnologia, desenvolvimento e a necessidade de uma espécie de sincronia entre o Brasil e os países centrais. Embora o software livre tenha uma ampla utilização também nos EUA (estão sediadas lá tanto a Free Software Foundation como a Open Source Initiative), os exemplos dados por Branco vêm principalmente do continente Europeu. Branco, fala em autonomia tanto do usuário como do Brasil, ao mesmo tempo que fala em relação com o exterior.

*“Tenho a certeza de que o próximo governo Federal vai proporcionar grandes oportunidades para o desenvolvimento de uma tecnologia nacional **autônoma** e com grande **intercâmbio** com a comunidade científica internacional. A comunidade internacional está muito ligada nas mudanças que estão acontecendo no Brasil e está pronta para ajudar. A questão do software livre não é apenas uma oportunidade para as empresas brasileiras e para o setor de informática buscarem uma liderança no cenário internacional, mas a grande e, talvez, a única oportunidade de combatermos a "exclusão digital". A inserção **passiva**, isto é, sem **domínio** da tecnologia e dos conteúdos da internet acarreta um aumento da **dependência** digital, ao invés de diminuir a **exclusão**. Com o software livre estamos em "**tempo real**" com a tecnologia do **primeiro mundo** e os desenvolvedores e as empresas brasileiras dominam totalmente essa tecnologia. Poderemos ser **sujeitos** dessa revolução digital e não meros **consumidores** de conteúdos e tecnologias do **hemisfério norte**.”*

É possível demarcar certos personagens dessa fala. Estão nela o Brasil e o “internacional”, que se desdobra em “primeiro mundo” e “hemisfério norte”. Com o “primeiro mundo”, que está disposto a “ajudar”, por meio da “comunidade científica”, é possível um “intercâmbio” “autônomo”, em que o “nacional” seja “sujeito”. Com o software livre e com a “ajuda” da comunidade internacional, o Brasil será como que acelerado para ficar “em tempo real”. Coloca-se aqui, então, uma relação entre “desenvolvimento” (pois trata-se do “desenvolvimento de uma tecnologia nacional”) e estar no passado ou no presente/futuro. Já do “hemisfério norte”, termo que faz lembrar a relação desigual entre Norte e

Sul, entre centro e periferia, o Brasil só poderá ser “consumidor”.

Também é muito interessante a relação que se estabelece entre o país, Brasil, e os indivíduos que serão objetos de programas de “inclusão digital”. Quando Branco fala de “inserção passiva”, pode estar falando tanto de alguém que se tornou um incluído digital como da inserção do país no cenário econômico internacional. Indivíduos e nação se misturam e um desenvolvimento que seja dependente (o que nos faz lembrar da teoria da dependência desenvolvida pelo ex-presidente Fernando Henrique Cardoso) pode ser criticado, falando-se tanto de programas de inclusão digital que usam software proprietário como de um certo papel a ser assumido pelo Brasil no mercado global.

Um pouco mais adiante na entrevista, Branco é questionado por sua participação em eventos internacionais. Ao elencá-los, fala da visita a cinco países da América Latina (Cuba, Porto Rico, Argentina, Chile e Uruguai) e a dois países da Europa (Espanha e Itália).

Para responder às declarações de Branco, Eduardo Campos, da Microsoft, vai procurar dar ao software livre não essa característica de acelerador (ou uma espécie de atalho) para o desenvolvimento, mas procura significá-lo com o “atraso”, assemelhando-o a políticas estabelecidas no passado no Brasil (a reserva de mercado para a informática) que, na tentativa de estabelecerem uma autonomia para o país, o teriam condenado ao “atraso”, à uma espécie de descompasso com o resto do mundo. Assim, Campos tenta mostrar que só há um caminho para o desenvolvimento e esse caminho é a integração (sem questionar se ela será passiva ou não).

Como a entrevista é uma réplica a outra, Campos, logo no início, procura estabelecer algo que teria sido dito por Branco e, a partir dessa idéia sobre o que teria sido dito, fará suas afirmações. Segundo ele o “objetivo é oferecer aos leitores do *Baguete* [o website que publicou ambas as entrevistas] uma visão mais ampla, baseada em dados importantes sobre a indústria brasileira de software que não foram levados em consideração e os impactos da obrigatoriedade do uso desta ou daquela tecnologia pelo governo”. Branco, quando foi entrevistado, acabara de deixar um cargo na

Universidade Estadual do Rio Grande do Sul. Descrevendo-se como portador de “dados importantes sobre a indústria brasileira”, Campos se coloca como alguém do mercado enquanto Branco seria alguém sem essa experiência⁸⁵.

“Não existe uma crítica ao que foi falado, mas sim ao que não foi. O entrevistado não esclarece como a adoção obrigatória de uma única tecnologia no âmbito governamental geraria benefício para o país. Na verdade, o efeito é o contrário. Seriam criadas as condições ideais para o surgimento de uma nova reserva de mercado; a mesma que no passado aprisionou nosso país em 30 anos de atraso tecnológico, na qual apenas software e hardware nacional podiam ser comercializados no país, incluindo-se aí as compras do governo. A reserva levou o PC a custar US\$20.000, mais que muitos carros, e a maioria dos programas nacionais a serem cópias fiéis de programas importados, porém com nomes nacionais, desrespeitando direitos de propriedade intelectual, que no mundo todo, foram os grandes responsáveis pelo fantástico desenvolvimento da indústria de tecnologia. Assim como nas propostas de adoção obrigatória de open source pelo governo, na época tínhamos tecnologia nacional autônoma, com subsídios, proteção da lei federal e intercâmbio com a comunidade científica internacional.”

Ao comparar as políticas em benefício ao uso de software livre com a reserva de mercado, Campos liga uma ação do governo no sentido de orientação do mercado – e, por consequência, o próprio software livre – a um fato do passado recente bastante ligado à idéia de déficit tecnológico do Brasil frente aos países centrais. Ao mesmo tempo, faz com que essas políticas signifiquem uma ameaça de aumento muito grande no preço dessas tecnologias. Sem estabelecer a formulação “o software livre significa o desrespeito à propriedade intelectual”, faz com que essa associação seja lembrada, pela relação de equivalência que estabeleceu entre políticas pelo software livre e reserva de mercado que, segundo ele, desrespeitaram a propriedade intelectual.

Falando sempre em “*open source*”, Campos lembra que na época da reserva de mercado havia “intercâmbio com a comunidade internacional”, porém, isso não teria bastado para fazer da política de reserva um sucesso. Desse modo, acaba colocando que, para o Brasil, não bastará a comunidade internacional para atingir o desenvolvimento, será necessária a integração econômica com os mercados.

Em seguida, respondendo a uma outra questão, Campos descreverá o mercado como algo

⁸⁵ Em outro momento, na mesma entrevista, Campos também diz: “*Na minha opinião, apenas as pessoas cujo salário não depende diretamente do open source, é que conseguem defendê-lo tanto sob o ponto de vista de adoção empresarial*”

complexo e ao mesmo tempo incontrolável. Estabelecendo uma analogia com a natureza (“ecossistema”), dará a impressão de que ele é algo que não deve ser tocado, sob o risco de se alterar seu equilíbrio natural.

“O mais importante é cadeia de valor que existe hoje, essa cadeia é tão ampla e complexa que pode ser considerada um ecossistema digital. Segundo estudo da PriceWaterhouseCoopers, são mais de 145 mil brasileiros trabalhando direta ou indiretamente em uma indústria robusta, que movimenta anualmente mais de US\$ 3 bilhões em vendas de software e serviços, recolhendo cerca de R\$ 600 milhões em impostos. Além disso, o fator de geração de empregos indiretos na indústria de tecnologia brasileira é de 1 para 5. Ou seja, para cada emprego direto, outros 5 indiretos são gerados.

(...) Geração de empregos com o open source, desenvolvimento sustentado, é muito bonito na teoria. Na prática, o modelo não se sustenta.”

Ao afirmar a existência desse ecossistema e descrevendo sua importância a partir dos empregos que gera, fica a ameaça de que o desequilíbrio, a tentativa de controle, implicará na perda de empregos para milhares de pessoas. A metáfora biológica, seja com o corpo humano (“viral”, “câncer”, como vimos no segundo capítulo), seja com o mundo natural, aparece novamente um pouco mais tarde: “O resultado é que o open source **mata** a cadeia de valor de uma indústria **saudável** e ninguém ganha nada com isso.”

É interessante também, quando o entrevistador pergunta “qual a sua visão particular sobre a questão do software livre?”, com esse “particular” apontando para uma opinião pessoal e não da empresa, Campos responde:

“Minha visão particular é que, como tudo na vida, o software livre tem suas vantagens e desvantagens. A vantagem que mais admiro é a comunidade, apenas deixando claro que nem todo mundo tem boas intenções e vai criar programas e patches seguros de graça.... Mas é inegável que o conceito de "usuários ajudam usuários" contribui para a evolução da tecnologia. A maior desvantagem está no modelo de negócios. Ele destrói a cadeia de valor. (...)”

Mesmo ao ressaltar um valor universal contra o qual é difícil se pronunciar (o espírito comunitário, o auxílio ao próximo), Campos semeia a desconfiança e dá mostras de um realismo, de um desencanto de alguém que não é ingênuo ou sonhador. A confiança não poderia ser dada a qualquer

um, pois o mundo real, se é um ecossistema, é também uma selva cheia de perigos.

E, ao falar da ajuda mútua da comunidade, Campos não classifica o desenvolvedor independente, que desenvolve tecnologia também como um hobby (idéia sempre associada à comunidade do software livre), como um produtor. O desenvolvedor de software livre continua sendo um “usuário”, mesmo que possa ajudar outro “usuário”. O produtor é a empresa (o lugar da produção do valor é a empresa) e nunca um indivíduo ou um coletivo não profissionalizado.

Mas esse debate, entre Microsoft e o software livre brasileiro, ganhou uma intensidade muito maior (pelo menos do ponto de vista da repercussão junto à imprensa) no início de junho de 2004. O estopim foi uma matéria da revista *Carta Capital*, intitulada “O Pinguim Avança”⁸⁶, em que o presidente do ITI, Sérgio Amadeu, acusou a empresa de “prática de traficante”.

A Carta Capital ocupa um nicho bastante peculiar de mercado. Ao mesmo tempo que a publicação tem um bom trânsito entre os setores mais progressistas da sociedade – nas eleições presidenciais de 2002 ela assumiu-se favoravelmente à eleição do presidente Lula, fato raro na imprensa brasileira -, ela dirige-se principalmente aos empresários e executivos, com uma quantidade importante de informações sobre investimentos.

A reportagem da revista tratou da disseminação do uso de softwares livres no mercado corporativo, ressaltando a possível diminuição de custos para as empresas e as vantagens técnicas derivadas da livre modificação do código (as empresas poderiam adaptá-lo conforme suas necessidades). Para isso, foram entrevistados empresários que já usam GNU/Linux (tratado como “Linux”), empresários do setor de hardware, executivos da Microsoft e o presidente do ITI, Amadeu.

Quero, então, ressaltar alguns pontos da matéria, em especial as falas de Amadeu reportadas. Mas primeiro quero destacar a descrição sobre o que seriam os “sistemas abertos”, que o nome usado no texto:

86 Marques Marineide. “O Pinguim Avança”, em *Carta Capital*,
http://cartacapital.terra.com.br/site/exibe_materia.php?id_materia=1324

“Ao contrário dos chamados softwares proprietários, cujo uso está baseado no pagamento de licenças, os sistemas abertos, como o próprio nome diz, dão livre acesso ao código-fonte, conjunto de comandos que forma um determinado programa. Isso permite ao usuário conhecer exatamente o que tem dentro do software e modificá-lo se assim desejar. É sonho de todo nerd conhecer, por exemplo, o código-fonte do Windows, da Microsoft, um segredo tão cobiçado quanto a fórmula da Coca-Cola.”

Para alterar um software não basta conhecer o seu código-fonte, é preciso ter a permissão legal para isso. Ao descrever o “sistema aberto” como um sistema com o código-fonte em exposição, a reportagem reúne as definições de software livre, open source (conforme a OSI) e código-aberto (não pela definição da OSI, mas o sistema em que o código pode ser lido) a essa única categoria. Quase no final da matéria o termo “software livre” é usado por um dos entrevistados. Mas, como está em um parágrafo em que é referido como sinônimo de “Linux”, por consequência também é tomado como sinônimo de “sistema livre”.

Voltando ao trecho acima, este descreve um objeto de desejo (“o código-fonte do Windows”), um personagem (“o nerd”) e, assim, faz com que se imagine que a motivação para a criação dos sistemas livres (pelos nerds) tenha sido o desejo de conhecer/ler/dominar o código-fonte de um sistema operacional.⁸⁷

No parágrafo seguinte, aparecem que vantagens a livre alteração do código do sistema pode trazer para os empresários:

“A possibilidade de adotar um sistema que pode ser adaptado ao seu negócio, ou customizado, na linguagem gerencial, aliada ao menor custo em relação aos sistemas proprietários, seduz um número cada vez maior de empresas”

Em seguida são descritos os benefícios que a adoção do GNU/Linux trouxe para a rede de supermercados Carrefour e, completando a fala de seu entrevistado, o repórter traz novamente a Microsoft.

⁸⁷ Ao fazer uma pequena digressão para contar a história do surgimento do Linux, o repórter volta a usar a imagem do “sistema para nerds”. Ao mesmo tempo, atribui a criação da licença livre a Linus Torvalds e não a Richard Stallman: “Criado em 1991 por Torvalds, na época estudante de Ciência da Computação da Universidade de Helsinque, e tendo como mascote um pingüim, o Linux fez a alegria dos nerds, graças à possibilidade de acesso ao código-fonte. Em pouco tempo, despertou a atenção das empresas. Para evitar que alguém se apropriasse das melhorias e fizesse uma versão fechada, Torvalds criou uma licença especial de uso que proíbe que o código original ou qualquer modificação feita com base nele seja fechada”.

“Ele [o entrevistado] destaca ainda o ganho, embora não mensurável, de independência de fornecedor, ou seja, não ficar preso a um único fabricante, uma vez que o Linux é desenvolvido por várias empresas. O gerente não cita nomes, mas quem conhece um pouco do universo de bits e bytes sabe que ele está falando da Microsoft, dona do Windows, (...)”

Continuando, e antes de dar a palavra ao executivo da Microsoft, o repórter descreve a empresa com adjetivos que repetirá por toda a matéria: a empresa é “gigante” e seu dono um “milionário”:

*“O avanço do Linux no mercado corporativo não é desprezado pela Microsoft, mas a **gigante** fundada pelo **milionário** Bill Gates vê os sistemas abertos como aplicações de nicho”*

Um pouco mais adiante no texto a palavra gigante será usada também para adjetivar duas empresas de hardware e de software que já prestigiam o “sistema aberto”. a formulação é a seguinte:

*“ À medida que cresce o número de empresas que utilizam softwares abertos, aumenta a oferta de fornecedores desses sistemas. **Gigantes mundiais** de hardware e software como HP e IBM perceberam que não dá para ignorar essa expansão e incorporaram a plataforma às suas máquinas.”*

No último parágrafo da matéria a palavra gigante será usada novamente duas vezes, ambas para qualificar a Microsoft. Vejamos as formulações:

*“O apelo da **gigante americana** não comove Amadeu, que promete colocar mais pimenta na discussão(...). O sucesso do software livre só o tempo vai provar, mas a briga com a **gigante** de Bill Gates será travada no presente.”*

Embora a Microsoft esteja sendo insistentemente classificada como “gigante”, a imagem será a de um gigante em decadência, que em breve deverá ser vencido. Vimos que as “gigantes mundiais” do hardware e do software (HP e IBM) já usam o software livre. Apesar de serem descritas como gigantes das duas áreas, do hardware e do software, o leitor um pouco habituado ao uso de produtos de informática sabe que a especialidade dessas empresas, hoje, é o hardware (a IBM é mais lembrada pelos seus microcomputadores e a HP é muito conhecida no mercado de impressoras). Temos, então, o gigante (americano) do software, Microsoft, perdendo a exclusividade no uso de seus produtos por parte de outros gigantes (estes “mundiais”), HP e IBM.

E na relação com o presidente do ITI a Microsoft também aparece em desvantagem. Ela “apela”

e esta ação não é dirigida ao governo brasileiro e nem mesmo à instituição hierarquicamente acima do ITI, o Ministério da Casa Civil, a Microsoft “apela” a Amadeu e este, como um David prestes a vencer Golias, não se “comove”, não mostra misericórdia – e muito menos se sente ameaçado pelo poder do gigante.

A imagem belicosa das ações de Amadeu está também em todo o texto que acompanha e descreve sua fala. Vejamos:

*“Ex-funcionário da prefeitura petista de São Paulo, Amadeu decidiu **comprar uma guerra** com os maiores fabricantes de sistemas operacionais fechados, a Microsoft principalmente. Para o presidente do ITI, o modelo de software proprietário inviabiliza a informatização do País, dada a necessidade de pagamento de licenças a empresas estrangeiras. “Um país pouco informatizado como o Brasil pagou US\$ 1,1 bilhão em licenças de software em 2002. O valor só tende a subir”, diz. “É uma **guerra do ponto de vista tecnológico**”, afirma, enumerando **China e Índia como aliados do Brasil na batalha para “libertar a Esplanada dos Ministérios da reserva de mercado do software proprietário”**”.*

Na defesa do software livre, Amadeu não poupa críticas à Microsoft, a quem acusa de “prática de traficante” por oferecer o sistema operacional Windows a alguns governos e prefeituras para a instalação em programas de inclusão digital. “Isso é presente de grego, uma forma de assegurar massa crítica para continuar aprisionando o País.”

Amadeu “compra uma guerra”, “acusa”, “não poupa críticas” e luta contra “traficantes” e contra a empresa quer continuar “aprisionando o país”. Nessa batalha, ele (e o Brasil) tem “aliados”, sendo estes dois países de grande extensão territorial, bastante populosos e freqüentemente anunciados como emergentes no cenário econômico mundial: China e Índia. Também é interessante a inversão que ele faz das acusações anteriores da empresa de que qualquer ação em favor do software livre seria “reserva de mercado”. A “reserva de mercado do software proprietário”, segundo Amadeu, está é sendo combatida pelo governo e não ressuscitada.

Em seguida à publicação dessa reportagem deu-se a ação legal da Microsoft, que teria respondido às acusações de “prática de traficante”. No entanto, essa analogia com o tráfico é uma das mais frequentes (junto com a do Presente de Grego) usadas pela comunidade da Tecnologia da Informação para qualificar as doações da empresa. Até mesmo o presidente da Sun Microsystems, uma

das maiores empresas de software do mundo, Scott McNealy, já a utilizou. Ao comentar para a revista estadunidense *Wired*⁸⁸ a dependência dos usuários britânicos dos produtos Microsoft para acessar a página do governo, que só funcionavam com o navegador da companhia, McNealy afirmou: "A primeira dose de heroína é sempre de graça".

A ação judicial foi logo interpretada pela comunidade software livre brasileira como uma tentativa de intimidação do governo⁸⁹, dado que a analogia não era nova. Uma outra hipótese para a ação pode ser levantada pela análise que fizemos da matéria. Mais do que ser comparada a traficantes, a empresa foi retratada como um gigante de joelhos diante de um David obstinado e atrevido. A resposta judicial não se dirigiu à revista ou ao repórter que descreveram essa imagem, mas a Amadeu, aquele que "comprou uma guerra" e enumera "seus aliados"⁹⁰.

Na imprensa, a Microsoft evitou comentar a ação judicial contra Amadeu. A ação foi anunciada em 14 de junho um pouco menos de duas semanas depois da realização do V Fórum Internacional de Software Livre, em Porto Alegre, período em que o presidente da filial brasileira da empresa fez críticas mais gerais. É interessante reproduzir o trecho de uma outra matéria publicada na imprensa que reproduz essas críticas⁹¹:

"A Microsoft está lançando uma ofensiva política no Brasil para tentar barrar o avanço do software livre. O presidente da Microsoft Brasil, Emílio Umeoka, deixou claro o tom dessa ofensiva ao dizer que a decisão do governo brasileiro de apoiar o software livre nos computadores do setor público está sendo "influenciada pela ideologia". Segundo Umeoka, a escolha do governo Lula pode levar o país na "direção errada", na questão dos programas de computador. Ou seja, na direção contrária aos interesses da Microsoft. "Se o país se fechar de novo - como fez quando protegeu o setor de tecnologia da informação (referência à Lei de Informática, adotada pelo governo brasileiro na década de 80 para proteger a indústria nacional) - daqui a 10 anos teremos uma posição dominante em algo insignificante", argumentou o executivo."

88 "MS Monopolizes U.K. Gov't Site" in *Wired*, 01/06/2001 <http://www.wired.com/news/business/0,1367,44186,00.html>

89 Branco, Marcelo. "Microsoft tenta intimidar o Governo Brasileiro". <http://portal.softwarelivre.org/news/2479>

90 Após o início da ação da Microsoft surgiu um grande movimento de apoio à Amadeu, organizado pela comunidade nacional e internacional. Amadeu não respondeu ao questionamento judicial e empresa decidiu não dar prosseguimento à ação.

91 Weissheimer, Marco Aurélio. "Microsoft lança ofensiva política contra o uso de software livre", *Agência Carta Maior*, 04/06/2004 <http://agenciartamaior.uol.com.br/agencia.asp?coluna=reportagens&id=1896>

Antes mesmo da ação a empresa já anunciava que o objetivo era político. De fato, o objetivo é silenciar a fala daqueles que afirmam a existência de uma desigualdade. Ao dizer que a política do governo é “influenciada pela ideologia” ela está lembrando a história socialista do partido que ocupa o governo e procura deslegitimá-la como um objetivo que não caberia à função do Estado contemporâneo, que pode ser tanto regular a economia como combater inimigos associados ao capitalismo e aos Estados Unidos.

Como represália/consequência dessa atitude do governo, a empresa acena com esse retorno ao passado, usando novamente da imagem de “atraso” derivada da política da reserva de mercado/”Lei de Informática”. Adotar o software livre seria “fechar” o país, levá-lo à “direção errada” - aqui surge a imagem de que o desenvolvimento segue um certa linha, um caminho único, no qual os líderes estão à frente. Ser “dominante” no software livre seria não “dominar” ninguém, não estar à frente de ninguém, pois essa tecnologia não será usada por nenhum país relevante.

Conclusão

A discussão em torno do uso do software livre no países periféricos, em nações cuja diferença de renda é mais acentuada e cuja posição no cenário político-econômico mundial é menos vantajosa, ganha certas especificidades. Essas características únicas podem servir de hipótese explicativa para que, diferente do resto do mundo, o termo software livre tenha um emprego muito maior do que o termo *open source*. Mais do que a inexistência da possível confusão entre os termos free, livre e grátis no português e no espanhol, ganha força como explicação para a popularização do termo software livre a ênfase que esse movimento coloca na liberdade de modificação, alteração e distribuição do código, em lugar de aspectos que seriam apenas técnicos.

Embora o uso do software livre ou open source como ferramenta para a diminuição do poder da

Microsoft também seja um fator importante para o movimento em escala mundial, na periferia isso ganha outros aspectos. É possível descrever a seguinte relação:

Software livre	Software proprietário
usuário autônomo (produtor)	usuário “aprisionado” (consumidor)
país independente (produtor)	país dependente (consumidor)

Nesse jogo, a figura do usuário dependente de soluções proprietárias funciona como metáfora para a própria inserção subordinada do país na ordem econômica mundial. O discurso em favor do software livre se torna, então, também um discurso em favor de um desenvolvimento autônomo do país. O *open source*, ao enfatizar apenas as vantagens técnicas, sem dar relevo à transformação do usuário em produtor de seu próprio software, acaba tendo menos penetração. No Brasil, enquanto há núcleos do Projeto Software Livre em todos os estados, o termo *open source*/código aberto é utilizado apenas pelas empresas que atuam nesse mercado (e algumas misturam as duas expressões).

Nessa relação de dependência e dominação imaginada (não cabe aqui discutir se real ou não), a Microsoft, fortemente identificada com os Estados Unidos, será o grande inimigo, ganhando a imagem de potência dominadora, agressiva, monopolista.

Em resposta, a empresa procura mobilizar a imagem do “atraso”, tentando mostrar que suas soluções são as mais usadas (praticamente únicas) no mundo e que isso continuará por muito tempo. Qualquer tentativa de construção de uma alternativa há de fracassar por ser utópica e não responder à “realidade” do mercado.

Nessa resposta, tem grande relevância a idéia de “atraso”, uma perspectiva em que a injustiça social nos países do Terceiro Mundo se mistura com a idéia de que a periferia vive em uma espécie de passado em relação aos países ricos. Essa idéia não está somente no discurso da empresa, mas também se mostra no discurso dos defensores do software livre. Para eles, ainda se misturam duas concepções sobre o papel do software para vencer esse atraso: uma afirma o software livre como a tecnologia do futuro, aquela que ainda não é a mais utilizada mas, em questão de tempo, será; a outra afirma a idéia

de um caminho alternativo, imaginando que, para alcançar o “desenvolvimento” não necessariamente será preciso percorrer um caminho único, podendo ser construído um desenvolvimento à moda da periferia.

Pensando a idéia de atraso de uma maneira mais ampliada, Marta Mourão Kanashiro e eu, no artigo “Ciência, Comunicação e Sociedade no Brasil, a narrativa do déficit”⁹², procuramos refletir, a partir de textos de divulgação científica e de documentos de apoio a essas iniciativas, como a idéia de déficit de conhecimentos do público leigo, para quem a divulgação científica é dirigida, se mistura com a idéia de que os países da periferia, com grande diferença social, estão em déficit em relação aos países ricos do centro. Nossa conclusão foi que nessa idéia de déficit entre os países repousava em um imaginário sobre o atraso, em que os países pobres seriam tomados como que estando no passado e a tecnologia teria um papel relevante para uma espécie de aceleração, materializada no conceito de desenvolvimento. Tanto naquele texto como neste, não se trata de negar ou afirmar a tecnologia como capaz de desempenhar esse papel, mas de analisar como essas idéias estão presentes.

Naquele artigo, mostramos como alguns setores (no caso as empresas de biotecnologia), que defendem a adoção de certos modelos tecnológicos os quais possuem os direitos de propriedade intelectual, afirmam que esses modelos são os únicos possíveis para o desenvolvimento, estando a busca de alternativas fadada ao fracasso. Naquele caso, assim como para a discussão em torno do software livre, o fracasso significaria a condenação ao “atraso”.

Como pudemos acompanhar, a idéia de “atraso” também está presente aqui, seja no medo de que ele se perpetue ou se aprofunde, seja na concepção de que é preciso vencê-lo. Nesse sentido, a Microsoft funciona tanto como a empresa que perpetua essa condição, “aprisionando” os usuários ou o Estado aos seus padrões, quanto como a empresa de quem o distanciamento pode significar o “isolamento” ou a “insignificância” - como nas palavras do presidente da filial brasileira da empresa.

92 Kanashiro, Marta M. e Evangelista, Rafael “Ciência, Comunicação e Sociedade no Brasil, a narrativa do déficit” in *JCom* (3) 4, Dezembro 2004, http://jcom.sissa.it/focus/foc030402_or.pdf

Capítulo 5 - Conclusões gerais

Após o percurso de análise estabelecido até o momento, acredito que seja interessante reformular alguns dos pontos colocados para que outras conclusões possam ser estabelecidas.

Acredito que, pela análise do texto das duas licenças que são mais utilizadas pelos softwares livre e pelos softwares proprietários, tenha sido possível marcar o caráter contraditório de ambas, sendo a GPL, como símbolo das licenças livres, uma negação das restrições impostas pelas licenças proprietárias.

A característica mais importante dessa contradição parece ser o fato de as licenças livres permitirem uma assimilação fundamental entre a figura do proprietário e do usuário. No mercado de informática, há outras licenças que permitem o uso (chamada freeware) e a redistribuição não comercial (chamada shareware). No entanto, nenhuma delas foi capaz de gerar um movimento social, muito menos com a força do movimento software livre. As duas permanecem sendo apenas uma estratégia de distribuição e promoção no mercado de software.

A partir do conceito de política que aqui utilizei - como formulado por Eduardo Guimarães - é possível dizer que as licenças livres, ao permitirem que o código seja modificado e redistribuído por qualquer usuário, está pleiteando um novo ordenamento de relações na esfera da tecnologia da informação. Nessa nova ordem, usuários e desenvolvedores podem se igualar (mesmo que isso nunca aconteça), a distinção entre produtor e consumidor torna-se circunstancial. Ao consumidor passa a ser facultado o direito de também produzir. Ao existir, a licença livre afirma que a desigualdade não

precisa existir e que é possível um novo ordenamento social em que ela não existe.

Ao mesmo tempo, por contradizer uma norma tão forte da sociedade capitalista como a propriedade privada dos meios de produção, as licenças livres parecem gerar um movimento tão forte contra elas. Comparadas a patologias (“câncer”, “vírus”) elas seriam como enfermidades a atingir a sociedade, que é freqüentemente comparada com organismos vivos. A doença afetaria as funções normais do ser vivo, desordenando seus órgãos e afetando suas funções vitais, podendo causar a morte, ou seja, a dissolução daquela ordem social. Os defensores do software proprietário, dessa forma, protestam contra uma nova ordem social em que o papel de produtores deixaria de ser desempenhado com exclusividade por alguns, aqueles que detêm o direito de alterar o código-fonte.

Mas como explicar, então, que algo tão subversivo, classificado na terra do capitalismo como “comunista” pode alcançar tanto sucesso, inclusive dando origem a empresas dos Estados Unidos e sendo adotado e incentivado por grandes corporações?

Para isso, o movimento software livre teve que ser refundado, resignificado sobre outras bases. O open source/código aberto, sem enfatizar a indistinção entre produtores e consumidores/usuários, deu relevância a uma nova prática de produção de software, só possível a partir dos anos 1990, com a criação da internet. Nessa prática, a rede passou a funcionar como uma metáfora do mundo natural, em que os códigos mais competentes/melhor escritos/mais inovadores, encontravam programadores dispostos a aplicá-los e a melhorá-los. A relação mais flexível com a propriedade passou a ser justificada pela melhoria técnica, pela seleção natural estabelecida na internet. “Liberar cedo, liberar frequentemente”, como afirmou Eric Raymond, tornou-se a regra para garantir o mínimo de tempo perdido no desenvolvimento de um software que provavelmente não “sobreviveria” na selva do mercado e para garantir a adoção das melhores soluções de desenvolvimento proporcionada pela profusão de olhos a ler o código. Ao mesmo tempo, a rede passou a funcionar como a experiência de vida que melhoraria o código (tornando-o mais forte).

A metáfora com o mundo natural encontrou grande ressonância nos ouvidos das empresas e nos países centrais. Para aqueles que estão em condição privilegiada em termos sociais, entender sua condição como derivada de uma maior capacitação, inteligência ou habilidade, parece ser muito mais adequado do que atribuir a diferença a injustiças sociais. O mundo natural como metáfora da organização social (os mais fortes sobrevivem e prosperam, enquanto os mais fracos morrem) é algo bastante recorrente na sociedade estadunidense.

No discurso *open source*/código aberto, a persistência da existência da figura do usuário/consumidor pode ser verificada. Para as empresas, interessadas no negócio software (seja ele a venda de código aperfeiçoado ou serviços), a figura do consumidor precisa continuar a existir. A possibilidade de alteração do código tornou-se algo não essencial, embora ainda desejável. Mais importante que a não-distinção entre produtor/usuário serão os milhares de olhos capazes de inspecionar o código e fazê-lo evoluir para algo melhor. Essencialmente, na licença, nada muda, mas surge uma nova instituição (a Open Source Initiative) e novos advogados que enfatizam as vantagens técnicas e não a alteração na ordem social.

Já na periferia, quem encontrou eco foi o discurso em favor do software livre. A figura do usuário que pode se tornar um programador foi expandida para a idéia do país que pode deixar de ocupar uma posição subalterna, de consumidor de tecnologia, para tornar-se autônomo e produtor. A dependência de produtos vindos do exterior e a injustiça social, misturada com o fato de se tratar de uma discussão sobre tecnologia, são entendidos como “atraso”, como se a incapacidade de produzir as mercadorias vendidas pelos países centrais significasse que esses países estão em um tempo diferente, como se estivessem no “passado”. Nesse sentido, uma tecnologia aberta (sem segredos) e de custo baixo (sem custo de licenciamento) aparece como uma possível alternativa de desenvolvimento. Algumas vezes, essa alternativa significará construir um caminho totalmente novo (independente, autônomo), outros, esse pode ser um “atalho”, um anúncio do que será o futuro.

Ao estabelecer esse percurso de análise, este trabalho procurou mostrar, embora os produtos que estão em debate na discussão sobre o software livre sejam tidos como técnicos, a adoção de uma ou de outra tecnologia tem profundas consequências sociais que se refletem e se referem ao imaginário sobre um ordenamento do mundo, principalmente relacionado à distinção ou não-distinção entre quem produz e quem só pode consumir. Como vimos, há também outros elementos, contudo, a proposição de uma igualdade de direitos para a alteração do código tem caráter fundamental e deriva-se em outras formas aqui discutidas. Esse imaginário sobre o ordenamento do mundo inscreve-se e pode ser observado na linguagem e nos sentidos, como foi demonstrado neste trabalho.

Epílogo

Compartilhando textos

Cabem aqui algumas observações sobre a dinâmica de compartilhamento de códigos instaurada pelo movimento software livre. Ter a liberdade para alterar o código recebido de outro seria algo muito difícil, talvez até impossível, se os programas de computador que são compartilhados estivessem apenas em sua forma executável.

O que significa isso? O programa executável é aquele que já está pronto para “rodar” no computador. No Windows, por exemplo, são aqueles arquivos que terminam com a extensão “.exe”. Ler o código desses programas é impossível para um humano, pois ele está pronto para ser entendido pela máquina.

Por isso, ao disponibilizar os programas para que outros possam alterá-lo as licenças livres estipulam que, junto com o programa executável, deve estar disponível também o código-fonte do programa. A GPL, por exemplo, diz:

“[Para redistribuir você deve] Incluir o código-fonte correspondente completo, passível de leitura pela máquina, o qual terá de ser distribuído de acordo com as Cláusulas 1 e 2 acima, em um meio ou mídia habitualmente usado para intercâmbio de software; (...)”

O código fonte de um programa é a linguagem usada pelo programador para escrever aquele software. É lá que ele estipula o que o programa vai fazer. Depois de descrever essas funções em uma certa linguagem, o programador aplica um compilador sobre o código, que nada mais é do que um

outro software, capaz de “traduzir” o código-fonte para a máquina. Ou seja, o código-fonte é uma linguagem em que tanto o programador é capaz de escrever e ler (e ele lê diretamente, já imaginando o que o software fará), como a máquina (por meio de um software de “tradução”, o compilador) é capaz de ler.

O que o movimento software livre instituiu foi, então, um acordo em que todos os programadores comprometem-se a estabelecer uma comunicação inteligível entre si mesmos. Ao invés de apenas disponibilizarem o produto de seu trabalho na forma de uma linguagem que só poderia ser entendida pelas máquinas, o movimento software livre estabeleceu que esses textos, escritos nas mais diversas linguagens de programação, deveriam poder ser lidos por todos.

Mais, os códigos-fontes dos programas são, em geral, comentados. Essas observações podem ser descrições das funções que certos comandos pretendem realizar como podem ser comentários sobre produtos concorrentes ou o que mais o programador desejar. Tanto o código dos softwares livres como o código dos softwares proprietários tem comentários mas a diferença é que os comentários dos códigos livres são públicos e essenciais para que o desenvolvimento continue⁹³. Não é raro ouvir de programadores que um código “está sujo” ou é “bem escrito” ou ainda que “está mal formatado, desorganizado”.

Para explicar o que são os códigos-fonte, Richard Stallman estabelece uma analogia com receitas de cozinha. Segundo ele, as licenças proprietárias impõem restrições que seriam semelhantes a impedir que um vizinho passe uma receita de um bolo, por exemplo, a outro. O movimento fundado por Stallman fez justamente o contrário, criou uma regra em que, para se colocar um produto na prateleira de produtos “livres” de um supermercado, é preciso que ele contenha as instruções sobre

93 No início de 2004 foi noticiado que partes do código do Windows teriam vazado, sendo disponibilizados na internet. O trecho de uma notícia sobre esse fato aponta que o código tinha comentários sobre falhas do produto e sobre software da concorrência. Rodrigues, Giordani. “MS admite que código do Windows vazou na Web”, publicada em http://www.aeontech.com.br/noticias_seguranca_codigo_windows.html, em 13/02/2004:

“Entre as informações mais curiosas que ele encontrou nas linhas de programação estão os comentários dos programadores e citações a marcas concorrentes, como Netscape e Unix. Também foram encontradas várias referências a bugs, como a seguinte: “// Ensure the caller has read access. // BUGBUG : Is this necessary?””

como fazer, para que todos possam se tornar cozinheiros.

7. Bibliografia

Barbrook, Richard e Cameron, Andy. *Californian Ideology*, 1995. Em:
http://www.alamut.com/subj/ideologies/pessimism/califIdeo_I.html

Gadet, Françoise; Hak, Tony (org). *Por uma análise automática do discurso*. Campinas, 1997, Ed. da Unicamp.

Guimarães, Eduardo. *Semântica do Acontecimento*. Pontes, Campinas, 2002

Gurovitz, Hélio. *Linux*. Coleção Para saber mais. Ed. Abril, 2003.

Mundie, Craig. “Prepared Text of Remarks by Craig Mundie, Microsoft Senior Vice President - The Commercial Software Model” em *Microsoft*, site institucional. Visualizado em 15/12/2004
<http://www.microsoft.com/presspass/exec/craig/05-03sharedsource.asp>

Kanashiro, Marta M. e Evangelista, Rafael “Ciência, Comunicação e Sociedade no Brasil, a narrativa do déficit” in *JCom* (3) 4, Dezembro 2004, http://jcom.sissa.it/focus/foc030402_or.pdf

Orlandi, Eni. *As Formas do Silêncio*. Campinas, Editora da Unicamp, 1992.

Orlandi, Eni. *Interpretação. Autoria, leitura e efeitos do trabalho simbólico*. Ed. Vozes, 1996.

Orlandi, Eni. *Análise de discurso: princípios e procedimentos*. Campinas, Pontes, 2003

Pêcheux, Michel. *Semântica e Discurso: uma crítica à afirmação do óbvio*. Campinas, 1997, E. da Unicamp.

Raymond, Eric. *A Catedral e o Bazar*. Versão traduzida par o português por Erik Kohler.
<http://www.geocities.com/CollegePark/Union/3590/pt-cathedral-bazaar.html>

Rifkin,Jeremy. *A era do acesso*. Makrom Books, São Paulo.

Söderberg, Johan. “Copyleft vs. Copyright: a marxist critique” in *First Monday*. Fevereiro, 2002.
http://www.firstmonday.org/issues/issue7_3/soderberg/

Stallman, Richard. *Free Software, Free Society*. Disponível em: <http://www.notabug.com/2002/rms-essays.pdf>

Williams, Sam. *Free as in Freedom: Richard Stallman's Crusade for Free Software*. EUA, março, 2002. Disponível em <http://www.oreilly.com/openbook/freedom/>

8. Anexos

a. General Public License (GPL)

Versão 2, Junho de 1991 Direitos Autorais Reservados © 1989, 1991 Free Software Foundation, Inc.
59 Temple Place, Suite [conjunto] 330, Boston, MA [Massachusetts] 02111-1307 USA [Estados Unidos da América]

É permitido a qualquer pessoa copiar e distribuir cópias sem alterações deste documento de licença, sendo vedada, entretanto, qualquer modificação.

Introdução

As licenças da maioria dos softwares são elaboradas para suprimir sua liberdade de compartilhá-los e modificá-los. A Licença Pública Geral do GNU, ao contrário, visa garantir sua liberdade de compartilhar e modificar softwares livres para assegurar que o software seja livre para todos os seus usuários. Esta Licença Pública Geral é aplicável à maioria dos softwares da [Free Software Foundation](#) [Fundação do Software Livre] e a qualquer outro programa cujos autores se comprometerem a usá-la. (Em vez dela, alguns outros softwares da Free Software Foundation são cobertos pela Licença Pública Geral de Biblioteca do GNU). Você também poderá aplicá-la aos seus programas.

Quando falamos de software livre, estamos nos referindo à liberdade, não ao preço. Nossas Licenças Públicas Gerais visam garantir que você tenha a liberdade de distribuir cópias de software livre (e cobrar por isso se desejar), que receba código-fonte ou possa obtê-lo se desejar, que possa modificá-lo ou usar partes dele em novos programas livres; finalmente, que você tenha ciência de que pode fazer tudo isso.

Para proteger seus direitos, necessitamos fazer restrições que proíbem que alguém negue esses direitos a você ou que solicite que você renuncie a eles. Essas restrições se traduzem em determinadas responsabilidades que você deverá assumir, se for distribuir cópias do software ou modificá-lo.

Por exemplo, se você distribuir cópias de algum desses programas, tanto gratuitamente como mediante uma taxa, você terá de conceder aos receptores todos os direitos que você possui. Você terá de garantir que, também eles, recebam ou possam obter o código-fonte. E você terá a obrigação de exibir a eles esses termos, para que eles conheçam seus direitos.

Protegemos seus direitos através de dois passos: (1) estabelecendo direitos autorais sobre o software e (2) concedendo a você esta licença, que dá permissão legal para copiar, distribuir e/ou modificar o software.

Além disso, para a proteção de cada autor e a nossa, queremos ter certeza de que todos entendam que

não há nenhuma garantia para este software livre. Se o software for modificado por alguém e passado adiante, queremos que seus receptores saibam que o que receberam não é o original, de forma que quaisquer problemas introduzidos por terceiros não afetem as reputações dos autores originais.

Finalmente, qualquer programa livre é constantemente ameaçado por patentes de software. Queremos evitar o risco de que redistribuidores de um programa livre obtenham individualmente licenças sob uma patente, tornando o programa, com efeito, proprietário. Para impedir isso, deixamos claro que qualquer patente deve ser licenciada para o uso livre por parte de qualquer pessoa ou, então, simplesmente não deve ser licenciada.

Os exatos termos e condições para cópia, distribuição e modificação seguem abaixo.

TERMOS E CONDIÇÕES PARA CÓPIA, DISTRIBUIÇÃO E MODIFICAÇÃO

0. Esta Licença se aplica a qualquer programa ou outra obra que contenha um aviso inserido pelo respectivo titular dos direitos autorais, informando que a referida obra pode ser distribuída em conformidade com os termos desta Licença Pública Geral. O termo "Programa", utilizado abaixo, refere-se a qualquer programa ou obra, e o termo "obras baseadas no Programa" significa tanto o Programa, como qualquer obra derivada nos termos da legislação de direitos autorais: isto é, uma obra contendo o Programa ou uma parte dele, tanto de forma idêntica como com modificações, e/ou traduzida para outra linguagem. (Doravante, o termo "modificação" inclui também, sem reservas, a tradução). Cada licenciado, doravante, será denominado "você".

Outras atividades que não a cópia, distribuição e modificação, não são cobertas por esta Licença; elas estão fora de seu escopo. O ato de executar o Programa não tem restrições e o resultado gerado a partir do Programa encontra-se coberto somente se seu conteúdo constituir uma obra baseada no Programa (independente de ter sido produzida pela execução do Programa). Na verdade, isto dependerá daquilo que o Programa faz.

1. Você poderá fazer cópias idênticas do código-fonte do Programa ao recebê-lo e distribuí-las, em qualquer mídia ou meio, desde que publique, de forma ostensiva e adequada, em cada cópia, um aviso de direitos autorais (ou copyright) apropriado e uma notificação sobre a exoneração de garantia; mantenha intactas as informações, avisos ou notificações referentes a esta Licença e à ausência de qualquer garantia; e forneça a quaisquer outros receptores do Programa uma cópia desta Licença junto com o Programa.

Você poderá cobrar um valor pelo ato físico de transferir uma cópia, e você pode oferecer, se quiser, a proteção de uma garantia em troca de um valor.

2. Você poderá modificar sua cópia ou cópias do Programa ou qualquer parte dele, formando, dessa forma, uma obra baseada no Programa, bem como copiar e distribuir essas modificações ou obra, de acordo com os termos da Cláusula 1 acima, desde que você também atenda a todas as seguintes condições:
 - a. Você deve fazer com que os arquivos modificados contenham avisos, em destaque, informando que você modificou os arquivos, bem como a data de qualquer modificação.
 - b. Você deve fazer com que qualquer obra que você distribuir ou publicar, que no todo ou em parte contenha o Programa ou seja dele derivada, ou derivada de qualquer parte dele, seja licenciada como um todo sem qualquer custo para todos terceiros nos termos desta licença.
 - c. Se o programa modificado normalmente lê comandos interativamente quando executado, você deverá fazer com que ele, ao começar a ser executado para esse uso interativo em sua forma mais simples, imprima ou exiba um aviso incluindo o aviso de direitos

autorais (ou copyright) apropriado, além de uma notificação de que não há garantia (ou, então, informando que você oferece garantia) e informando que os usuários poderão redistribuir o programa de acordo com essas condições, esclarecendo ao usuário como visualizar uma cópia desta Licença. (Exceção: se o Programa em si for interativo mas não imprimir normalmente avisos como esses, não é obrigatório que a sua obra baseada no Programa imprima um aviso).

Essas exigências se aplicam à obra modificada como um todo. Se partes identificáveis dessa obra não forem derivadas do Programa e puderem ser consideradas razoavelmente como obras independentes e separadas por si próprias, nesse caso, esta Licença e seus termos não se aplicarão a essas partes quando você distribuí-las como obras separadas. Todavia, quando você distribuí-las como parte de um todo que constitui uma obra baseada no Programa, a distribuição deste todo terá de ser realizada em conformidade com esta Licença, cujas permissões para outros licenciados se estenderão à obra por completo e, conseqüentemente, a toda e qualquer parte, independentemente de quem a escreveu.

Portanto, esta cláusula não tem a intenção de afirmar direitos ou contestar os seus direitos sobre uma obra escrita inteiramente por você; a intenção é, antes, de exercer o direito de controlar a distribuição de obras derivadas ou obras coletivas baseadas no Programa.

Além do mais, a simples agregação de outra obra que não seja baseada no Programa a ele (ou a uma obra baseada no Programa) em um volume de mídia ou meio de armazenamento ou distribuição, não inclui esta outra obra no âmbito desta Licença.

3. Você poderá copiar e distribuir o Programa (ou uma obra baseada nele, de acordo com a Cláusula 2) em código-objeto ou formato executável de acordo com os termos das Cláusulas 1 e 2 acima, desde que você também tome uma das providências seguintes:
 - a. Incluir o código-fonte correspondente completo, passível de leitura pela máquina, o qual terá de ser distribuído de acordo com as Cláusulas 1 e 2 acima, em um meio ou mídia habitualmente usado para intercâmbio de software; ou,
 - b. Incluir uma oferta por escrito, válida por pelo menos três anos, para fornecer a qualquer terceiro, por um custo que não seja superior ao seu custo de fisicamente realizar a distribuição da fonte, uma cópia completa passível de leitura pela máquina, do código-fonte correspondente, a ser distribuído de acordo com as Cláusulas 1 e 2 acima, em um meio ou mídia habitualmente usado para intercâmbio de software; ou,
 - c. Incluir as informações recebidas por você, quanto à oferta para distribuir o código-fonte correspondente. (Esta alternativa é permitida somente para distribuição não-comercial e apenas se você tiver recebido o programa em código-objeto ou formato executável com essa oferta, de acordo com a letra b, acima).

O código-fonte de uma obra significa o formato preferencial da obra para que sejam feitas modificações na mesma. Para uma obra executável, o código-fonte completo significa o código-fonte inteiro de todos os módulos que ela contiver, mais quaisquer arquivos de definição de interface associados, além dos scripts usados para controlar a compilação e instalação do executável. Entretanto, como uma exceção especial, o código-fonte distribuído não precisa incluir nada que não seja normalmente distribuído (tanto no formato fonte como no binário) com os componentes principais (compilador, kernel e assim por diante) do sistema operacional no qual o executável é executado, a menos que este componente em si acompanhe o executável.

Se a distribuição do executável ou código-objeto for feita mediante a permissão de acesso para copiar, a partir de um local designado, então, a permissão de acesso equivalente para copiar o código-fonte a partir do mesmo local será considerada como distribuição do código-fonte, mesmo que os terceiros não sejam levados a copiar a fonte junto com o código-objeto.

4. Você não poderá copiar, modificar, sublicenciar ou distribuir o Programa, exceto conforme expressamente estabelecido nesta Licença. Qualquer tentativa de, de outro modo, copiar, modificar, sublicenciar ou distribuir o Programa será inválida, e automaticamente rescindirá seus direitos sob esta Licença. Entretanto, terceiros que tiverem recebido cópias ou direitos de você de acordo esta Licença não terão suas licenças rescindidas, enquanto estes terceiros mantiverem o seu pleno cumprimento.
5. Você não é obrigado a aceitar esta Licença, uma vez que você não a assinou. Porém, nada mais concede a você permissão para modificar ou distribuir o Programa ou respectivas obras derivativas. Tais atos são proibidos por lei se você não aceitar esta Licença. Conseqüentemente, ao modificar ou distribuir o Programa (ou qualquer obra baseada no Programa), você estará manifestando sua aceitação desta Licença para fazê-lo, bem como de todos os seus termos e condições para copiar, distribuir ou modificar o Programa ou obras nele baseadas.
6. Cada vez que você redistribuir o Programa (ou obra baseada no Programa), o receptor receberá, automaticamente, uma licença do licenciante original, para copiar, distribuir ou modificar o Programa, sujeito a estes termos e condições. Você não poderá impor quaisquer restrições adicionais ao exercício, pelos receptores, dos direitos concedidos por este instrumento. Você não tem responsabilidade de promover o cumprimento por parte de terceiros desta licença.
7. Se, como resultado de uma sentença judicial ou alegação de violação de patente, ou por qualquer outro motivo (não restrito às questões de patentes), forem impostas a você condições (tanto através de mandado judicial, contrato ou qualquer outra forma) que contradigam as condições desta Licença, você não estará desobrigado quanto às condições desta Licença. Se você não puder atuar como distribuidor de modo a satisfazer simultaneamente suas obrigações sob esta licença e quaisquer outras obrigações pertinentes, então, como conseqüência, você não poderá distribuir o Programa de nenhuma forma. Por exemplo, se uma licença sob uma patente não permite a redistribuição por parte de todos aqueles que tiverem recebido cópias, direta ou indiretamente de você, sem o pagamento de royalties, então, a única forma de cumprir tanto com esta exigência quanto com esta licença será deixar de distribuir, por completo, o Programa.

Se qualquer parte desta Cláusula for considerada inválida ou não executável, sob qualquer circunstância específica, o restante da cláusula deverá continuar a ser aplicado e a cláusula, como um todo, deverá ser aplicada em outras circunstâncias.

Esta cláusula não tem a finalidade de induzir você a infringir quaisquer patentes ou direitos de propriedade, nem de contestar a validade de quaisquer reivindicações deste tipo; a única finalidade desta cláusula é proteger a integridade do sistema de distribuição do software livre, o qual é implementado mediante práticas de licenças públicas. Muitas pessoas têm feito generosas contribuições à ampla gama de software distribuído através desse sistema, confiando na aplicação consistente deste sistema; cabe ao autor/doador decidir se deseja distribuir software através de qualquer outro sistema e um licenciado não pode impor esta escolha.

Esta cláusula visa deixar absolutamente claro o que se acredita ser uma conseqüência do restante desta Licença.

8. Se a distribuição e/ou uso do Programa for restrito em determinados países, tanto por patentes ou por interfaces protegidas por direito autoral, o titular original dos direitos autorais que

colocar o Programa sob esta Licença poderá acrescentar uma limitação geográfica de distribuição explícita excluindo esses países, de modo que a distribuição seja permitida somente nos países ou entre os países que não foram excluídos dessa forma. Nesse caso, esta Licença passa a incorporar a limitação como se esta tivesse sido escrita no corpo desta Licença.

9. A Free Software Foundation poderá de tempos em tempos publicar novas versões e/ou versões revisadas da Licença Pública Geral. Essas novas versões serão semelhantes em espírito à presente versão, mas podem diferenciar-se, porém, em detalhe, para tratar de novos problemas ou preocupações.

Cada versão recebe um número de versão distinto. Se o Programa especificar um número de versão desta Licença que se aplique a ela e a "qualquer versão posterior", você terá a opção de seguir os termos e condições tanto daquela versão como de qualquer versão posterior publicada pela Free Software Foundation. Se o Programa não especificar um número de versão desta Licença, você poderá escolher qualquer versão já publicada pela Free Software Foundation.

10. Se você desejar incorporar partes do Programa em outros programas livres cujas condições de distribuição sejam diferentes, escreva ao autor solicitando a respectiva permissão. Para software cujos direitos autorais sejam da Free Software Foundation, escreva para ela; algumas vezes, abrimos exceções para isso. Nossa decisão será guiada pelos dois objetivos de preservar a condição livre de todos os derivados de nosso software livre e de promover o compartilhamento e reutilização de software, de modo geral.

EXCLUSÃO DE GARANTIA

11. COMO O PROGRAMA É LICENCIADO SEM CUSTO, NÃO HÁ NENHUMA GARANTIA PARA O PROGRAMA, NO LIMITE PERMITIDO PELA LEI APLICÁVEL. EXCETO QUANDO DE OUTRA FORMA ESTABELECIDO POR ESCRITO, OS TITULARES DOS DIREITOS AUTORAIS E/OU OUTRAS PARTES, FORNECEM O PROGRAMA "NO ESTADO EM QUE SE ENCONTRA", SEM NENHUMA GARANTIA DE QUALQUER TIPO, TANTO EXPRESSA COMO IMPLÍCITA, INCLUINDO, DENTRE OUTRAS, AS GARANTIAS IMPLÍCITAS DE COMERCIALIZABILIDADE E ADEQUAÇÃO A UMA FINALIDADE ESPECÍFICA. O RISCO INTEGRAL QUANTO À QUALIDADE E DESEMPENHO DO PROGRAMA É ASSUMIDO POR VOCÊ. CASO O PROGRAMA CONTENHA DEFEITOS, VOCÊ ARCARÁ COM OS CUSTOS DE TODOS OS SERVIÇOS, REPAROS OU CORREÇÕES NECESSÁRIAS.
12. EM NENHUMA CIRCUNSTÂNCIA, A MENOS QUE EXIGIDO PELA LEI APLICÁVEL OU ACORDADO POR ESCRITO, QUALQUER TITULAR DE DIREITOS AUTORAIS OU QUALQUER OUTRA PARTE QUE POSSA MODIFICAR E/OU REDISTRIBUIR O PROGRAMA, CONFORME PERMITIDO ACIMA, SERÁ RESPONSÁVEL PARA COM VOCÊ POR DANOS, INCLUINDO ENTRE OUTROS, QUAISQUER DANOS GERAIS, ESPECIAIS, FORTUITOS OU EMERGENTES, ADVINDOS DO USO OU IMPOSSIBILIDADE DE USO DO PROGRAMA (INCLUINDO, ENTRE OUTROS, PERDAS DE DADOS OU DADOS SENDO GERADOS DE FORMA IMPRECISA, PERDAS SOFRIDAS POR VOCÊ OU TERCEIROS OU A IMPOSSIBILIDADE DO PROGRAMA DE OPERAR COM QUAISQUER OUTROS PROGRAMAS), MESMO QUE ESSE TITULAR, OU OUTRA PARTE, TENHA SIDO ALERTADA SOBRE A POSSIBILIDADE DE OCORRÊNCIA DESSES DANOS.

FINAL DOS TERMOS E CONDIÇÕES

Como Aplicar Estes Termos para Seus Novos Programas

Se você desenvolver um programa novo e quiser que ele seja da maior utilidade possível para o público, o melhor caminho para obter isto é fazer dele um software livre, o qual qualquer pessoa pode redistribuir e modificar sob os presentes termos.

Para fazer isto, anexe as notificações seguintes ao programa. É mais seguro anexá-las ao começo de cada arquivo-fonte, de modo a transmitir do modo mais eficiente a exclusão de garantia; e cada arquivo deve ter ao menos a linha de "direitos autorais reservados" e uma indicação de onde a notificação completa se encontra.

<uma linha para informar o nome do programa e uma breve idéia do que ele faz.>

Direitos Autorais Reservados (c) <ano> <nome do autor>

Este programa é software livre; você pode redistribuí-lo e/ou modificá-lo sob os termos da Licença Pública Geral GNU conforme publicada pela Free Software Foundation; tanto a versão 2 da Licença, como (a seu critério) qualquer versão posterior.

Este programa é distribuído na expectativa de que seja útil, porém, SEM NENHUMA GARANTIA; nem mesmo a garantia implícita de COMERCIALIZABILIDADE OU ADEQUAÇÃO A UMA FINALIDADE ESPECÍFICA. Consulte a Licença Pública Geral do GNU para mais detalhes.

Você deve ter recebido uma cópia da Licença Pública Geral do GNU junto com este programa; se não, escreva para a Free Software Foundation, Inc., no endereço 59 Temple Street, Suite 330, Boston, MA 02111-1307 USA.

Inclua também informações sobre como contatar você por correio eletrônico e por meio postal.

Se o programa for interativo, faça com que produza uma pequena notificação como esta, quando for iniciado em um modo interativo:

Versão 69 do Gnomovision, Direitos Autorais Reservados (c) ano nome do autor. O Gnomovision NÃO POSSUI QUALQUER TIPO DE GARANTIA; para detalhes, digite 'show w'. Este é um software livre e você é bem-vindo para redistribuí-lo sob certas condições; digite 'show c' para detalhes.

Os comandos hipotéticos `show w` e `show c` devem mostrar as partes apropriadas da Licença Pública Geral. Naturalmente, os comandos que você utilizar poderão ter outras denominações que não `show w` e `show c`; eles poderão até ser cliques do mouse ou itens de um menu - o que for adequado ao seu programa.

Você também pode solicitar a seu empregador (se você for um programador) ou sua instituição acadêmica, se for o caso, para assinar uma "renúncia de direitos autorais" sobre o programa, se necessário. Segue um exemplo; altere os nomes:

A Yoyodyne Ltda., neste ato, renuncia a todos eventuais direitos autorais sobre o programa `Gnomovision` (que realiza passagens em compiladores), escrito por James Hacker.

<Assinatura de Ty Coon>

1º de abril de 1989, Ty Coon, Presidente

Esta Licença Pública Geral não permite a incorporação do seu programa a programas proprietários. Se seu programa é uma biblioteca de sub-rotinas, você poderá considerar ser mais útil permitir a ligação de aplicações proprietárias à sua biblioteca. Se isso é o que você deseja fazer, utilize a Licença Pública Geral de Biblioteca do GNU, ao invés desta Licença.

CONTRATO DE LICENÇA DE USUÁRIO FINAL PARA SOFTWARE MICROSOFT

MICROSOFT[r] WINDOWS[r] 95

IMPORTANTE - LEIA COM ATENÇÃO: Este Contrato de Licença de Usuário Final ("EULA", End-User License Agreement) é um acordo legal entre V.Sa. (pessoa física ou jurídica) e o fabricante ("Fabricante") do sistema de computador ("COMPUTADOR") adquirido por V.Sa. juntamente com o (s) programa(s) de computador Microsoft indicado(s) acima ("PRODUTO" ou "SOFTWARE"). Caso o SOFTWARE não esteja acompanhado de um novo sistema de computador, V.Sa. não poderá utilizar ou copiar o SOFTWARE. O SOFTWARE inclui o programa de computador, meios físicos, materiais impressos e qualquer documentação "on-line" ou eletrônica. Ao instalar, copiar ou de qualquer outra forma utilizar o SOFTWARE, V.Sa. está concordando em vincular-se aos termos deste EULA. Caso V.Sa. não esteja de acordo com os termos deste EULA, o Fabricante e a Microsoft não licenciarão o SOFTWARE para V.Sa. Neste caso, V.Sa. não poderá utilizar ou copiar o SOFTWARE e deverá contatar imediatamente o Fabricante para obter instruções sobre como devolver o(s) produto(s) não utilizados e para receber um reembolso.

LICENÇA DO SOFTWARE

Este SOFTWARE é protegido pelas leis de copyright e tratados internacionais, bem como por outras legislações e tratados sobre propriedade intelectual. O SOFTWARE é licenciado e não vendido.

1. **CONCESSÃO DA LICENÇA.** Este EULA concede à V.Sa. os direitos a seguir:

- * **Software.** V.Sa. poderá instalar e utilizar uma cópia do SOFTWARE no COMPUTADOR.
- * **Serviços de Rede.** Caso o SOFTWARE contenha uma função que permita que o COMPUTADOR atue como um servidor de rede, qualquer número de computadores ou estações de trabalho poderá acessar ou de qualquer outra forma utilizar os serviços básicos de rede daquele servidor. Os serviços básicos de rede estão descritos mais detalhadamente nos materiais impressos que acompanham o SOFTWARE.
- * **Armazenamento/Utilização de Rede.** V.Sa. também poderá armazenar ou instalar uma cópia da parte de programa de computador do SOFTWARE no COMPUTADOR, para permitir que outros computadores utilizem o SOFTWARE em uma rede interna e o distribuam para outros computadores de uma rede interna. Todavia, V.Sa. deverá adquirir uma licença do SOFTWARE dedicada para cada computador no qual o SOFTWARE seja utilizado ou para o qual seja distribuído. É vedado compartilhar ou utilizar a licença do SOFTWARE simultaneamente em computadores diferentes.
- * **Escolha do Sistema Operacional.** O Fabricante poderá optar por fornecer à V.Sa. uma opção de sistema operacional Microsoft para o COMPUTADOR. Caso o SOFTWARE inclua tanto o Windows 95 como um "sistema operacional Microsoft alternativo" (seja (a) o Windows para Workgroups, (b) o MS-DOS e o Windows, (c) o sistema operacional MS-DOS e o Windows para Workgroups, ou (d) o sistema operacional Windows), V.Sa. está autorizado a utilizar somente uma das opções de sistema operacional fornecidas. Como parte do processo de configuração do SOFTWARE, será dada a V.Sa. uma opção única para escolher o Windows 95 ou um sistema operacional Microsoft alternativo. O sistema operacional selecionado por V.Sa. será configurado no COMPUTADOR e o sistema operacional Microsoft não selecionado será excluído automática e permanentemente do disco rígido do COMPUTADOR.

* Utilitário de Backup. Caso o Fabricante não tenha incluído uma cópia de salvaguarda do SOFTWARE no COMPUTADOR, V.Sa. poderá utilizar o utilitário de backup Microsoft, caso esteja incluído no SOFTWARE, para efetuar uma única cópia de salvaguarda do SOFTWARE. V.Sa. somente poderá utilizar a cópia de salvaguarda para fins de arquivo. Após a elaboração de uma única cópia de salvaguarda, o utilitário de backup será desativado permanentemente.

2. DESCRIÇÃO DE OUTROS DIREITOS E LIMITAÇÕES.

* Engenharia Reversa, Descompilação e Desmontagem. É vedada a engenharia reversa, descompilação ou desmontagem do SOFTWARE, exceto e somente se esta atividade for expressamente permitida pela legislação aplicável, inobstante tal limitação.

* Separação de Componentes. O SOFTWARE é licenciado como um produto individual. Seus componentes não poderão ser separados para utilização em mais de um computador.

* COMPUTADOR individual. O SOFTWARE é licenciado com o COMPUTADOR como um produto individual integrado ao COMPUTADOR. O SOFTWARE somente poderá ser utilizado com o COMPUTADOR.

* Aluguel. É vedado o aluguel ou arrendamento do SOFTWARE.

* Transferência do Software. V.Sa. somente poderá transferir permanentemente todos os seus direitos sob este EULA, como parte da venda ou transferência do COMPUTADOR, desde que V.Sa. não mantenha cópias, transfira todo o SOFTWARE (incluindo todos os componentes, meios físicos e materiais impressos, quaisquer atualizações, este EULA, e caso aplicável, o(s) Certificado(s) de Autenticidade), e o adquirente concorde com os termos deste EULA. Caso o SOFTWARE seja uma atualização, qualquer transferência deverá incluir todas as versões anteriores do SOFTWARE.

* Rescisão. Sem prejuízo de quaisquer outros direitos, a Microsoft poderá rescindir este EULA caso V.Sa. não cumpra com seus termos e condições. Neste caso, V.Sa. deverá destruir todas as cópias do SOFTWARE e de seus componentes.

3. ATUALIZAÇÕES ("UPGRADES"). Caso o SOFTWARE seja uma atualização de outro produto, da Microsoft ou de outro fornecedor, V.Sa. somente poderá utilizar ou transferir o SOFTWARE em conjunto com o produto atualizado, a menos que V.Sa. destrua o produto atualizado. Caso o SOFTWARE seja uma atualização de um produto Microsoft, V.Sa. somente poderá utilizar o produto atualizado de acordo com este EULA. Caso o SOFTWARE seja uma atualização de um componente de um pacote de programas de computador licenciado por V.Sa. como um produto individual, o SOFTWARE somente poderá ser utilizado e transferido como parte de tal produto individual e não poderá ser separado para utilização em mais de um computador.

4. COPYRIGHT. Todos os títulos e direitos autorais relativos ao SOFTWARE (incluindo quaisquer imagens, fotografias, animações, vídeos, áudios, músicas, textos e "applets" incorporados ao SOFTWARE), os materiais impressos que o acompanham e quaisquer cópias do SOFTWARE são de propriedade da Microsoft ou seus fornecedores. O SOFTWARE é protegido pelas leis de copyright e tratados internacionais. É vedada a reprodução dos materiais impressos que acompanham o SOFTWARE.

5. SUPORTE DUAL. V.Sa. poderá receber o SOFTWARE em mais de um meio físico.

Independentemente do tipo ou tamanho dos meios físicos que V.Sa. receba, somente poderá utilizar o meio físico adequado para o seu computador individual. V.Sa. não poderá utilizar ou instalar o outro meio físico em outro computador, bem como emprestar, alugar, arrendar ou de qualquer outra forma transferir o outro meio físico para outro usuário, exceto como parte de uma transferência permanente do SOFTWARE (conforme indicado acima).

6. SUPORTE AO PRODUTO. O suporte ao produto para o SOFTWARE não é prestado pela Microsoft Corporation ou suas subsidiárias. Para obter informações quanto ao suporte ao produto, favor consultar

o número de suporte do Fabricante indicado na documentação do COMPUTADOR. Caso V.Sa. tenha quaisquer dúvidas relativas a este EULA ou deseje contatar o Fabricante por qualquer outra razão, utilize o endereço indicado na documentação do COMPUTADOR.

NO TOCANTE ÀS GARANTIAS LIMITADAS E DISPOSIÇÕES ESPECIAIS RELATIVAS AO PAÍS DE V.SA., FAVOR REPORTAR-SE À BROCHURA DE GARANTIA LIMITADA CONTIDA NESTE PACOTE OU NOS MATERIAIS IMPRESSOS QUE ACOMPANHAM O SOFTWARE.

ADENDO AO CONTRATO DE LICENÇA PARA O SOFTWARE MICROSOFT[r] WINDOWS[r]
95

Este Adendo descreve termos adicionais aplicáveis ao uso do programa de computador para conferência "Microsoft Internet Explorer and NetMeeting[tm]" ("Software"), por V.Sa.. V.Sa. somente poderá usar o Software como um componente do Windows 95 e sua utilização está sujeita aos termos deste Adendo e do Contrato de Licença de Usuário Final ("EULA") fornecido com o Windows 95. Termos não definidos neste Adendo terão o mesmo significado estabelecido no EULA. Ao usar ou instalar este Software, V.Sa. estará concordando em vincular-se aos termos do EULA e deste Adendo. Caso não concorde com estes termos, V.Sa. não estará autorizada a usar este Software.

1. Os seguintes termos adicionais são aplicáveis ao Microsoft Internet Explorer:

* OBSERVAÇÃO SOBRE O SUPORTE A JAVA. O SOFTWARE PODE CONTER SUPORTE PARA PROGRAMAS ESCRITOS EM JAVA. A TECNOLOGIA JAVA NÃO É TOLERANTE A ERROS ("FAULT TOLERANT") E NÃO FOI PROJETADA, FABRICADA OU PLANEJADA PARA USO OU REVENDA COMO EQUIPAMENTO DE CONTROLE ON-LINE EM AMBIENTES DE RISCO QUE NECESSITAM DE UM DESEMPENHO LIVRE DE ERROS, COMO NA OPERAÇÃO DE INSTALAÇÕES NUCLEARES, SISTEMAS DE COMUNICAÇÃO E NAVEGAÇÃO AÉREAS, CONTROLE DE TRÁFEGO AÉREO, EQUIPAMENTOS DE MANUTENÇÃO DE VIDAS E SISTEMAS DE ARMAMENTO, NOS QUAIS A FALHA DA TECNOLOGIA JAVA PODE, DIRETAMENTE, CAUSAR MORTE, LESÕES CORPORAIS, DANOS FÍSICOS OU AMBIENTAIS GRAVES.

2. Os seguintes termos adicionais são aplicáveis ao NetMeeting[tm] Conferencing Software:

* COMPARTILHAMENTO DE APLICATIVOS. O Software possui tecnologia que permite o compartilhamento de aplicativos entre dois ou mais computadores, mesmo que um aplicativo esteja instalado apenas em um dos computadores. V.Sa. poderá usar esta tecnologia em todos os produtos Microsoft para conferências com vários participantes. Para aplicativos que não sejam da Microsoft, V.Sa. deverá consultar os respectivos contratos de licença que os acompanham ou entrar em contato com o licenciante dos mesmos a fim de determinar se o compartilhamento de tais aplicativos é permitido.

* V.Sa. deverá cumprir todas as leis aplicáveis no que diz respeito ao uso do Software.

É vedado o uso do Software caso V.Sa. não possua uma licença válida para usar o Windows 95. Qualquer outro uso do Software é proibido.

c. Open Source Definition

Introduction

Open source doesn't just mean access to the source code. The distribution terms of open-source software must comply with the following criteria:

1. Free Redistribution

The license shall not restrict any party from selling or giving away the software as a component of an aggregate software distribution containing programs from several different sources. The license shall not require a royalty or other fee for such sale.

Rationale: By constraining the license to require free redistribution, we eliminate the temptation to throw away many long-term gains in order to make a few short-term sales dollars. If we didn't do this, there would be lots of pressure for cooperators to defect.

2. Source Code

The program must include source code, and must allow distribution in source code as well as compiled form. Where some form of a product is not distributed with source code, there must be a well-publicized means of obtaining the source code for no more than a reasonable reproduction cost—preferably, downloading via the Internet without charge. The source code must be the preferred form in which a programmer would modify the program. Deliberately obfuscated source code is not allowed. Intermediate forms such as the output of a preprocessor or translator are not allowed.

Rationale: We require access to un-obfuscated source code because you can't evolve programs without modifying them. Since our purpose is to make evolution easy, we require that modification be made easy.

3. Derived Works

The license must allow modifications and derived works, and must allow them to be distributed under the same terms as the license of the original software.

Rationale: The mere ability to read source isn't enough to support independent peer review and rapid evolutionary selection. For rapid evolution to happen, people need to be able to experiment with and redistribute modifications.

4. Integrity of The Author's Source Code

The license may restrict source-code from being distributed in modified form only if the license allows the distribution of "patch files" with the source code for the purpose of modifying the program at build time. The license must explicitly permit distribution of software built from modified source code. The license may require derived works to carry a different name or version number from the original software.

Rationale: Encouraging lots of improvement is a good thing, but users have a right to know who is responsible for the software they are using. Authors and maintainers have reciprocal right to know what they're being asked to support and protect their reputations.

Accordingly, an open-source license must guarantee that source be readily available, but may require that it be distributed as pristine base sources plus patches. In this way, "unofficial" changes can be made available but readily distinguished from the base source.

5. No Discrimination Against Persons or Groups

The license must not discriminate against any person or group of persons.

Rationale: In order to get the maximum benefit from the process, the maximum diversity of persons and groups should be equally eligible to contribute to open sources. Therefore we forbid any open-source license from locking anybody out of the process.

Some countries, including the United States, have export restrictions for certain types of software. An OSD-conformant license may warn licensees of applicable restrictions and remind them that they are obliged to obey the law; however, it may not incorporate such restrictions itself.

6. No Discrimination Against Fields of Endeavor

The license must not restrict anyone from making use of the program in a specific field of endeavor. For example, it may not restrict the program from being used in a business, or from being used for genetic research.

Rationale: The major intention of this clause is to prohibit license traps that prevent open source from being used commercially. We want commercial users to join our community, not feel excluded from it.

7. Distribution of License

The rights attached to the program must apply to all to whom the program is redistributed without the need for execution of an additional license by those parties.

Rationale: This clause is intended to forbid closing up software by indirect means such as requiring a non-disclosure agreement.

8. License Must Not Be Specific to a Product

The rights attached to the program must not depend on the program's being part of a particular software distribution. If the program is extracted from that distribution and used or distributed within the terms of the program's license, all parties to whom the program is redistributed should have the same rights as those that are granted in conjunction with the original software distribution.

Rationale: This clause forecloses yet another class of license traps.

9. License Must Not Restrict Other Software

The license must not place restrictions on other software that is distributed along with the licensed software. For example, the license must not insist that all other programs distributed on the same medium must be open-source software.

Rationale: Distributors of open-source software have the right to make their own choices about their own software.

Yes, the GPL is conformant with this requirement. Software linked with GPLed libraries only inherits the GPL if it forms a single work, not any software with which they are merely distributed.

10. License Must Be Technology-Neutral

No provision of the license may be predicated on any individual technology or style of interface.

Rationale: This provision is aimed specifically at licenses which require an explicit gesture of assent in order to establish a contract between licensor and licensee. Provisions mandating so-called "click-wrap" may conflict with important methods of software distribution such as FTP download, CD-ROM anthologies, and web mirroring; such provisions may also hinder code re-use. Conformant licenses must allow for the possibility that (a) redistribution of the software will take place over non-Web

channels that do not support click-wrapping of the download, and that (b) the covered code (or re-used portions of covered code) may run in a non-GUI environment that cannot support popup dialogues.

d. Projeto de Lei de Walter Pinheiro

PROJETO DE LEI N.º 2.269/99 DE 1999

(de Walter Pinheiro)

Dispõe sobre a utilização de programas abertos pelos entes de direito público e de direito privado sob controle acionário da administração pública

O Congresso Nacional decreta:

Artigo 1º- A administração pública, em todos os níveis, os Poderes da República, as empresas estatais e de economia mista, as empresas públicas, e todos os demais organismos públicos ou privados sob controle da sociedade brasileira, ficam obrigadas a utilizarem preferencialmente, em seus sistemas e equipamento de informática, programas abertos, livres de restrição proprietária quanto a sua cessão alteração e distribuição.

Artigo 2º - Entende-se por programa aberto aquele cuja licença de propriedade industrial ou intelectual não restrinja sob nenhum aspecto a sua cessão, distribuição, utilização ou alteração de suas características originais.

Artigo 3º - O programa aberto deve assegurar ao usuário acesso irrestrito ao seu código fonte, sem qualquer custo, com vista a modificar o programa, integralmente, se necessário, para o seu aperfeiçoamento.

Parágrafo Único. O código fonte deve ser o recurso preferencial utilizado pelo programador para modificar o programa, não sendo permitido ofuscar a sua acessibilidade, nem tampouco introduzir qualquer forma intermediária como saída de um pré-processador ou tradutor.

Artigo 4º - A licença de utilização dos programas abertos deve permitir modificações e trabalhos derivados e sua livre distribuição sob os mesmos termos da licença do programa original.

§ 1º - A licença somente poderá restringir a distribuição do código fonte em forma modificada caso permita a distribuição de programas alterados conjuntamente com o código fonte original, objetivando a alteração do programa durante o processo de compilação.

§ 2º - Deve permitir também explicitamente a distribuição de programa compilado a partir do código fonte modificado, podendo para tanto exigir que os programas derivados tenham diferentes nomes ou números de versão, que os diferenciem do original.

Artigo 5º - Não poderá haver cláusula na licença que implique em qualquer forma de discriminação a pessoas ou grupos.

Artigo 6º - Nenhuma licença poderá ser específica para determinado produto, possibilitando que os programas extraídos da distribuição original tenham a mesma garantia de livre alteração, distribuição ou utilização, que o programa original.

Artigo 7º - As licenças de programas abertos ou restritos, não restringirão outros programas distribuídos conjuntamente.

Artigo 8º - Os certames licitatórios que objetivem transacionar programas de computador com os entes especificados no artigo 1º desta lei, deverão obrigatoriamente ser regidos pelos princípios estabelecidos nesta legislação.

Artigo 9º - Apenas será permitida a utilização pelos entes do artigo 1º, de programas de computador cujas licenças não estejam em acordo com esta lei, na ausência de programas abertos que contemplem a contento as soluções objeto da licitação pública.

JUSTIFICAÇÃO

Há mais de quinze anos discute-se em todo o mundo a livre manipulação dos programas de computador ou "open/ free software". Em 1984 era impossível usar um computador moderno sem a instalação de um sistema operacional proprietário, fornecido mediante licenças restritivas de amplo espectro. Ninguém tinha permissão para compartilhar programas (software) livremente com outros usuários de computador, e dificilmente alguém poderia mudar os programas para satisfazer as suas necessidades operacionais específicas.

O projeto GNU, que data do início do Movimento do Software livre, foi fundado para mudar isso. Seu primeiro objetivo foi desenvolver um sistema operacional portátil compatível com o UNIX que seria 100% livre para alteração e distribuição, proporcionando aos usuários que contribuíssem com o seu desenvolvimento e alteração de qualquer parte de sua constituição original.

Tecnicamente GNU é como UNIX, mas difere do UNIX pela liberdade que se proporciona aos seus usuários. Para a confecção deste programa aberto, foram necessários muitos anos de trabalho, por centenas de programadores, para desenvolver este sistema operacional. Em 1991, o último componente mais importante de um sistema similar ao UNIX foi desenvolvido: o LINUX.

Hoje a combinação do GNU e do Linux é usada por milhões de pessoas, de forma livre, em todo o mundo. Este programa é apenas um exemplo de como a liberdade na alteração, distribuição e utilização de programas de computador poder transformar ainda mais rapidamente, e de maneira mais democrática, o perfil do desenvolvimento social e tecnológico no mundo.

O Estado, como ente fomentador do desenvolvimento tecnológico e da democratização do acesso a novas tecnologias para a sociedade, não pode ser furtar a sua responsabilidade de priorizar a utilização de programas abertos ou os "free software/open source". E se as pequenas, médias e grandes empresas

multinacionais já estão adotando programas abertos, evitando assim o pagamento de centenas de milhões de dólares em licenciamento de programas, porque deveria o Estado, com uma infinidade de causas sociais carentes de recursos, continuar comprando, e caro, os programas de mercado.

Sala das Sessões em 15 de Dezembro de 1999

Deputado federal Walter Pinheiro

e. Proyecto de Lei de Edgar V. Nuñez

PROYECTO DE LEY SOFTWARE LIBRE

CONSIDERANDO:

Que, el artículo 14° de la Constitución establece que es deber del Estado promover el desarrollo científico y tecnológico del país.

Que, el artículo 44° de la Constitución señala que es deber primordial del Estado promover el bienestar general que se fundamenta en la justicia y en el desarrollo integral y equilibrado de la Nación.

Que, una de las claves para el progreso del país consiste en el desarrollo de la tecnología referente a los programas informáticos o “software” que se emplean en las computadoras u ordenadores.

Que, es necesario establecer la política del Estado con relación al desarrollo de la tecnología de los programas informáticos para uso propio.

Por los considerandos antes expuestos y en cumplimiento de lo dispuesto por el artículo 75° del Reglamento del Congreso de la República paso a desarrollar la presente propuesta:

EXPOSICIÓN DE MOTIVOS

En la época de globalización que vivimos uno de los factores claves para el desarrollo de los países es el dominio de la ciencia y de la tecnología.

Esto es especialmente cierto con relación a la informática que permite mediante programas o “software” procesar información necesaria para las distintas áreas del conocimiento. En la actualidad no existe prácticamente ninguna actividad que no este relacionada de una u otra manera con la informática y obviamente con los programas informáticos.

Esto crea una situación de dependencia en la que nos convertimos en consumidores de tecnología elaborada en otros países. Es más, esta tecnología se encuentra en constante desarrollo, y como consecuencia de ello lo que hoy es novedoso en poco tiempo se convierte en obsoleto; obligándonos así a tener que adquirir la nueva tecnología sino queremos quedar rezagados.

Esta carrera por no quedar atrasados resulta en elevados costos por pago del derecho de uso de esta tecnología informática. Esto es particularmente cierto en la administración pública, como en la actividad privada, en las que por concepto de licencias de “software” se debe abonar elevadas sumas de dinero cada año.

Esta situación ocasiona que muchas veces tanto el Estado como los particulares no tengan la capacidad para pagar el valor de las licencias. Prueba de ello la encontramos en los resultados que arrojan las intervenciones realizadas por INDECOPI que muestran como inclusive en las dependencias públicas se hace uso del denominado “software pirata”.

Este panorama hace necesario que el Estado se preocupe de dar alternativas de solución que permitan romper el círculo vicioso de dependencia tecnológica en la que nos encontramos, así como a los problemas que de ello se derivan.

Precisamente, la alternativa inteligente y viable a los programas informáticos o “software” comercial viene a ser el denominado “software libre” denominado así porque se basa en el libre empleo de los programas de computación.

El uso de software libre se basa en las siguientes premisas fundamentales, propuestas por la Fundación para el Software Libre:

- Libertad para ejecutar el programa con cualquier propósito.
- Libertad para modificar el programa con el fin de adaptarlo a sus necesidades.
- Libertad para redistribuir copias, tanto en forma gratuita como a través del pago de un canon.
- Libertad para distribuir versiones modificadas del programa, de tal manera que la comunidad pueda beneficiarse con sus mejoras.

Este esfuerzo en pro del uso del software libre ha tenido como resultado que varias de las principales empresas de la industria se hayan adherido a estos principios.

Así por ejemplo, el sistema operativo LINUX- creado por el estudiante finlandés Linus Torvalds- es usado por millones de personas en el mundo entero, en forma libre, permitiendo un acceso masivo a quienes no pueden costear programas comerciales.

Cabe reiterar que debido a las ventajas que ofrece que los mismos usuarios creen soluciones propias, estableciendo sus propios patrones de seguridad hace que sean cada vez más el número de empresas y organismos que adopten el sistema de software libre, como por ejemplo el Ministerio de Defensa de Francia y la República Popular China.

Actualmente existen soluciones de software libre para instalar y usar en las siguientes funciones, entre otras:

- Desktop en estación de trabajo
- Infraestructura de red para máquinas con cualquier sistema operativo: Linux, Windows, Macintosh, Unix, etc.
- Servidor de archivos.
- Servidor de impresoras.
- Servidor de correo electrónico, correo electrónico y comunicaciones en general.
- Servidor de World Wide Web – WWW.
- Servidor de acceso a internet y navegación.
- Gestión de bases de datos.
- Lenguajes de Programación – desarrollo de software.
- Gestión de oficina - Ofimática (procesador de textos, hoja de cálculo, etc.)
- Conversación en línea, voz, datos e imagen – Video conferencia.

Otra ventaja es que la adopción de programas abiertos prolonga la vida útil de las computadoras toda vez que los requerimientos de nuevo hardware son menores, lo cual redundará en beneficio de la economía de los usuarios.

ANÁLISIS COSTO BENEFICIO

La adopción por parte del Estado del uso del software libre va a redundar en beneficio del proceso de reforma y modernización en que se haya inmerso debido a que este tipo de procesos requiere de la sistematización informática, de la pronta comunicación entre los diferentes componentes de la administración pública, de la uniformización de los sistemas operativos y de procesamientos de datos.

Todo este urgente y necesario proceso de modernización implica el uso intensivo de las computadoras y redes de comunicación – sea Intranet y/o internet. Lo cual significa que de hacer uso de los programas informáticos comerciales se tendría que desembolsar ingentes sumas de dinero por concepto de licencias, actualización de programas y renovación de equipos.

Vale decir, el Estado ahorrará una considerable suma de dinero que puede dedicar a los programas de eliminación de la pobreza. Igualmente, al ser libre el software el Estado a través de sus técnicos desarrollará las aplicaciones que cada uno de los componentes de la administración necesita en función de sus requerimientos y necesidades, a la vez de establecer sus propios estándares de seguridad.

EFFECTOS SOBRE LA LEGISLACIÓN

El presente proyecto no afecta ninguna Ley toda vez que no se ha legislado aún sobre la materia.

FORMULA LEGAL

DEFINICIONES

ARTÍCULO 1º.- A los efectos de la presente Ley entiéndase por:

- a. Programa o “software”, a cualquier secuencia de instrucciones usada por un dispositivo de procesamiento digital de datos para llevar a cabo una tarea específica o resolver un problema determinado.
- b. Ejecución o empleo de un programa, al acto de utilizarlo sobre cualquier dispositivo de procesamiento digital de datos para realizar una función.
- c. Usuario, a aquella persona física o jurídica que emplea el software.
- d. Código fuente o de origen, o programa fuente o de origen, al conjunto completo de instrucciones y archivos digitales originales creados o modificados por quien los programara, más todos los archivos digitales de soporte, como las tablas de datos, imágenes, especificaciones, documentación, y todo otro elemento que sea necesario para producir el programa ejecutable a partir de ellos. Como excepción podrán excluirse de este conjunto aquellas herramientas y programas que sean habitualmente distribuidos como software libre por otros medios como, entre

otros, compiladores, sistemas operativos y librerías.

- e. Programa o software libre, a aquel cuyo empleo garantice al usuario, sin costo adicional, las siguientes facultades:
 - e.1 Ejecución irrestricta del programa para cualquier propósito.
 - e.2 Acceso irrestricto al código fuente o de origen respectivo.
 - e.3 Inspección exhaustiva de los mecanismos de funcionamiento del programa.
 - e.4 Uso de los mecanismos internos y de cualquier porción arbitraria del programa para adaptarlo a las necesidades del usuario.
 - e.5 Confección y distribución pública de copias del programa.
 - e.6 Modificación del programa y distribución libre, tanto de las alteraciones como del nuevo programa resultante bajo las mismas condiciones del programa original.

Además, el costo de obtención de una copia del código fuente del programa por parte del usuario no podrá ser significativamente mayor al costo habitual de mercado en concepto de materiales, mano de obra y logística necesarias para la confección de dicha copia.

- f. Programa o software “no libre” o “propietario” o “comercial”, a aquel que no reúna todos los requisitos expresados en el artículo 1º inciso 'e' de la presente Ley.
- g. Formato abierto a cualquier modo de codificación de información digital que satisfaga las siguientes condiciones tales que:
 - g.1 Su documentación técnica completa esté disponible públicamente.
 - g.2 El código fuente de al menos una implementación de referencia completa esté disponible públicamente.
 - g.3 No existan restricciones para la confección de programas que almacenen, transmitan, reciban, o accedan a datos codificados de esta manera.

ÁMBITO DE APLICACIÓN

ARTÍCULO 2º.- Los Poderes Ejecutivo, Legislativo y Judicial, los Organismos Descentralizados y las Empresas donde el Estado posea mayoría accionaria, emplearán en sus sistemas y equipamientos de informática exclusivamente programas o software libres.

ARTÍCULO 3º.- La autoridad para la aplicación de esta Ley será la Presidencia del Consejo de Ministros.

EXCEPCIONES

ARTÍCULO 4º.- En caso de no existir una solución que utilice software libre y permita satisfacer una necesidad determinada, los organismos estatales mencionados en el artículo 2º podrán adoptar las siguientes alternativas prelativamente:

- a. En caso de inexistencia o indisponibilidad de software no libre que permita dar solución a la necesidad planteada, y que como consecuencia de ello se determinara la necesidad de su desarrollo, la solución técnica resultante deberá ser, en todos los casos, software libre, en los términos definidos en el artículo primero de esta Ley.
- b. Si mediaran exigencias de tiempo verificables para la solución del problema técnico, y se encontraran disponibles en el mercado programas o software no libre o propietario, el organismo que los demande podrá gestionar ante la autoridad de aplicación un permiso de excepción para la utilización de software no libre. La elección del producto deberá ser realizada de acuerdo al siguiente orden de preferencia:
 - b.1. Se seleccionará en primer término a los programas que cumplan con todos los criterios

enunciados en el artículo 1º inciso 'e' de la presente Ley, excepto por la facultad de distribución del programa modificado. En este caso, el permiso de excepción podrá ser definitivo.

b.2. Si no se pudiera disponer de programas de la categoría precedente, se deberán escoger aquellos para los que exista un proyecto de desarrollo avanzado, de tipo libre. En este caso, el permiso de excepción será transitorio y caducará automáticamente cuando el software libre pase a estar disponible con la funcionalidad que sea necesaria.

b.3. Si no se encontraren productos de estas condiciones, se podrá optar por programas no libres, pero el permiso de excepción emanado de la autoridad de aplicación caducará automáticamente a los dos años de emitido, debiendo ser renovado previa constatación que no exista disponible en el mercado una solución de software libre satisfactoria.

La autoridad de aplicación sólo otorgará un permiso de excepción de si el organismo estatal solicitante garantice el almacenamiento de los datos en formatos abiertos.

ARTÍCULO 5º.- Las Universidades Nacionales y toda otra entidad educativa dependiente del Estado podrán, además, gestionar un permiso de empleo de software no libre para su uso en investigación, siempre que el objeto de investigación esté directamente asociado al uso del programa en cuestión.

TRANSPARENCIA EN LAS EXCEPCIONES

ARTÍCULO 6º.- Las excepciones emanadas de la autoridad de aplicación deberán ser fundamentadas y publicadas en los medios de comunicación que señale el reglamento.

La resolución que autoriza la excepción deberá enumerar los requisitos funcionales concretos que el programa debe satisfacer.

ARTÍCULO 7º.- Si cualquiera de los organismos comprendidos en el artículo segundo fuera autorizado en forma excepcional para adquirir o utilizar programas o software no libre para almacenar o procesar datos cuya reserva sea necesario preservar, fueren confidenciales, críticos o vitales para el desempeño del Estado, la autoridad de aplicación deberá publicar en los medios de comunicación social que determine el reglamento, en forma adicional, un informe donde se expliquen los riesgos asociados con el uso de software de dichas características para esa aplicación en particular.

Quedan exceptuados de esta obligación los permisos de excepción otorgados a los organismos del Estado vinculados con la seguridad y defensa nacionales; así como aquellos que por la naturaleza de sus funciones la autoridad de aplicación estime necesario guardar reserva.

RESPONSABILIDADES

ARTÍCULO 8º.- La máxima autoridad administrativa, junto con la máxima autoridad técnica informática de cada organismo del Estado comprendido en los alcances del artículo segundo de la presente Ley asumen la responsabilidad por el cumplimiento de esta Ley.

PLAZOS DE TRANSICIÓN

ARTÍCULO 9º.- El Poder Ejecutivo reglamentará en un plazo de ciento ochenta días las condiciones, tiempos y formas en que se efectuará la transición de la situación actual a una que satisfaga las condiciones de la presente Ley y orientará, en tal sentido, las licitaciones y contrataciones futuras de programas o software realizadas a cualquier título.

Lima, 11 de diciembre de 2001.

Firmado

Edgar Villanueva Nuñez

Congresista de la República