

**UNIVERSIDADE ESTADUAL DE CAMPINAS - UNICAMP
INSTITUTO DE FILOSOFIA E CIENCIAS HUMANAS - IFCH
PROGRAMA DE PÓS-GRADUAÇÃO EM ANTROPOLOGIA SOCIAL**

Traidores do movimento: política, cultura, ideologia e trabalho no Software Livre

Rafael de Almeida Evangelista

Orientadora: Prof. Dra. Bela Feldman-Bianco

Campinas,
fevereiro de 2010

**FICHA CATALOGRÁFICA ELABORADA PELA
BIBLIOTECA DO IFCH – UNICAMP
Bibliotecária: Maria Silvia Holloway – CRB 2289**

Ev14t **Evangelista, Rafael de Almeida**
Traidores do movimento: política, cultura, ideologia e trabalho
no software livre / Rafael de Almeida Evangelista.
-- Campinas, SP : [s. n.], 2010.

Orientador: Bela Feldman-Bianco.
Tese (doutorado) - Universidade Estadual de Campinas,
Instituto de Filosofia e Ciências Humanas.

1. Software livre. 2. Código aberto. 3. Tecnologia da
informação. 4. Tecnologia – Aspectos antropológicos.
5. Movimentos sociais. 6. Tecnologia e Estado I. Feldman-
Bianco, Bela. II. Universidade Estadual de Campinas. Instituto
de Filosofia e Ciências Humanas. III. Título.

Título em inglês: Betrayers of the movement: politics, culture, ideology and labor in free software

Palavras chaves em inglês (keywords) :

Free software
Open source
Information technology
Technology – Anthropological aspects
Social movements
State technology

Área de Concentração: Cultura e poder

Titulação: Doutor em Antropologia Social

Banca examinadora: Bela Feldman-Bianco, Tom Dwyer, Carlos Vogt, Gustavo
Lins Ribeiro, Osvaldo Lopes-Ruiz..

Data da defesa: 25-02-2010

Programa de Pós-Graduação: Antropologia Social

RAFAEL DE ALMEIDA EVANGELISTA

**“TRAIDORES DO MOVIMENTO: POLÍTICA, CULTURA, IDEOLOGIA E TRABALHO
NO SOFTWARE LIVRE”**

Tese apresentada ao Departamento de Antropologia Social do Instituto de Filosofia e Ciências Humanas da Universidade Estadual de Campinas para obtenção do grau de Doutor em Antropologia Social sob orientação da Profa. Dra. Bela Feldamn.

Este exemplar corresponde à redação final da Tese defendida e aprovada pela Comissão Julgadora no dia 25/02/2010.

Comissão Julgadora:

Titulares:

Profa. Dra. Bela Feldman – (DA-IFCH) - (Presidente)

Prof. Dr. Carlos Vogt (IEL-UNICAMP)

Prof. Dr. Tom Patrick Dwyer (DS-IFCH-UNICAMP)

Prof. Dr. Gustavo Lins Ribeiro (UnB)

Prof. Dr. Osvaldo Javier López Ruíz (INCIHUSA)

Suplentes

Prof. Dr. Mauro William Barbosa de Almeida (DA-IFCH-UNICAMP)

Prof. Dr. Omar Ribeiro Thomaz (DA-IFCH-UNICAMP)

Prof. Dr. Yuriy Castelfranchi (UFMG)

**Campinas
Fevereiro/2010**

201002105

Resumo

Esta tese procura investigar o movimento software livre entendendo-o como um movimento social amplo, formado não apenas por militantes dedicados à escrita de software, mas por um complexo conjunto de entusiasta e promotores, dando especial ênfase a questões que envolvem cultura, poder, trabalho e ideologia. Afirma-se uma divisão política fundamental de âmbito internacional entre os grupos free e open e procura-se entender sua resignificação no contexto brasileiro. O método para a realização dessa análise é um relato etnográfico da nona edição do Fórum Internacional de Software Livre, maior evento mundial a envolver de forma ampla os militantes do movimento brasileiro, enquanto drama social. A tese destaca elementos contidos na etnografia, mas sintetiza um percurso de pesquisa de seis anos em eventos, listas de discussões, entrevistas e encontros do movimento brasileiro. Examina-se detidamente as relações entre a ideologia do software livre e outras questões que emergem ao final do século XX, como o neoliberalismo e seus movimentos de resistência e contestação. Procura-se observar como essas relações articulam-se com alguns personagens e conceitos imaginados pelo movimento, como liberdade, o hacker – e a cultura hacker -, o nerd e o geek.

Palavras-chave: software livre, código aberto, FOSS, antropologia da tecnologia, cultura e poder, tecnologia da informação, movimentos sociais

Abstract

This thesis aims to investigate the free software movement seen as a broad social movement, constituted not only by programmers and developers but also by a complex group of promoters and enthusiasts. This work gives special emphasis to topics involving culture, power, labor and ideology, stating that there is a fundamental political division between the “free” and “open” groups in the international free software movement scenario. It also tries to understand how this process re-signifies such categories in regard to the Brazilian context. The method of analysis is an ethnography report of the ninth edition of the Fórum Internacional de Software Livre, the world's biggest event in the area, and the one which brings together the militants of the Brazilian movement in a social drama. The thesis highlights elements found on the ethnography and puts together data gathered in more than six years of field work, as well as information collected in other events, mailing lists, interviews and meetings of the Brazilian social movement. The relations between the free software ideology and some of the issues that emerged in the end of the 20th century, such as Neoliberalism and the acts of resistance to it, are strongly analyzed. The aim is to observe how these relations articulate themselves with some of the characters and ideas imagined by the movement, such as “freedom”, the “hacker” – and their culture -, the “nerd” and the “geek”.

Keywords: free software, FOSS, anthropology of technology, culture and power, information technology, social movements, open source

A presente obra encontra-se licenciada sob a licença Creative Commons Atribuição-Compartilhamento pela mesma licença 3.0 Brasil. Para visualizar uma cópia da licença, visite <http://creativecommons.org/licenses/by-sa/3.0/br/> ou mande uma carta para: Creative Commons, 171 Second Street, Suite 300, San Francisco, California, 94105, USA.

Agradecimentos

Em primeiro lugar, gostaria de agradecer à professora Bela Feldman-Bianco, não somente por ter aceito orientar um trabalho sobre um tema por vezes técnico, mas principalmente por sempre manter os horizontes da pesquisa abertos e ter incentivado as conexões entre política, economia, poder e cultura feitas neste trabalho. Suas dicas e conselhos de pesquisa permitiram que muitas pontas que pareciam desconexas em minha cabeça pudessem ser amarradas neste trabalho. Além disso, sua empolgação e vitalidade para o novo e para a pesquisa de campo são sempre estimulantes.

Agradeço também aos professores membros da banca final e aos professores Omar Thomaz e Mauro Almeida, que foram compreensivos com minhas urgências no exame de qualificação e contribuíram significativamente no aprimoramento deste trabalho. Agradeço ainda ao prof. Carlos Vogt por ter aceitado de súbito estar na banca final.

Devo agradecer também ao Laboratório de Estudos Avançados em Jornalismo (Labjor) pela estrutura material e apoio durante esta pesquisa. Foi em meu trabalho no Labjor que tive o contato inicial com o tema software livre e foram nas máquinas do laboratório que comecei meus experimentos com esses sistemas. Agradeço à equipe de apoio técnico-administrativo e aos colegas de redação, cujas cabeças azucrinei com minha obsessão pelas matérias sobre informática.

Agradeço a Alexandre Zarias e a Juliana Schober Gonçalves Lima pelo apoio e demonstração de confiança no momento de entrada no doutorado. Juliana ainda teve a paciência de me ouvir despejar,

por quase três horas, todo o meu caderno de campo em seu ouvido.

A Simone Pallone, Marta Kanashiro, Alessandro Piolli e Susana Dias sou grato pelas revisões de última hora neste texto e pelos papos a qualquer momento, de onde saíram algumas ideias contidas aqui.

A Tiago C. Soares, amigo de Fóruns Mundiais, Fisls, Palestra Itália, e parceiro de inúmeros projetos, agradeço pela constante troca de ideias e reflexões sobre muitos dos temas aqui discutidos.

Agradeço ainda aos inúmeros amigos que fiz nestes intensos anos de software livre. Cito alguns, correndo o risco de esquecer-me de outros igualmente queridos: Stefano Barale, banto, Juan Gentili, Giovanni Bonenti, Marco Ciurcina, Sérgio Amadeu, Carlinhos Cecconi, Luiz Fuzaro, Wilken Sanchez, Fabrício Solagna, Mario Teza, Marcelo Branco, Corinto Meffe, Cesar Brod, Aurélio Heckert, Antonio Terceiro, Wilson Sobrinho, Andre Deak, Oona Castro, Felipe Machado, Daniel Merli, Claudio Machado, Aaron Shaw, Rubens Queiroz e Vilson Gartner.

Dedico este trabalho aos meus pais, Zão e Beise, e ao meu irmão, Juca. Lutando sempre.

Sumário

Agradecimentos.....	vii
Apresentação.....	1
Cap. 1. O software livre como movimento e uma breve apresentação.....	10
O neoliberalismo enquanto modo de pensar.....	16
Da arte ao software.....	18
Do software ao mercado.....	20
Cap. 2. Free, open, divisão política e aceleração tecnológica.....	27
Trabalho e convencimento.....	29
Surgimento nos anos 1980, cisma nos anos 1990.....	39
O elogio à velocidade.....	58
Conclusão.....	70
Cap. 3. Free e open do 9o Fórum Internacional de Software Livre.....	74
O Fisl.....	77
O surgimento do Fisl: entre movimentos sociais e partidos de esquerda.....	79
O Fisl em 2008.....	87
Hackers, políticos e o público.....	92
Nas imagens, as filiações.....	96
Na abertura, as autoridades fazem o choque entre free e open.....	104
O encerramento.....	120
Conclusão.....	131
Cap. 4. Nerds e geeks.....	137
Richard Stallman: de líder a motivo de piada.....	140
No caminho, os nerds.....	150
O verão do código.....	157
O ethos dos executivos e o capital humano.....	164
Conclusão.....	169
Cap. 5. Hackers.....	172
Hacker não é cracker, ser hacker é uma honra.....	174
O surgimento da ideia de hackers.....	175
Como se tornar um hacker.....	180
A ética hacker, o espírito da era da informação e o trabalho.....	187
O ser hacker em discussão.....	191
Conclusão.....	200

Conclusões gerais.....	203
Bibliografia.....	210
Anexos.....	218
Como se Tornar um Hacker.....	219
General Public License (GPL).....	230
Sulamita Garcia - Cabelinho enebado não dá!.....	239

Apresentação

Antes de iniciar o percurso da tese propriamente dito, acho importante expor como se deu minha inserção no campo de pesquisa e mostrar a partir de qual perspectiva escrevo e me posiciono no contexto político do próprio software livre. Aqui, faço uma introdução de caráter mais pessoal, procurando localizar o leitor, explicitando relações que estabeleci com parte do movimento e falando sobre algumas de minhas motivações. No capítulo seguinte, trato mais detidamente dos objetivos desta tese e aponto e discuto as questões gerais.

Meu contato inicial com o software livre se dá por volta do ano 2000 quando, recém saído do curso de graduação em Ciências Sociais e ao término de curso de pós-graduação em Jornalismo Científico, tomo o software livre como tema de notícias e reportagens. Desde logo, interessa-me o viés mais politizado do movimento, em especial o discurso anti-corporativo voltado sobretudo contra a Microsoft. O primeiro passo foi tentar entender a viabilidade daquilo tudo, como seria possível economicamente a construção colaborativa de um sistema operacional completo a ser distribuído de graça. Logo “aprendi” - fui ensinado - que não deveria associar gratuidade ao software livre, mas pensá-lo a partir da ausência de restrição à cópia, cujo efeito seria a gratuidade, mas não sendo esta uma regra. Pensar os efeitos da distribuição livre de conteúdos digitais (que não se referem somente ao software, mas também a música, livros filmes, etc) no sistema de trocas do capitalismo foi o que me atraiu definitivamente para o tema. No início de 2001, estive no primeiro Fórum Social Mundial, em Porto Alegre, em uma de minhas primeiras experiências de cobertura jornalística de eventos. Nesse

fórum, esteve presente Tim Ney, diretor da Free Software Foundation, o que certamente contribuiu para minha percepção do software livre como ligado às alternativas ao processo de globalização neoliberal e integrado a processos de resistência como a dos movimentos camponeses aos transgênicos e à Monsanto.

O primeiro contato que estabeleci com grupos do software livre foi por meio de uma lista de discussão, o Quilombo Digital. O grupo da lista, recém reunido, criou um manifesto em reação à intenção do Ministério da Educação de utilizar recursos do Fundo de Universalização de Serviços de Telecomunicações para compra de computadores com Windows, a serem distribuídos nas escolas. Advogavam pelo uso de software livre em lugar do Windows e afirmavam, entre outros, que ele seria mais seguro e teria uma "filosofia" capaz influenciar positivamente os alunos, que seriam “incetivados a compartilhar conhecimento, ao trabalho em grupo e a pensar no próximo”¹. A lista se mostrou especialmente interessante para mim por discutir o software livre não a partir de seu ponto de vista técnico, mas pelas suas implicações sociais. Como dito pelos próprios membros, tratava-se de uma lista sobre a “filosofia”.

A partir desse contato, passei a escrever ativamente sobre software livre, usando a lista como fonte para pautas e como fonte de especialistas que poderiam me orientar tanto em assuntos técnicos como sobre as divisões políticas do movimento, que até então não eram muito claras para mim. Como não acredito em imparcialidade jornalística, escrevi matérias a partir do ponto de vista do software livre, tornando-me de alguma forma participante do movimento.

Em 2002, estive novamente no Fórum Social Mundial, que naquele ano teve como um de seus participantes o “guru” do software livre Richard Stallman. Em meados do mesmo ano, ocorreu o segundo Fórum Internacional de Software Livre, o qual acompanhei de longe, por meio da intensa

¹ Ver “Não ao Software Proprietário no Setor Público!” em <http://www.midiaindependente.org/pt/blue/2002/01/15203.shtml>

cobertura realizada pelos jornalistas locais, parte deles ligada à estrutura de comunicação do estado, um dos patrocinadores do evento. Nessa época, passo a utilizar software livre em meu computador de trabalho, percebendo como uma contradição defender a ideia do sistema livre e não fazer uso dele no cotidiano.

A partir de meados de 2003, surge a lista de discussão PSL-Brasil, com o propósito de reunir membros de diversos coletivos em favor do software livre espalhados pelo país, em especial os PSLs, sigla para Projeto Software Livre, grupos estaduais e/ou temáticos de defesa dos programas livres. Como já conhecia alguns líderes do movimento por tê-los contatado para entrevistas, fui convidado a integrar a lista, que originalmente era fechada a membros convidados. Nesse mesmo ano, comecei a colaborar, como jornalista, com o website Planeta Porto Alegre, com sede em São Paulo e editado por um dos organizadores do Fórum Social Mundial. A expressão “planeta Porto Alegre” faz referência a o que alguns militantes e autores classificaram como o “espírito de Porto Alegre”, em referência aos valores do Fórum Social Mundial. Não foi necessariamente a minha abordagem do tema software livre que me levou ao Planeta Porto Alegre, mas este acabou se tornando um dos principais temas de meus textos.

No ano seguinte, participei pela primeira vez do Fórum Internacional de Software Livre, então em sua quarta edição. Na época, passei a me envolver em esforços para tornar o software livre, então apenas um dos temas do Fórum Social Mundial, uma realidade de fato no evento, com a adoção de software livres no website do Fórum Social e em sua estrutura administrativa. Nessa empreitada, aprofundei contatos com lideranças do software livre, além de ter tomado contato com alguns grupos internacionais, em especial da América Latina. Um deles é a Hipatia, organização “pela liberdade do conhecimento” que reúne, em especial, membros da Itália, Argentina e Brasil. No final de 2004, participei do III Fórum Social Europeu, realizado em Londres, onde conheci grupos europeus.

Também passei a contatar outros grupos brasileiros de apoio ao software livre, tendo em mente o desenvolvimento de uma atividade chamada Laboratório de Conhecimentos Livres, uma reunião de grupos artistas, ativistas e técnicos em oficina permanente durante o V Fórum Social Mundial, em Porto Alegre. Fui um dos criadores e articuladores da atividade, conhecendo e convidando grupos. Parte substantiva desses grupos já estava sendo reunida nos Pontos de Cultura, programa do governo federal de incentivo à cultura digital e que faz uso prioritário de softwares livres em atividades audiovisuais.

Entre 2003 e 2005, desenvolvi as pesquisas de meu mestrado em Linguística, intitulado "Política e linguagem nos debates sobre o software livre". Nesse trabalho, já discuto as divergências entre os grupos *free* e *open*, de que trato também nesta tese, porém meu enfoque se dá pela análise de textos, em especial de licenças de software, artigos e projetos de lei. Argumento que a GPL, licença preferida pelo grupo *free*, traz em si a ideia de que não deve haver diferenças entre produtores e consumidores de software, sendo um instrumento de introdução de igualdade de condições de produção do bem software (Evangelista, 2005).

No final de 2005, ajudei a criar a primeira edição da CoberturaWiki, uma iniciativa de relato do evento a partir de textos construídos coletivamente em um site wiki. A ideia original foi desenvolvida tendo em mente o Fórum Social Mundial, tendo como objetivo produzir um relato coletivo dos debates mais importantes. Dadas as dificuldades técnicas – dotar a infra-estrutura do Fórum Social de grande número de computadores, incentivando e ensinando o uso de um site wiki – resolvemos aplicá-la em um evento de menor porte e com público mais habituado à Internet e aos wikis, o LACFree 2005, encontro latino-americano sobre software livre, apoiado pela Unesco. Na primeira metade do ano seguinte, voltei a participar do Fórum Social Europeu, desta vez em Atenas, Grécia, também frequentando prioritariamente as – poucas – atividades relacionadas ao software livre.

A partir de 2006, passei a frequentar o Fisl não apenas como jornalista e militante, mas também como palestrante no evento, apresentando as conclusões de minha pesquisa de mestrado. Nesse mesmo ano, ajudei a organizar a CoberturaWiki no Fisl, que desta vez alcança mais sucesso e reúne relatos interessantes, alguns deles utilizados nesta tese – embora este nunca tenha sido o objetivo.

Em 2007, voltei a participar do Fisl como palestrante. Em todo esse período continuei observando o movimento por meio de listas de discussão e por conversas com contatos em São Paulo e Brasília – já que vários militantes passam a trabalhar no governo federal a partir da chegada do Partido dos Trabalhadores ao poder. Nessa época, ingressei no programa de doutorado em Antropologia Social, tendo redigido projeto de pesquisa tendo o software livre no Brasil como tema. É um período em que também reduzo minhas atividades como jornalista e como militante do movimento.

No ano de 2008, participei novamente do Fisl, porém dedicado inteiramente à construção de um caderno de campo. Apresentei palestra no evento, mas sobre assunto lateral ao movimento: discuto um artigo - produzido em co-autoria – sobre as possíveis aplicações do método livre de desenvolvimento de software em textos jornalísticos, questionando a ideia de *open source journalism*. Nessa edição do evento, tive interessantes contatos de pesquisa com pesquisadores internacionais interessados no movimento software livre brasileiro, reconhecido por seu viés politizado. Parte dessas reflexões conjuntas aparecem no texto desta tese.

Parece-me claro não ser possível construir uma separação rígida entre meu percurso pessoal no campo de pesquisa, como militante e jornalista, e minha presença como pesquisador. Contudo, cabe apontar que essa mudança de condição foi acompanhada de uma outra postura em relação ao campo de pesquisa, em que minha interação com este passa a ser mais como observador do que como sujeito ativo nos debates públicos. Não postulo que essa mudança tenha me colocado concretamente em algum tipo de posição “mais neutra”, apenas assinalo um momento diferente em minha posição de

pesquisador. Além disso, passei a empreender um duro esforço de estranhamento das teses e bandeiras as quais defendi nos anos anteriores; não no sentido de buscar necessariamente o ponto de vista oposto – no caso, dos adversários do movimento – mas de desnaturalizá-las, refletir sobre suas contradições, determinações e consequências.

Os anos iniciais de minha presença no campo, no entanto, foram essenciais não somente pelo mapeamento dos agentes, como também pelo acompanhamento de certa forma histórico da evolução dos debates sobre o software livre. Foi a partir deles que pude intensificar meu processo de pesquisa, partindo diretamente para a construção de um caderno de campo – cujo foco principal foi a edição de 2008 do Fórum Internacional de Software Livre – e para a leitura da bibliografia geral e específica. A partir desse caderno de campo e de ideias surgidas na pesquisa bibliográfica, pude muitas vezes retornar a um extenso conjunto de mensagens que acumulei durante anos ao participar de diversas listas de discussão na Internet sobre o tema². Esse retorno aconteceu tanto no sentido de reavivar a memória sobre os debates acompanhados – os quais foram observados em seu contexto – quanto como busca de evidências materiais para o reforço das hipóteses afirmadas.

Neste trabalho, procuro fazer um movimento que vai da apresentação do contexto histórico e ideológico do software livre, onde são apresentados os principais elementos que permitem o entendimento das posições em disputa, passando para uma abordagem mais direta do campo de pesquisa, trazendo relatos de campo acompanhados de análises. Nesses dois momentos o elemento essencial que busco capturar são os conflitos em torno do que é e para que servem o software livre e o movimento em seu entorno. O caminho é buscar entender os significados do software livre não apenas no momento analisado, mas a partir de referenciais que se encontram na história do movimento, assim como em seu diálogo com elementos de vão além dele, como as mudanças contemporâneas do

² A maior parte delas continua com acesso disponível em diversos sistemas de armazenamento de mensagens na Internet. Outras, em especial as listas fechadas ao público, estão armazenadas em arquivos pessoais.

capitalismo.

Como dito, essa pesquisa trabalha com informações e contatos estabelecidos em quase dez anos de presença no campo. Dado que boa parte da atuação do movimento – de seus militantes, grupos e instituições – se dá por meio da interação via computador, foi possível estabelecer um contato continuado, ainda que intermitente, por extenso período de tempo. A partir desse contato inicial foi possível mapear lideranças, relações sociais e principais tópicos de debate do movimento. Já nesse momento foi possível identificar a principal divisão política do software livre, e construir um histórico dela. Em seguida, parti para incursões a campo, onde pude observar os conflitos do movimento em diversos encontros e eventos. Neste trabalho abordo detidamente apenas um desses encontros – cruzando informações coletadas em mais de uma edição desse encontro -, porém a análise desenvolvida é feita a partir de uma presença em campo mais ampla, em diversos eventos. Neles, pude observar e envolver-me em situações e dramas sociais semelhantes – e que de certa forma estão exemplificados – aos aqui descritos.

A maior parte das informações e interpretações aqui contidas não veio diretamente de entrevistas com informantes, mas da observação da interação dos atores em listas de discussão, blogues, fóruns na internet e nos eventos em que estive presente. As entrevistas e conversas com informantes serviram mais como orientação sobre questões técnicas e informações sobre conflitos de bastidores e sobre outros atores. Como o movimento é muito ativo na discussão sobre qual o seu próprio significado e sentido, muitas das perguntas relevantes acabaram sendo formuladas e respondidas pelos próprios militantes.

Esta tese está dividida em cinco capítulos e uma conclusão geral em que procuro sintetizar conclusões parciais acumuladas nos capítulos anteriores.

No capítulo 1, procuro fazer uma apresentação geral do objeto, o movimento software livre,

afirmando sua relevância e procurando mostrar sua abrangência. O movimento software livre, para ser entendido em sua completude, não deve ser limitado apenas ao conjunto de entusiastas profissionais e amadores que efetivamente produzem software, que trabalham com os códigos. Estes são acompanhados por um conjunto complexo de simpatizantes e apoiadores que contribuem de maneira definitiva com a promoção, educação para uso dos softwares, e no sentido de serem apresentadas demandas políticas ao Estado e à sociedade pela adoção softwares que utilizem licenças livres. Em consequência, afirmo o movimento software livre como um movimento social. Nesse mesmo capítulo, afirmo o liberalismo de um modo geral, mas mais especificamente o que vem a ser chamado de neoliberalismo no final do século XX, enquanto grade de pensamento de forte relevância e influência para o movimento. E é a partir desse caminho que aponto um dos paradoxos do movimento software livre no Brasil: sua forte ancoragem nos movimentos de contestação da globalização neoliberal que emergem entre o final do século XX e o início do século XXI.

No capítulo 2, detenho-me no que considero ser a divisão política fundamental do software livre em seu contexto mundial: a existência de dois grupo, *free* e *open*, que polarizam o debate e o fundamentam, inclusive no Brasil. O grupo *open* surge em um momento posterior, em reação ao que seria uma politização excessiva do grupo *free* e tendo em vista a atração de apoio empresarial ao software livre. Procuro expor as diferenças históricas e de argumentos entre os dois grupos e afirmo que ambos disputam por militantes que contribuam voluntariamente em projetos distintos de software.

A partir do capítulo 3, intensifico a exposição e análise de meus dados etnográficos relativos ao movimento software livre brasileiro. Parto da distinção internacional dos grupos *free* e *open* e busco analisá-los no Brasil, contextualizados pela minha incursão no principal evento do movimento software livre brasileiro, o Fórum Internacional de Software Livre (Fisl), realizado anualmente em Porto Alegre. Trago observações de campo realizadas desde 2004, mas me ocupo mais especialmente da edição de

2008 (a nona), elegendo como momentos cruciais da descrição as sessões de abertura e encerramento.

No capítulo seguinte, o de número 4, continuo a relatar extensivamente situações acompanhadas na nona edição do Fisl, porém procuro me deter em um fenômeno que considero ter se intensificado nas últimas edições do evento: o funcionamento do Fisl como evento de recrutamento profissional indireto. A intensificação do uso de software livre por empresas abriu um campo de trabalho profissional importante para profissionais de informática que conhecem softwares livres e sabem trabalhar em suas redes voluntárias de produção. Assim, passam a ser atraídos para o movimento de uma maneira geral, e para o Fisl em particular, jovens estudantes de informática, interessados em construir seu futuro profissional. Argumento que esse processo impacta o movimento e ao Fisl, e analiso alguns dos exemplos e tipos projetados nesse imaginário pessoal e profissional, como os nerds e os geeks.

No capítulo 5, que antecede às conclusões finais, detenho-me sobre o “hacker”, termo que encerra a qualificação de maior prestígio no movimento software livre. Muitas vezes, o software livre é definido como um “movimento de hackers”, ou então os programas utilizados são definidos como “escritos por hackers”. O “ser hacker” ou ter o “espírito hacker” ou agir e pensar de acordo com a “ética hacker” estão em relação de equivalência com o que viria a ser o próprio software livre, seja enquanto movimento, comunidade ou conjunto de softwares. Por meio da análise de alguns textos que tratam do que é ser “hacker” – ou como se tornar um – e de um debate nacional em torno do termo, procuro mostrar como a articulação deste, a escrita frequente de sua definição e o recontar de sua história estão ligados a questões de poder dentro do movimento software livre, servindo como norma para atitudes, posicionamentos, inclusões/exclusões e autoridade.

Cap. 1. O software livre como movimento e uma breve apresentação

Desde meados da década de 1990, um grupo de pessoas vem atuando no Brasil no sentido de propor a adoção e contribuir para o uso do que se convencionou chamar de softwares livres. Esse grupo, nem sempre homogêneo em seus posicionamentos, intitula-se “movimento software livre” e reúne técnicos, desenvolvedores, ativistas, usuários, organizações, empresas, empresários, artistas e intelectuais. Este trabalho é uma tentativa de aproximação desse grupo, uma escrita de caráter etnográfico no sentido de tentar entender o software livre - um movimento que é mundial - a partir de quase dez anos de experiência de campo junto a militantes brasileiros.

Embora tenha nascido nos Estados Unidos, em meio à popularização do uso dos micro-computadores, o movimento software livre ganhou especial relevância política no Brasil. Após menos de dez anos de atuação, já ganhou destaque pelo seu número de integrantes, pelo tamanho de seus eventos (o Fórum Internacional de Software Livre, realizado anualmente em Porto Alegre, Rio Grande do Sul, está entre os dois maiores do mundo) e por sua influência junto a governos municipais, estaduais e federal. Há leis aprovadas e diversos projetos de lei tramitando em câmaras municipais, assembleias e no Congresso Federal que pleiteiam, de diferentes formas, o uso preferencial de

softwares livres por parte de órgãos da administração do Estado³. Na imprensa internacional, o Brasil já foi classificado, em matérias de publicações especializadas, como “o maior e melhor amigo do software livre”⁴. O então ministro da Cultura, Gilberto Gil, se disse apoiador da “ética hacker”⁵ e classificou a si mesmo como um “hacker”, termo utilizado por integrantes do movimento para, entre outras adjetivações possíveis, qualificar seus membros mais importantes. Não é exagerado afirmar que os brasileiros conseguiram um grau de influência e penetração na política e nos partidos tradicionais maior do que qualquer outro grupo de defesa do software livre no mundo.

Porém, até que se chegasse ao momento atual, de vitalidade e respeito institucional, os integrantes do movimento software livre debateram, criaram organizações, escreveram textos, construíram veículos de imprensa (a maioria na Internet), deram entrevistas, fizeram protestos, realizaram diversos eventos locais, ações e manifestações. Essa atuação teve como objetivo principal a popularização do software livre e, para isso, foi preciso adotar diversas práticas e agir em diferentes frentes: explicar o que são os softwares livres e convencer novos usuários e a imprensa de suas vantagens; oferecer apoio técnico aos usuários, para que estes não abandonassem os softwares livres por falta de suporte; pressionar instituições públicas a adotarem padrões técnicos que permitam a troca de arquivos compatíveis com os softwares livres; participar de programas de inclusão digital com software livre; e, finalmente, desenvolver softwares livres, participando da comunidade internacional e ajudando a garantir a existência de um conjunto completo e crescente de programas não-proprietários.

³ No Congresso Federal tramitam atualmente seis projetos diferentes, apensados ao primeiro projeto apresentado em dezembro de 1999, pelo deputado Walter Pinheiro (PTBA) (Ver http://www.camara.gov.br/Internet/sileg/Prop_Detalhe.asp?id=17879)

⁴ “Governments push opensource software”, em *CNet* <http://news.com.com/21001001272299.html>; “Brazil adopts opensource software”, em *BBC News* <http://news.bbc.co.uk/1/hi/business/4602325.stm>; “Brazil: Free Software's Biggest and Best Friend”, em *New York Times* <http://select.nytimes.com/gst/abstract.htmlres=F40614FD395B0C7A8EDDAA0894DD404482>. Todos os endereços foram acessados em 24/10/2006.

⁵ “Gilberto Gil: 'Yo impulso la ética hacker'”, em *Clarín.com* <http://www.clarin.com/diario/2006/05/29/um/m-01204505.htm>. Acesso em 24/10/2006.

A proposta deste trabalho é considerar todo esse amplo grupo como igualmente relevante para o movimento, e não somente os setores mais técnicos. A literatura sobre o movimento software livre, que o pensa tanto em sua configuração global como brasileira, em grande parte tem escolhido como objeto de investigação as empresas ou os indivíduos envolvidos mais diretamente na produção e melhoria dos softwares (Coleman, 2005; Keltz, 2008; Weber, 2004; Sanchez, 2007; Apagua, 2004; entre outros), muitas vezes referindo-se a eles como “hackers”, para designar o grupo em seu caráter específico⁶.

E é nesse sentido que considero produtivo entender o software livre como um movimento social, em meio a outras manifestações de caráter semelhante e que se acumulam a partir do século XX e às quais alguns autores classificam como “novos movimentos sociais”. Goss e Prudencio (2004) resumem assim a posição de Laclau (1986) com relação às limitações do conceito tradicional de movimentos sociais frente à emergência dos novos movimentos.

Os “novos movimentos” que surgem na América Latina não se baseiam mais em um único modelo totalizante de sociedade, como ocorria anteriormente. Segundo o autor, as organizações tradicionais, como sindicatos, partidos políticos e movimentos de trabalhadores eram definidas por meio da conjugação de três características: a identidade dos atores determinada por categorias relacionadas à estrutura social — camponeses, burgueses e trabalhadores —; o tipo de conflito definido por um paradigma evolucionário, ou seja, haveria um esquema teleológico e objetivo que guiaria as lutas (o socialismo); e, por fim, os espaços dos conflitos reduzidos a uma dimensão política fechada e unificada (representação de interesses, institucionalidade política). Os “novos movimentos sociais” romperam justamente com a unidade desses três aspectos. Em relação ao primeiro, a posição que o sujeito assume nas relações de produção não determina necessariamente suas demais posições. No que se refere ao segundo aspecto, não é mais possível determinar a realidade por meio de estágios que apareceriam em sucessivas fases do desenvolvimento da sociedade. Finalmente, o político é uma dimensão presente em toda prática social e não um espaço específico.

Buscando evitar uma banalização do conceito de movimento social, derivada do uso dessa

⁶ Aqui, ao contrário, pretendo entender – e discutir – o termo hacker como categoria nativa utilizada como ferramenta de distinção de certos membros, notadamente os de mais prestígio.

classificação para qualquer tipo de ação coletiva, Touraine (2000) propõe a diferenciação em três tipos, os movimentos sociais, culturais e históricos. Os movimentos sociais combinariam um conflito social com um projeto cultural e defenderiam um modo diferente de uso dos valores culturais. O movimento social teria consciência do conflito com um adversário social (Touraine, 2000; Goss e Prudencio, 2004).

É interessante refletir sobre alguns desses aspectos com relação ao movimento software livre. Em uma análise inicial, é possível identificar a presença forte de profissionais de informática, como seria de se esperar, tanto no contexto nacional quanto internacional. E, como não poderia deixar de ser, são eles que de fato trocam códigos e produzem os programas livres. Porém, além desse conjunto de indivíduos também contar com profissionais de outras áreas, que produzem programas de computador como atividade de lazer, nas horas vagas, ser programador ou desenvolvedor não é condição necessária para alguém entender a si mesmo como membro do movimento. Para fazer parte do software livre é preciso defender certos valores e certas práticas que estão, no limite, identificadas com o que o movimento chama de “cultura hacker”, mas que se traduzem, no cotidiano, na defesa das posições políticas do movimento e no uso de determinados softwares.

Um outro elemento importante é o adversário político do movimento. Formalmente, nos documentos e manifestos que circulam entre os membros, esse adversário são as “licenças proprietárias” de software. Estas seriam as licenças em que os usuários de software não têm garantidas as quatro liberdades consideradas essenciais: o direito de usar o software sem qualquer restrição; o direito de ler e estudar o código-fonte, a “receita” do programa; o direito de alterar o código-fonte para produzir uma versão diferente; e o direito de fazer cópias do software e distribuí-las a quem desejar. Porém, veremos como esse adversário pode ser entendido, de acordo com cada diferente corrente política do movimento, como a Microsoft, as grandes corporações, o capitalismo, o modelo de negócio

do software proprietário, ou até mesmo desaparecer, constituindo-se o movimento como simplesmente a construção auto-suficiente e independente de uma alternativa.

É impossível falar do software livre, enquanto movimento, sem falar de suas ambiguidades, contradições e divisões. Certamente, são elas que permitem que segmentos sociais diferentes, muitas vezes com interesses opostos – como ativistas anti-globalização e empresas – encontrem algum tipo de representação a partir da qual descreverão o adversário e os objetivos do movimento com diferentes matizes (Coleman, 2004). Tomo como divisão política primeira do software livre o que chamo de grupos *free* e *open*. Esses grupos, a partir de um mesmo conjunto de valores gerais, mobilizarão argumentos diferentes, seduzindo grupos distintos a participarem do movimento.

No artigo “Copyleft vs. Copyright: A Marxist critique”, Johan Söderberg (2002) discorre um pouco sobre o que classifica como “movimento hacker” e a política: “O movimento hacker é um projeto político”. Segundo ele, entretanto, o movimento é bastante diverso, unindo desde anarco-socialistas até libertários (aqui ele refere-se liberais radicais norte-americanos, inclinados à direita) *high-tech*. O que os uniria seria a oposição à Microsoft. A partir da definição de Richard Barbrook (1995) para o que este chama de “Ideologia da Califórnia”, Söderberg (2002) considera o fenômeno do open source (código aberto) como uma tendência mais à direita do movimento hacker. Em “Californian Ideology”, Barbrook caracteriza 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. A antropóloga estadunidense Gabriella Coleman (2004) publicou diversos trabalhos em que analisa as motivações e o movimento software livre, em especial a comunidade Debian dos Estados Unidos (Debian é o nome de um determinado empacotamento padrão de softwares livres que formam um sistema operacional e aplicativos; há diversos empacotamentos disponíveis). Ela

argumenta ter o código-fonte (linguagem em que o software é escrito, linguagem esta que pode ser compreendida por humanos), para os desenvolvedores, um estatuto equivalente ao de um texto. Lutar pela livre troca de códigos-fonte seria, então, similar à luta pela liberdade de expressão. Esse argumento, embora pareça bastante válido e seja certamente verificável para desenvolvedores de software livre dos Estados Unidos, não explica a penetração do movimento em lugares onde outros ideais têm maior força, tradição e premência do que a liberdade de expressão. Além disso, deve-se enfatizar que os desenvolvedores, aqueles que efetivamente podem se expressar pelo código-fonte, são apenas uma parte, embora bastante importante, do movimento software livre.

No Brasil, embora a disseminação do software livre seja o fim último de suas ações, muitos dos membros do movimento não consideram a expansão do uso desses softwares “um fim em si mesmo”. Os softwares livres são entendidos como parte de uma luta social, como a afirmação de certos valores ou um caminho para mudanças sociais. Ao falar no movimento software livre estadunidense, Coleman (2004) afirma a existência de uma forte rejeição à política tradicional (“agnosticismo político”), articulado à defesa da liberdade para programar, entendida como liberdade de expressão. No Brasil, contudo, parte considerável daqueles que se entendem como pertencentes ao movimento software livre e que se identificam com suas lutas, não são desenvolvedores. A defesa da liberdade de expressão é sim, como veremos, um dos valores, mas agrega-se a outros: os softwares livres seriam um meio para se atingir, por exemplo, a “justiça social e o desenvolvimento sustentado”⁷. Grupos diferentes do software livre (dentro de um mesmo país ou de regiões distintas) parecem atribuir sentidos e finalidades diversas à revisão do sistema de propriedade intelectual que propõem como consequência

⁷ Na seção “o projeto”, do site do Projeto Software Livre Brasil, grupo que reúne membros históricos do movimento brasileiro, escreve-se: “o projeto investe na produção e qualificação do conhecimento local a partir de um novo paradigma de desenvolvimento sustentado e de uma nova postura, que insere a questão tecnológica no contexto da construção de mundo com inclusão social e igualdade de acesso aos avanços tecnológicos”. Disponível em <http://www.softwarelivre.org/theproject.php>; acesso em 27/10/2006.

da disseminação de softwares livres. Mas, para todos esses grupos, o uso de programas não-proprietários configura-se como meio para a afirmação de certos direitos ou a promoção de novas relações sociais.

O objetivo aqui não é fazer uma análise detalhada desses grupos, mas apontar sua existência e mostrar sua articulação com as duas correntes políticas principais já comentadas, *free* e *open*. Além disso, procuro indicar como o discurso à esquerda e a presença de Richard Stallman – estadunidense fundador do movimento e tido como um de seus representantes mais radicais –, que esteve no Brasil por diversas vezes entre o final da década de 1990 e o início do século XX, tiveram impacto e foram a primeira influência direta ao movimento no Brasil. Este cresceu principalmente em torno de um evento organizado por sindicalistas e militantes de movimentos sociais, o Fórum Internacional de Software Livre. Ao longo dos anos, contudo, a força internacional do grupo *open* cresce progressivamente, a partir de empresas que usam o software livre como base de seu negócio e que passam a ocupar posição dominante no mercado de software e da Internet. Procuro identificar o impacto dessa ascensão do modelo *open* – um modelo de negócios que envolve a venda de serviços associados aos programas e não o licenciamento dos programas em si – junto aos membros do movimento software livre.

O neoliberalismo enquanto modo de pensar

O movimento software livre atua em um dos setores de maior crescimento do capitalismo contemporâneo: o das tecnologias de comunicação e informação. Sua ação se dá em relação e é acompanhada de perto por grandes corporações, com fortes interesses comerciais. Além disso, muitos de seus membros trabalham, já trabalharam ou desejam trabalhar nessas grandes empresas.

David Harvey (2008) aponta as corporações, junto com os meios de comunicação e certas instituições da sociedade civil, como os principais vértices de “geração de consentimento popular para legitimar a virada neoliberal” ocorrida a partir de meados dos anos 1970. Segundo ele, as mudanças em termos de política econômica realizadas pelos governos Reagan, nos EUA, e Thatcher, na Inglaterra, necessitaram antes da construção de um consentimento político em boa parte da população. Usando a ideia de Gramsci de senso comum (“o sentido sustentado em comum”), Harvey afirma que este foi operacionalizado usando-se especialmente a palavra liberdade. “A palavra 'liberdade' ressoa tão amplamente na compreensão do senso comum que têm os norte-americanos que se 'tornou um botão que as elites podem pressionar para abrir a porta às massas' a fim de justificar quase qualquer coisa”. (Harvey, 2008: 50).

O apertar do botão de que fala Harvey abriu as portas para o que Michel Foucault descreveu, ainda em 1979, como a utopia liberal. O movimento consciente de construção dessa utopia, em contraposição às que a esquerda vinha construindo há anos, poderia ser lido no economista austríaco Friederich Hayek. Segundo Foucault, o neoliberalismo americano seria mais do que uma opção econômica, “mas um estilo geral de pensamento, análise e de imaginação” (Foucault, 2008: 302).

“...o liberalismo americano não é – como é na França destes dias [1979], como ainda era na Alemanha no imediato pós-guerra – simplesmente uma opção econômica e política formada e formulada pelos governantes ou no meio governamental. O liberalismo, nos Estados Unidos, é toda uma forma de ser e de pensar. É um tipo de relação entre governantes e governados, muito mais do que uma técnica dos governantes em relação aos governados. Digamos, se preferirem, que, enquanto num país como a França o contencioso dos indivíduos em relação ao Estado gira em torno do problema do serviço e do serviço público, o contencioso nos Estados Unidos entre os indivíduos e o governo adquire ao contrário o aspecto do problema das liberdades. É por isso que eu creio que o liberalismo americano, atualmente, não se apresenta apenas, não se apresenta tanto como uma alternativa política, mas digamos que é uma espécie de reivindicação global, multiforme, ambígua, com ancoragem à direita e à esquerda. É também uma espécie de foco utópico sempre reativado. É também um método de pensamento, uma grade de análise econômica e

sociológica.” (Foucault, 2008: 302)

Veremos a seguir que, ao usar a palavra “liberdade”, o software livre parece ter permitido essa dupla ancoragem de que fala Foucault, tanto à direita como à esquerda. Neste momento, contudo, quero ressaltar o neoliberalismo como a utopia e a grade de pensamento em ascensão quando da disputa sobre a ideologia do movimento de que tratarei aqui. A oposição entre direita e esquerda, que aparecerá no conflito entre *open* e *free*, se dá nos termos colocados principalmente pelo pensamento neoliberal de modelo estadunidense, a partir de suas questões e grade de pensamento. É fato importante, também, a data de nascimento do movimento – meados dos anos 1980 – e o período de sua popularização em nível internacional – o final dos anos 1990 –, quando as posições até então mantidas pelo software livre entram em choque e precisam responder e conformarem-se às questões colocadas pelo neoliberalismo. Temas como o tamanho do Estado; papel da inovação, empreendedorismo e mercado como geradores de riquezas; práticas monopolistas e competição; futuro do capitalismo e do trabalho; entre outros, são discutidos ativamente pelo movimento a partir das ideias de liberdade e cultura de compartilhamento e abertura que o norteiam, mas também sob o signo de ideias econômicas e sociais gerais que circulam pela sociedade.

Da arte ao software

Desejo aqui fazer uma aproximação introdutória para dar termos aos personagens que estamos discutindo aqui, muitos dos quais acabam por estabelecer ligações – profissionais ou amadoras – na produção de softwares. O artista é socialmente reconhecido como aquele que, por meio de suas obras,

expressa ideias ou habilidades que servem à reflexão e/ou ao deleite dos sentidos do público. Já o desenvolvedor de software é entendido como aquele que, sob a tutela de uma empresa e em conjunto com outros desenvolvedores, cria programas de computadores que permitirão às máquinas realizar atividades específicas. Ambos, o artista e o desenvolvedor, pertencem ao grupo dos trabalhadores intelectuais. Porém, como criam produtos que são socialmente utilizados de maneira diversa, são profissionais vistos – e que se veem – simbolicamente de maneira diferente.

O programa de computador como o usuário comum o conhece está em sua forma executável. Isso significa que ele está pronto para “rodar” em uma determinada arquitetura do processador do computador. Programas executáveis são formas últimas e específicas dos softwares. Contudo, nenhum desenvolvedor escreve um programa em sua forma executável. O trabalho desses profissionais é desenvolvido no que se conhece como código-fonte, que são conjuntos de instruções brutas dadas à máquina, escritas em linguagem especializada. O código-fonte, para se tornar executável, deve ser processado pelo que se conhece como compiladores, que são programas capazes de traduzir o código-fonte em código executável de acordo da estrutura do processador em que o software será executado. Um programa compilado para a arquitetura 386, por exemplo, a mais comum nos PCs domésticos, só funciona corretamente nessa arquitetura, assim como um programa compilado para uma arquitetura SPARC somente funciona nela.

Desenvolvedores de software interagem entre si em seu trabalho coletivo por meio da troca de códigos-fonte. Um código executável é ilegível para um humano, serve apenas para a máquina. Já o código-fonte pode ser escrito em qualquer uma das diversas linguagens da computação. Um desenvolvedor, que conheça a linguagem, é capaz de ler o código-fonte e imaginar o que o programa vai fazer. Existe, inclusive, uma determinada “estética” para o código, um desenvolvedor pode dizer se o código está bonito ou mal-feito, limpo ou poluído – o que em geral significa que está mal organizado,

que possui redundâncias ou que leva a máquina a desperdiçar processamento em funções inúteis. Via de regra, os códigos-fonte produzidos coletivamente, em especial no ambiente da Internet, carregam consigo comentários de seus autores que dão detalhes sobre o que faz o código, para que serve cada parte dele. Os comentários no código são um meio de comunicação entre desenvolvedores de um mesmo código e possuem características de qualquer comunicação humana – envolvem juízos, normas, brincadeiras, etc.

Do software ao mercado

Nesse mesmo sentido, acho importante reforçar a perspectiva presente nesta pesquisa sobre a inserção dos trabalhadores do campo da informática no mercado de trabalho. Em "O Imaterial", André Gorz trata tanto profissionais da arte como desenvolvedores de software como trabalhadores de uma hipotética “economia do conhecimento”. Esta traria “transtornos importantes para o sistema econômico”.

“Ela [a economia do conhecimento] indica que o conhecimento se tornou a principal força produtiva, e que, conseqüentemente, os produtos da atividade social não são mais, principalmente, produtos do trabalho cristalizado, mas sim do conhecimento cristalizado. Indica também que o valor de troca das mercadorias, sejam ou não materiais, não é mais determinado em última análise pela quantidade de trabalho social geral que elas contêm, mas, principalmente, pelo seu conteúdo de conhecimentos, informações, de inteligências gerais. É esta última, e não o trabalho abstrato mensurável segundo um único padrão, que se torna a principal substância social comum a todas as mercadorias. É ela que se torna a principal fonte de valor e lucro, e assim, segundo vários autores, a principal forma do trabalho e do capital.

O conhecimento, diferentemente do trabalho social geral, é impossível de traduzir e de mensurar em unidades abstratas simples. Ele não é redutível a uma quantidade de trabalho abstrato de que ele seria o equivalente, o resultado ou o produto. Ele recobre e designa uma grande diversidade de capacidades heterogêneas, ou seja, sem medida comum, entre as quais o julgamento, a intuição, o senso estético, o nível de formação e de informação, a faculdade de aprender e de se adaptar a situações imprevistas; capacidades elas mesmas

operadas por atividades heterogêneas que vão do cálculo matemático à retórica e à arte de convencer o interlocutor; da pesquisa técnico-científica à invenção de normas estéticas.” (Gorz, 2005: 29)

Acredito não ser o caso, aqui, de afirmar ou não a quantidade de trabalho social geral na determinação do valor de troca das mercadorias – até porque penso que, de fato, se não se trata mais de *quantidade* de trabalho social geral na criação do valor; o que gera o valor, como não poderia ser de outra forma, ainda é o trabalho social geral, embora se não quantificável em termos de horas, mas na forma dos tais “conteúdo de conhecimentos, informações” e “inteligências gerais”. Trata-se de ressaltar como, na criação de valores de troca, ganham ênfase as tais “capacidades heterogêneas” como intuição, senso estético, julgamento, retórica e invenção de normas estéticas. Como diz o próprio Gorz, vale a capacidade de colocar a “invenção no mercado como produto de marca patenteada” (Gorz, 2005: 42) e, acrescento, vale a capacidade de criar, em torno desse produto, necessidades e valores que o tornem objeto de desejo dos consumidores.

Alguns exemplos de Gorz nos ajudam a vislumbrar esse processo. Ele observa a nova divisão do trabalho entre empresas e capitais.

“O capital material é abandonado aos 'parceiros' contratados pela firma-mãe, que por sua vez assume para eles o papel de suserano: ela os força, pela revisão permanente dos termos de seu contrato, a intensificar continuamente a exploração de sua mão de obra. Ela compra, a um preço muito baixo, produtos entregues pelos contratados, e embolsa ganhos bastante elevados (...) revendendo-os já com sua marca. O trabalho e o capital fixo material são desvalorizados e frequentemente ignorados pela Bolsa, enquanto o capital imaterial é avaliado em cotações sem base mensurável.” (Gorz, 2005: 34)

Temos então que a empresa detentora dos direitos de produção, do desenho, da marca, não é mais responsável pela produção material, apenas pela concepção do produto, pela ideia e sua comercialização. Todos os produtores de tênis trabalham com os mesmos fornecedores em potencial,

que competem ferozmente entre si em termos de execução do projeto e preço mínimo. Cabe aos detentores dos direitos, “produzirem-se”, como diz o próprio Gorz, construir uma imagem artística de si.

Porém, talvez também seja interessante complexificar ainda mais a chave de compreensão por ele utilizada. Não são apenas as empresas que se produzem como pessoas: dadas as novas regras de contratação de trabalho, são pessoas que são contratadas como empresas e que, a partir dessa lógica – a do “empresário de si” (López-Ruiz, 2004) – precisam administrar suas carreiras também como se fossem empresas, buscando “parceiros” e não patrões. Ao mesmo tempo, nesse mesmo processo, os “empresários de si” também produzem suas marcas, procuram agregar valor ao seu patrimônio que, no caso, reduz-se a essa reputação.

Neste ponto, é interessante trazermos algumas considerações de Jean Lojkin (2007) acerca do “novo assalariado informacional”. Lojkin fala de “potencialidades contraditórias” da revolução informacional, chamando a atenção para uma nova configuração do conjunto dos trabalhadores em que o grupo assalariado decai, de maneira estatisticamente invisível pelas categorizações tradicionais, e caminha para assemelhar-se, proletarizando-se, aos grupos sociais inferiores. Estaríamos frente, então, a um “arquipélago salarial”, que ele descreve da seguinte forma:

“...A revolução informacional, no contexto capitalista atual, leva a uma reorganização das divisões das classes sociais (marcadas até agora pela divisão operários/quadros) em torno de três grandes pólos informacionais: o grupo que monopoliza as informações estratégicas (capitalistas proprietários dos principais meios de produção e de troca, grandes acionistas, quadros do estado-maior, os diretores executivos da esfera pública e da privada que se apropriam dos principais stock-options), o grupo que organiza e elabora a gestão das grandes empresas (quadros intermediários que perderam o monopólio da organização do trabalho, *employés* que têm uma autonomia de gestão) e, por fim, os executivos que criam, coletam, trocam as informações 'operacionais' (operários e *employés*, mas também *experts* muito qualificados em uma especialização técnica particular).

“Se, contudo, considerarmos o maior fato desses últimos anos, a saber, a

precarização e a desqualificação dos quadros intermediários da informação e dos profissionais intelectuais do setor público (professores, pesquisadores) podemos formular a hipótese de uma forte tendência à bipolarização de nossas sociedades capitalistas desenvolvidas. De um lado, de fato, assiste-se à pauperização, à desqualificação e à precarização dos quadros intermediários encarregados da organização da produção, das profissões intelectuais encarregadas da formação e da organização da sociedade: professores, assistentes sociais, profissionais da saúde, trabalhadores da informação e da cultura; e, de outro, se fortalece os privilégios de uma elite dominante que monopoliza o capital econômico, as informações estratégicas e as redes relacionais do poder econômico, político e ideológico...”

Tratar-se-ia, então, de um novo desenho do mundo do trabalho. De um lado, os detentores não apenas dos meios de produção e distribuição, mas também aqueles que controlam redes de poder, relações que implicam na valorização e operacionalização de atividades de produção e comércio. De outro, um conjunto complexo de trabalhadores do material e do simbólico, técnicos, profissionais da área de serviços e trabalhadores industriais em processo crescente de homogenização de suas condições profissionais. Aos sindicatos, partidos e associações progressistas caberia “costurar novos laços” entre esses trabalhadores, estabelecer alianças simbólicas.

É interessante como Lojkine elege como “figura simbólica desse novo trabalho informacional” o que ele chama de

“intermitentes do espetáculo, artistas aos quais é preciso somar-se os profissionais da informação e da mídia (...). De um lado, esses trabalhadores quase 'independentes' têm uma larga margem de iniciativa para conceber, criar, valorizar sua personalidade, mas de outro a esperança de êxito choca-se com a sombria realidade de um mercado de trabalho sem regras formalizadas (particularmente sem certificação), onde os múltiplos intermediários entre a empresa sub-contratante e o prestador de serviços estão no limite do crime de intermediação de mão-de-obra, enquanto que o sucesso de alguns artistas não pode esconder a exploração desavergonhada que vivem esses 'condenados do cachê’”.

É neste ponto que acredito ser possível estabelecer uma aproximação entre a construção artística da *personality*, o “produzir-se” de que fala longamente Gorz, e a emergência de um conjunto de

trabalhadores do simbólico apontada por Lojkin. Tendo como eixo principal as novas tecnologias de informação e comunicação e nela integrando artistas, publicitários, executivos de baixo ou intermediário escalão, desenvolvedores de software, professores, vendedores, entre outros, assistimos à emergência de um conjunto complexo de profissionais ocupados do desenho, manufatura e agregação de valor a objetos culturais vendidos no mercado sob a forma de softwares, games, tênis, filmes, livros, revistas, músicas, roupas, etc. Esse profissional coloca a si mesmo no mercado, vende sua força de trabalho àqueles que controlam as redes de valorização e comércio desses produtos culturais, por meio da construção de si mesmo como *personality* pública, mesmo que de circulação restrita a redes de relações específicas. Não é raro ouvir profissionais de comunicação ou de tecnologia dizerem que: “não contratam ninguém que não tenha um blog na Internet”⁸. Não basta produzir dentro do espaço de trabalho, é preciso colocar-se publicamente como alguém portador de uma “assinatura”.

Ao mesmo tempo que essas novas tecnologias e os processos sócio-políticos que a acompanham (a globalização financeira e os novos arranjos da produção sendo a face mais evidente) precarizam as antigas condições de trabalho, estruturalmente também parecem ser criadas condições para arranjos alternativos da produção cultural/tecnológica (tornada uma coisa só via digitalização). Os sujeitos, em busca de inserção no mercado de trabalho tradicional – seja pela via formal e assalariada, seja na venda do trabalho como free-lancer ou “intermitente do espetáculo” – ou por acreditarem que é possível construir redes paralelas de venda de seu trabalho e produção de novas mercadorias, estão fazendo uso fragmentação, deslocalização e da transformação do imaterial em produto comercializável para garantirem sua sobrevivência, conseguirem melhores condições de vida ou para transformarem o próprio mercado. Em paralelo às redes tradicionais de produção e comercialização, surgem grupos de artistas e profissionais das novas tecnologias de informação e comunicação que se organizam, às vezes

⁸ Ouvi essa frase, especificamente, de um professor de instituição pública universitária em evento público. Ele falava da contratação de pesquisadores para o grupo de pesquisadores que lidera e que investiga “novas mídias”.

à margem do próprio capitalismo, tendo em vista a criação de outros sistemas de trocas⁹. O digital permite a criação de sistemas distribuídos de trabalho visando a construção coletiva de produtos de mesma natureza que os colocados pelo mercado tradicional.

O exemplo mais gritante desse fenômeno é o movimento software livre. Ele reúne desenvolvedores no mundo todo, que atuam sob diversos sistemas organizativos de trabalho, com o objetivo de construir programas de computador que sejam regidas por regras específicas – e mais flexíveis – de propriedade intelectual. Não há um núcleo organizativo central, uma autoridade como a de uma empresa que congrega funcionários e organiza parte do tempo destas pessoas em torno de um plano de trabalho. São diversos pequenos núcleos produtores, em geral reunidos em torno de um software especificamente. Sobre esses grupos há outros, responsáveis pela integração de um conjunto de programas correlatos (softwares para uma determinada interface gráfica, por exemplo). Sobre eles, ou em paralelo, há ainda aqueles grupos que integram os softwares que perfazem um sistema operacional completo (“distribuições” é o termo usado pelos usuários). Ao mesmo tempo, trabalhando com proximidade ou distância dos desenvolvedores, às vezes em intersecção com eles, estão ainda os entusiastas, promotores, designers, educadores, que atuam pela divulgação dos softwares livres, pelo arrebanhamento de novos usuários, encaminhando problemas técnicos às vias corretas, na resolução de problemas jurídicos ou no esclarecimento de dúvidas dos usuários. Todos esses agentes colaboram, interagem e por vezes competem entre si para a criação de produtos e serviços. O que entra para o mercado podem ser os serviços prestados por esses sujeitos, mas muitas vezes são esses próprios sujeitos, que usam esse trabalho em colaboração para tornarem-se conhecidos, demonstrarem suas

⁹ Singer (2004: 12) fala na existência paralela de sistemas de produção não-capitalista e solidários desde a emergência do capitalismo. E acentua que a base para isso é a propriedade social dos meios de produção. “Isso não quer dizer a estatização desta propriedade, mas a sua repartição entre todos que participam da produção social. O desenvolvimento solidário não propõe a abolição dos mercados, que devem continuar a funcionar, mas sim a sujeição dos mesmos a normas e controles, para que ninguém seja excluído da economia contra a sua vontade”

habilidades e às vezes – mas não somente – serem inseridos no mercado de trabalho tradicional.

A adesão desses sujeitos a essa rede alternativa de produção é, na maioria das vezes, voluntária e não regulada por um contrato de trabalho (este pode surgir, eventualmente, mais tarde). Ao se disporem a trabalhar em determinado projeto – que pode ser um pequeno software, uma grande distribuição, uma lista de discussão e agitação política, um fórum de apoio a usuários, um grupo de organização de eventos – os sujeitos são levados a fazer parte de um grupo com características que são geográficas, ideológicas, comportamentais, de gênero, de afinidade, econômicas, etc. Dentro desse grupo, o sujeito submete-se a regras que regulam seu trabalho, informa-se sobre questões políticas e de direito autoral, integra-se em eventos presenciais, faz amigos e discute tecnologia. Os projetos, por sua vez, atuam no sentido de recrutar novos membros, que usam determinados softwares e aderem a certas ideias.

A busca neste trabalho é, ao lado de reconhecer o software livre como um movimento social complexo, cheio de divisões internas e disputas por membros, entendê-lo também como um ambiente que promove a profissionalização de alguns de seus membros. Isso acontece pela proximidade com as empresas e por ser o software livre um movimento em que não apenas apresenta suas demandas como produz suas alternativas em termos de seus próprios programas de computador. Como veremos, esse viés profissionalizante por sua vez tem impacto importante nas disputas internas do movimento.

Cap. 2. Free, open, divisão política e aceleração tecnológica

Neste capítulo, pretendo tratar da divisão política fundamental existente no ambiente em que se produzem os chamados softwares livres. Esse ambiente constrói-se como um movimento social, como iniciativa que trata não apenas da criação de certos objetos (programas de computador), mas da proposição de um modelo para a produção coletiva, consumo e troca de softwares. Em contrapartida, o mercado sinaliza uma tentativa de incorporação do movimento, dos bens comuns imateriais produzidos e do esquema de produção coletiva desenvolvido. Quando falo aqui em “incorporação” não pretendo atribuir a esse processo um sentido necessariamente negativo ou corrompedor, mas indicar, ao nível das ideias, a existência de pontos de afinidade que passam a ser mutuamente reforçados; e, ao nível material, o uso desses softwares livres em projetos comerciais e o financiamento profissional de parte de sua produção.

Entende-se aqui o movimento software livre como o conjunto de pessoas e instituições, públicas e privadas, que promovem publicamente e manifestam-se em favor da adoção maciça ou parcial de softwares livres e/ou do modelo de desenvolvimento aberto proporcionado pelas licenças livres. Opto por falar em “movimento software livre” em lugar de “comunidade software livre”¹⁰. Entende-se também o movimento software livre como um conjunto cultural específico (*cultural set*), que pode ser

¹⁰ O termo “comunidade” concorre com o termo “movimento” enquanto categoria nativa utilizada para designar o conjunto de indivíduos que usa, promove, testa, desenvolve, ensina o uso e/ou promove os software livres. O termo movimento é aqui preferido por referir com mais ênfase também à atividade de defesa pública das qualidades dos softwares livres, ressaltando aspectos que vão além dos internos ao grupo.

estudado antropologicamente, assim como outros conjuntos culturais o são. Embora a cristalização desse conjunto seja bastante recente – pouco mais de duas décadas – e a delimitação enquanto entidade autônoma bastante frágil ou inexistente, penso ser de especial relevância tomá-lo a partir dos termos colocados por Eric Wolf, que afirma a maioria das entidades estudadas pelos antropólogos como devedoras, em seu desenvolvimento, a processos que se originam fora e vão bastante além delas e, em contrapartida, afetam esses processos. (Wolf, 2001: 312). Nesse sentido, como dito, o movimento software livre nasce no seio de mudanças importantes do capitalismo e, em seu desenvolvimento, é afetado de maneira decisiva pela ascensão do neoliberalismo – como doutrina econômica aplicada e também como conjunto de valores sociais. Ao mesmo tempo, afeta o neoliberalismo e a ele oferece novos elementos.

Surgido no início dos anos 1980, o movimento software livre passa a apresentar, a partir do final dos anos 1990, uma disputa bastante clara. Formalmente estabelecem-se dois grupos: o *free*, que afirma ter como luta fundamental a “liberdade” dos usuários de software e ter como horizonte imediato o uso exclusivo de software livres; e o *open*, que embora afirme buscar as mesmas “liberdades” que o *free*, o faz a partir de outras instituições e com diferentes estratégias de luta – por exemplo, colocando o modelo livre de licenciamento de software como uma alternativa a coexistir com o modelo proprietário e argumentando que, acima de tudo, a abertura do código-fonte oferecida pelas licenças livres favorece o desenvolvimento de um software de melhor qualidade. Para o grupo *free* e para o grupo *open* existem instituições, organizações distintas¹¹, às quais indivíduos do movimento software livre podem mostrar-se ligados com graus variados de intensidade. Apenas alguns poucos são formalmente ligados a elas, vários colaboram com uma ou outra em campanhas específicas, sendo que a maioria manifesta apoio e

¹¹ A Free Software Foundation é a mais representativa do grupo *free*, enquanto a Open Source Initiative é a mais representativa do grupo *open*. Essas são organizações gerais de defesa do software livre, mas há diversas outras, que defendem pontos específicos, causas correlatas (como a inclusão digital com software livre) ou têm atuação regional, e que se alinham mais com o grupo *free* ou *open*.

concordância com elas, ou com o conjunto de ideias que representam, de maneira não direta. A fronteira entre os grupos é porosa e o comportamento pela maioria dos indivíduos dificilmente é completamente de acordo com os preceitos de cada um dos grupos.

Essa distinção entre *free* e *open* vai se fundamentar, como veremos, operando no terreno da construção ideológica, ou seja, trata-se da disputa entre duas correntes políticas que, por meio das ideias que divulgam, procuram arregimentar aliados que, por sua vez, mobilizarão trabalho social em benefício do movimento software livre como um todo, mas também mais especificamente em benefício de um dos grupos. Ao racionalizar, justificar, a prática de produção de softwares livres, cada um dos grupos vai apresentar um sentido, uma motivação geral, um propósito diferente (Wolf, 2001: 313). Ambos, contudo, o farão buscando elementos contidos em um ambiente cultural mais amplo e, assim, serão informados e sofrerão os efeitos das transformações pelas quais passa a sociedade capitalista contemporânea.

Trabalho e convencimento

A disputa entre os grupos *free* e *open* em torno da construção daquela que será a ideologia do movimento nos permite discutir ainda como o movimento software livre origina e se estrutura em um determinado arranjo da produção para o desenvolvimento de seus softwares. O grupo *open* faz do elogio às virtudes práticas desse arranjo da produção como o principal argumento para a defesa do software livre¹².

¹² O termo utilizado pelo grupo *open* para se referir ao software livre é *open source*. Utilizo, contudo, o termo software livre para me referir ao conjunto amplo dos softwares defendidos pelos grupos *open* e *free* que, fundamentalmente, é o mesmo.

Como dito, software livre nasce em meio a mudanças importantes do capitalismo, com um crescimento acelerado dos lucros das empresas de tecnologia de informação e comunicação. E colabora para uma mudança no estilo de fazer negócios e de produzir software dessas empresas, cujo modelo principal, até então, era semelhante ao de uma empresa manufatureira: produzia-se e vendia-se software como se fosse um bem material. No software livre, embora também estejam envolvidos em seu processo de produção trabalhadores contratados diretamente pelas empresas, que vendem sua força no mercado – formando parte importante do trabalho utilizado para a produção de softwares livres – o trabalho tido como modelo e simbolicamente ostentado como o mais característico da produção livre é de tipo voluntário, realizado no tempo “de folga” do trabalhador e fora dos espaços típicos de trabalho capitalista (não acontece nem na fábrica nem nos escritórios das empresas). Progressivamente, os softwares produzidos por esse modelo, e a própria ideia de modelo distribuído de produção, tem ganho espaço nas grandes empresas de tecnologia¹³.

Segundo Renata Apgaua, no ambiente do software livre, a partir da etapa em que as corporações passam a se fazer mais presentes, misturam-se elementos do mercado e da dádiva, que a autora pensa nos termos de Marcel Mauss¹⁴. Haveria “nódulos de dádiva” misturados a “momentos de mercado” (Apgaua, 2004). Entende-se, a partir de Apgaua, que, ao oferecer o software para uso livre, o desenvolvedor principal de um projeto obterá a recíproca em termos de colaboração para a melhoria desse software. Nesse sentido, acredito ser correto o apontamento da mistura entre dois modelos.

¹³ O conjunto de servidores que forma a Plataforma Google utiliza versões modificadas do Linux e de outros softwares livres. (Tawfik Jelassi and Albrecht Enders (2004). "Case study 16 — Google". *Strategies for E-business*. Pearson Education. p. 424). Para o desenho de produtos, diversas empresas estão criando softwares em que os próprios consumidores colaboram na criação. A prática é conhecida como crowdsourcing e baseia-se na descentralização da produção do software livre. (<http://www.npr.org/templates/story/story.php?storyId=93495217>)

¹⁴ Diz Apgaua, a partir de Mauss: “Direitos e deveres, que se mostram simétricos e contrários, dão vazão à circulação de dádivas entre os diversos grupos. Tudo circula, as dádivas circulam, mas, na realidade, o que está em jogo são as alianças espirituais. Trocam-se matérias espirituais por meio das dádivas. Os homens estão ligados espiritualmente a seus bens que, quando passados a outrem, estabelecem ligação espiritual com o doador. E, nesse sentido, misturam-se doadores e beneficiários, homens, coisas e matéria espiritual.” (Apgaua, 2001)

Porém, a dívida explica pouco dada a diversidade de projetos de software livre existente. Como entender a escolha feita pelo desenvolvedor sobre com qual projeto livre colaborar? Há uma ampla gama de projetos que oferecem códigos licenciados como livres, como entender as escolhas dos indivíduos sobre a que projetos retribuir ao oferecer, em troca, seu trabalho?

Para compreender melhor esse processo complexo é preciso deixar claros alguns pontos sobre o que é e como se dá a dinâmica do trabalho com o software livre. O software dito livre é aquele que é regulado por determinados tipos de licença que permitem o uso, cópia, alteração e distribuição do código sem restrição prévia de seus autores (exceção feita, em alguns casos, à restrição com relação à mudança da licença). Software é um conjunto de instruções escritas em formato de texto necessárias ao funcionamento dos computadores. Este é, ao mesmo tempo, produto e processo, ou seja, pode ser usado diretamente ou pode constituir a matéria-prima para a construção de um novo software. Em geral, um projeto de software livre que esteja “vivo” implica em desenvolvimento permanente, uma alteração constante do código, pequenas modificações que são lançadas constantemente. Estas são fruto da contribuição de desenvolvedores interessados no projeto e estão disponíveis para que sejam testadas pelos usuários.

Quando é regulado por uma licença não-livre, o software deixa de ser processo tornando-se apenas produto: o proprietário do software restringe a reutilização daquele código, do conjunto de instruções, evitando que seja alterado e dê origem a um novo software. O software livre ou não-proprietário altera o regime de propriedade do código: ele possui autor(es), mas não um dono que controle o destino daquele produto ou que realize com ele as trocas típicas do mercado capitalista. Como o autor do software livre não pode impedir que um usuário que tem em mãos esse software faça uma cópia e entregue a outro usuário, a comercialização do programa é bastante difícil e oferece lucro

muito baixo.

Esse regime de propriedade diferenciado traz consequências para o modo de produção. A não ser que haja algum cliente interessado em uso direto do software, e que possa arcar com os custos totais do pagamento dos trabalhadores, economicamente não é viável financiar a produção total de um software livre como empreendimento comercial. Como não se trata de produzir algo que poderá ser trocado no mercado capitalista de modo típico, a força de trabalho precisa ser arregimentada mediante a sedução de pessoas dispostas a dedicar tempo voluntário ao software livre. As empresas que oferecem seus trabalhadores para a manutenção de algum projeto de software livre em geral o fazem por obterem com ele lucros indiretos (prejudicar uma empresa concorrente, por exemplo, ou vender serviços agregados a esse programa de computador).

Aqui cabe ainda expandir o escopo dos “trabalhadores” do software livre a partir da perspectiva de que o valor do produto software não é criado apenas por aqueles envolvidos diretamente em seu processo de produção: está incluído aqui um conjunto de pessoas que, em suas diferentes atividades, incrementa o valor de uso desses softwares¹⁵. Para que um software seja utilizado de forma plena e com certo conforto por um usuário qualquer, é preciso que este já tenha tido algum contato prévio com o programa de computador. Para um usuário, um software com o qual ele já teve contato em algum momento de sua vida terá maior valor de uso do que um software completamente novo e estranho, com o qual ou ele é incapaz de realizar as tarefas necessárias ou gastará muito mais tempo para isso, pois precisa aprender como operar o novo programa. Além disso, esse mesmo usuário, se sabe usar o software X, mas nunca teve contato com o software Y, não poderá ajudar seus colegas que não sabem usar Y, apenas com relação a X. Não possuir uma significativa base de usuários tem sido um dos

¹⁵ Nesse sentido, é preciso ter em mente a não equivalência entre trabalho e emprego, como afirma Terranova (2000).

principais obstáculos ao crescimento do software livre, dificuldade identificada pelo próprio movimento. Faltaria, na sociedade, um número consistente de usuários avançados ou intermediários, capazes de realizar operações de manutenção simples ou de oferecer instruções básicas sem requerem remuneração profissional para isso. O número de usuários de um determinado software aumenta seu valor de mercado¹⁶, pois esses usuários são possíveis professores informais a quem novos usuários podem recorrer.

Uma vez que o usuário já esteja habituado a utilizar certo programa, deverá oferecer resistência ao uso de um programa diferente ou a uma nova versão do mesmo. A indústria de software proprietário, não-livre, utiliza de diversas estratégias de convencimento ou de pressão para levar seus clientes antigos a usarem uma nova versão do produto. Essa mesma indústria gasta uma quantidade considerável de seus recursos na contratação de profissionais de marketing e propaganda encarregados de enaltecer as qualidades e convencer o público do benefício prático e da economia de tempo futuro ao aprender a utilizar o novo produto à venda. Já no software livre, o recrutamento da maioria desses “profissionais” se dá pelo envolvimento ideológico, pelo convencimento desses entusiastas de que trata-se de algo mais do que promover um produto no mercado, mas sim de que promover o software livre significa incentivar um conjunto de novos valores sociais – conjunto que, como veremos, varia de acordo com os diferentes grupos do movimento software livre. Outro ponto é que os entusiastas, que acabam sendo aqueles que fazem a publicidade do produto, não fazem promoção das novas versões dos softwares – como fazem as empresas de software proprietário interessadas em novas vendas – mas sim dos ideais do software livre. Ou ainda, promovem alguns projetos de software livre específicos, nos quais esse entusiasta veja refletida sua visão do que é um bom projeto de software livre, projetos que

¹⁶ Embora esses softwares não sejam usualmente trocados no mercado de modo típico, sua maior popularidade leva à possibilidade de venda de uma ampla gama de serviços agregados.

reflitam a percepção do grupo com o qual esse entusiasta tenha mais afinidade sobre quais são os ideais do software livre.

Empresas que conseguem auferir lucros com softwares livres também pagam por publicidade e profissionais de marketing. Porém, se comparados com os recursos destinados pela indústria do software proprietário, estes são bastante reduzidos. Além disso, é razoável supor que a maneira tradicional de se promover produtos das empresas capitalistas muitas vezes funcione como publicidade negativa para o software livre, já que parte considerável de seus entusiastas têm resistência à caracterização do software livre como uma mercadoria capitalista. A ideia de que exista alguma grande empresa interessada na promoção do software livre é, de certa forma, um questionamento da imagem de projeto de mudança social propagada por parte considerável do movimento.

Soma-se a isso o fato de que um maior número de usuários é importante no incremento do valor de uso de um software também porque significa um teste pleno do produto. Softwares funcionam de maneira diferenciada de acordo com o equipamento físico (hardware) em que são executados, em que rodam. Um número maior de usuários significa um teste do software em um conjunto mais diverso de hardwares. Em resumo, cada novo usuário conquistado, que adquire os conhecimentos básicos para a operação de um determinado software ou que o opera em uma máquina diferente, significa um incremento no potencial de expansão desse software e em sua qualidade.

Temos então, até o momento, dois grupos de “trabalhadores” do software livre, que são objeto de recrutamento pelos diversos projetos: os usuários, que funcionam como professores em pequena escala e que também são responsáveis por testar o programa em vários modelos de hardware; e os entusiastas, que além disso incentivam publicamente o uso de softwares livres e enaltecem suas qualidades, seja como exemplares de modo de produção e usufruto social mais justo, ou como produto

tecnicamente superior.

O software livre depende principalmente, porém, de um tipo de “trabalhador” ainda mais especializado e cuja contribuição é essencial para o crescimento do movimento como um todo: os desenvolvedores. Como dito, o software livre é desenvolvido, em parte, tanto de uma maneira tradicional, mediante trabalho contratado no mercado capitalista (porém, com o produto desse trabalho sendo disponibilizado com um regime de propriedade diferenciado), como mediante ao voluntariado de desenvolvedores espalhados por todo o mundo, que integram-se em grupos de trabalho na Internet e que oferecem seu tempo e seus conhecimentos. São, então, duas frentes majoritárias de recrutamento do trabalho de desenvolvedores: uma refere-se ao convencimento de empresas e empresários a oferecerem parte ou o tempo integral de seus trabalhadores contratados à produção de softwares livres. Outra é a de recrutamento de trabalhadores voluntários, que associam-se a determinados projetos de software e que trabalham sem remuneração direta.

Tanto o envolvimento das empresas como o dos voluntários acontece por um conjunto razoavelmente definido de motivações. As empresas podem dirigir seus negócios totalmente ao software livre porque vislumbram conseguir lucros cobrando por serviços diversos prestados a seus clientes (instalação, suporte, publicidade, etc). Podem entrar parcialmente no negócio software livre, mantendo a produção de software proprietário que funcione adequadamente com o software livre, que também produzem, mas ganhando mesmo é com as licenças proprietárias vendidas, assim de alguma forma lucrando indiretamente com o trabalho voluntário. Podem decidir pela produção livre por acreditarem ser esse modelo distribuído de produção mais adequado para o desenvolvimento de software de maior qualidade, e por consequência de maior aceitação no mercado, com o qual ela lucrará ao prestar serviços.

Já o envolvimento dos voluntários pode acontecer por razões de militância política, por acreditarem estar impulsionando um sistema não-capitalista (ou capitalista mais justo) de produção. Pode acontecer por questões de afinidade e amizade, tendo em vista a socialização com um determinado grupo de desenvolvedores. Pode acontecer para ganhar experiência em programação, tendo em vista conseguir um bom emprego no futuro. Esse emprego pode ser ainda melhor se seu trabalho for reconhecido como de qualidade pelos seus pares diretos (o grupo de desenvolvedores de determinado software) ou indiretos (o movimento software livre como um todo). O mais provável é que vários desses motivos, e outros não descritos aqui, ocorram simultaneamente, tanto para os desenvolvedores voluntários como para as empresas.

A questão relevante aqui é que o recrutamento para esse trabalho, seja de usuários, entusiastas ou desenvolvedores, acontece tendo como pólo importante de atração uma determinada racionalização, atribuição de sentidos à prática, uma construção ideológica; ou seja, a ideologia, como esquema unificado de ideias¹⁷ que referendam ou manifestam poder (Wolf, 1999), é elemento essencial para se entender o movimento software livre.

Eric Wolf toma o poder não como concentrado em um pacote, não como “uma força unitária e independente, encarnada em imagens como a de um monstro gigante como Leviatã ou Behemoth, ou uma máquina que cresce em capacidade e ferocidade pelo acúmulo e geração de mais poder”, mas como um aspecto de todas as relações entre pessoas¹⁸. E, pensando-o em termos relacionais, o distingue em quatro modalidades pelas quais se enreda nas relações sociais. Penso ser adequado falar aqui na quarta modalidade¹⁹, o poder estrutural, “manifesto não apenas nas relações que operam dentro de

¹⁷ Wolf distingue ideias de ideologia, afirmando que as ideias servem para cobrir uma faixa inteira de constructos mentais tornados manifestos nas representações públicas (Wolf, 1999: 4)

¹⁸ Wolf diz procurar pensar o poder a partir de Norbert Elias, de quem ele teria aprendido que “more or less fluctuating balances of power constitute an integral element of all human relations”.

¹⁹ As outras três formas, além do poder estrutural, seriam a Nietzscheana (ou poder da potência), que repousaria atenção em porque alguém entra num jogo de poder sem se qualificar esse jogo; a Weberiana, manifestada em interações e

configurações e domínios, mas também organiza e orquestra as configurações por si mesmo, e especifica a direção e a distribuição do fluxo de energia” (Wolf, 1999: 5). O software livre, ao propor uma licença de software, uma adequação ao mundo jurídico da propriedade intelectual, que permite e estimula o trabalho colaborativo, coletivo, voluntário conjugado a outras formas tradicionais de trabalho, organizou certas configurações de relações de trabalho e produção. E, na busca por atrair todos os tipos de trabalhadores, os líderes do movimento software livre, ou de projetos específicos em software livre, vão atuar na readequação de velhas ideias para se ajustarem a circunstâncias diferentes, ou apresentarão novas ideias como verdades estabelecidas. Vão organizar e distribuir fluxos de energia produtiva diferenciados (veremos como correntes diferentes do software livre decidem por incorporar ou não certos fluxos produtivos).

E farão tudo isso a partir de determinadas bases culturais onde operam, obtendo mais sucesso junto a certos grupos e em certos lugares e menos em outros (Wolf, 1999:275). O movimento software livre tem seus vilões e heróis: vilões que são quase uma unanimidade, como a Microsoft, símbolo do software proprietário, fechado e não-livre; e heróis, cuja reputação é mais positiva ou negativa dependendo do grupo com que se conversa – além de quase-heróis como o Google, visto com desconfiança por alguns e modelo de empresa perfeita, para outros. Essas distinções e qualificações aparecem em permanente disputa, cujo prêmio é o número de militantes/trabalhadores mobilizados.

Possuir uma base mais ampla de “trabalhadores” para o conjunto dos softwares livres ou de determinados softwares significa um poder maior para o movimento software livre, de uma maneira geral, ou para determinados projetos de software em especial. Correntes ideológicas diferentes no

transações entre pessoas e referindo-se à habilidade de um *ego* em impor sua vontade na ação social sobre um *alter* sem especificar a natureza da arena em que essas ações se dão; e a tática ou organizacional, em que observa-se o contexto em que as pessoas exibem suas capacidades e interagem com as outras, chamando a atenção às instrumentalidades pelas quais indivíduos ou grupos direcionam ou circunscrevem as ações de outros em certas configurações. (Wolf, 1999: 5)

movimento software livre manifestam preferência por softwares distintos. Determinados projetos de desenvolvimento de software se mostram mais hábeis em recrutar usuários e desenvolvedores em nichos específicos de gosto, geográficos ou com certas inclinações políticas. Em determinado momento, dada a manifestação pública, de opinião de algum líder, um projeto de software pode expandir ou retrain sua base de usuários ou desenvolvedores. Os programas são associados a certas correntes ideológicas e significados como “mais livres”, “mais corporativos”, “de hacker”, etc.

Eric Wolf, ao falar sobre os três casos tratados em *Envisioning Power* – os kwakiutl, os aztecas e o nazismo alemão – diz o seguinte sobre o poder estrutural e sobre a relação entre organizadores e organizados:

“In each case, that structural power engendered ideas that set up basic distinctions between the organizers of the social labor and those so organized, between those who could direct and initiate action to others and who had to respond to these directives. The dominant mode of mobilizing social labor set the terms of structural power that allocated people to positions in society; the ideas that came to surround these terms furnished propositions about the differential qualifications or disqualifications of persons and groups and about the rationales underlying them”. (Wolf, 1999: 275)

Em nosso caso, não parece ser correto enfatizar um caráter tão rígido e controlado para a distinção entre dominantes e dominados. É de se afirmar que aqueles que desenvolveram a configuração do modo de produção livre de software estabeleceram e estabelecem as qualificações diferenciais de pessoas e grupos. Porém, veremos como um grupo, o *free*, ao longo dos anos, foi dando lugar e perdendo poder com relação a um outro grupo que ascendeu, o *open*. Assim, o movimento foi ressignificado por uma parcela de seus membros e a disputa sobre quem organiza o trabalho social persiste.

Pretendo mostrar como o modelo de desenvolvimento aberto de software, e os argumentos enfatizados principalmente pelo grupo *open*, ligam-se mais diretamente a um cenário geral e ideológico do capitalismo atual, em especial do neoliberalismo, em que evolução, aceleração tecnológica e a ideia do indivíduo como empresário de si mesmo tem um peso especial. E isso acontece tanto no nível prático da promoção de um modelo de desenvolvimento de software alternativo ao modelo proprietário (o modelo bazar em lugar do modelo catedral), como no nível político de debate entre os grupos *open* e *free*, marcado pelo enfraquecimento progressivo do último e pela predominância do grupo que melhor lidou com a ideia de velocidade progressiva, melhoria tecnológica e lucro. Em lugar de se afirmar que o software livre leva necessariamente à aceleração e à evolução tecnológica, busco entender como a ênfase nessas ideias deu força a uma corrente específica do movimento software livre, o grupo *open*, em detrimento de outra corrente. Não se trata de afirmar um distanciamento completo do grupo *free* com relação a esses argumentos, mas de apontar o quanto os mesmos são centrais e funcionam de uma maneira específica para o grupo *open*. Ao mesmo tempo, procurarei demonstrar como o *open*, embora seja em si uma corrente política do movimento software livre, coloca-se como negação da política, sendo parte importante de seu discurso o predomínio da técnica e da competição em detrimento da negociação e do acordo entre sujeitos e grupos. Não se trata somente do “agnosticismo político” de que fala Coleman (2004) – que, em particular, percebo como mais característico do grupo *open* – mas de uma perspectiva que valoriza a ideia de que a “política” muitas vezes é um obstáculo às soluções técnicas e ditas racionais.

Surgimento nos anos 1980, cisma nos anos 1990

Free e *open* apresentam versões ligeiramente diferentes para o surgimento do movimento software livre. A Free Software Foundation aponta o ano de 1983, com o lançamento do projeto GNU (acrônimo para a expressão em inglês GNU não é Unix) por Richard Stallman, como marco inicial do movimento²⁰. Já a Open Source Initiative descreve um percurso histórico mais longo, atribuindo o nascimento do movimento a uma cultura de compartilhamento de software existente desde a década de 1960, principalmente entre pesquisadores da Universidade de Berkeley, na Califórnia, envolvidos no desenvolvimento do sistema operacional Unix e do BSD (Berkeley Software Distribution). Steven Weber (2004) recupera essa história mostrando as tensões entre a companhia telefônica AT&T, detentora inicial do código do Unix, laboratórios de pesquisa e pesquisadores universitários em torno dos direitos de uso e compartilhamento desses códigos.

Enquanto movimento social com princípios e objetivos constituídos, o triênio 1983-1984-1985 parece ser particularmente relevante. A cultura de compartilhamento de software que Weber localiza especialmente entre os pesquisadores da Califórnia não era algo exclusivo. Contrariado com a impossibilidade de examinar o código-fonte do programa controlador de uma impressora devido a novas regras de propriedade sobre softwares que começavam a se estabelecer, Richard Stallman lança o projeto GNU em 1983. O objetivo era construir um sistema operacional similar ao Unix, mas que obedecesse a uma licença em que os programadores poderiam fazer tudo com o software, menos torná-lo proprietário. Entre 1984 e 1985, Stallman evolui essa ideia e escreve o Manifesto GNU, documento que desenha os princípios do *copyleft*²¹, que dará base para as regras descritas na GPL – a principal licença do software livre, publicada em 1989. O manifesto é um convite para que outros programadores se unam ao esforço da então recém-fundada Free Software Foundation (FSF) de produzir um sistema

²⁰ <http://www.fsf.org/about/what-is-free-software>

²¹ Copyleft é um termo criado para se opor ao copyright e foi criado por Richard Stallman. Segundo ele, a ideia 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.

operacional livre. Em 1984, Stallman abandona seu emprego no Massachusetts Institute of Technology (MIT) para dedicar-se totalmente à causa do software livre. É nesse período que ele delineia o que chama de princípios éticos, 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, a primeira licença redigida tendo em vista os objetivos do movimento²².

Outro ano importante é 1991, quando Linus Torvalds lança a primeira versão do *kernel*²³ Linux, que tornou completo o sistema livre projetado pela FSF, o GNU. Embora seja licenciado nos termos da GPL, o Linux significou, na prática, um forte impulso para uma nova corrente de poder dentro do movimento, que culminará com o ascensão do *open source*, enquanto ideia e grupo político, em 1998. Naquele ano, Eric Raymond publica o artigo “Goodbye, 'free software'; hello, 'open source'” e funda, com Bruce Perens, a Open Source Initiative (OSI)²⁴. Considero aqui a Free Software Foundation como a instituição mais representativa da visão do grupo *free*²⁵ e a Open Source Initiative como instituição que dará suporte inicial às ideias do grupo *open*.

Stallman continua, até hoje, tendo grande influência no movimento. No entanto, a partir de 1991 ele se vê obrigado a dividir o palco com uma então jovem estrela da Finlândia, Linus Torvalds. Carismático, empreendedor, e sabendo usar melhor a internet, ele conseguiu dar solução a um

²² Keltly (2008) conta os problemas que Stallman teve ao tentar compartilhar seu programa EMACS com outros desenvolvedores e suas tentativa de construir em torno do programa uma comunidade/comuna (Stallman utilizava o termo *commune*, mas o termo *community* acabou por se tornar mais popular ao longo dos anos para se fazer referência a esses grupos de usuários e desenvolvedores), preservando-o de empresas que desejavam torná-lo um software proprietário. Segundo Keltly, essas dificuldades serviram de aprendizado para que Stallman desenvolvesse a licença livre GPL.

²³ O kernel é uma parte central do sistema, responsável pela configuração e gerenciamento dos dispositivos (teclado, mouse, monitor etc)

²⁴ Raymond, Eric (1998). “Goodbye, “free software”; hello, “open source”” Visualizado em 27/12/2004 em <http://www.catb.org/~esr/open-source.html>

²⁵ Essa ideia é válida até bastante recentemente. Porém, há indícios que o enfraquecimento do subgrupo *free* tenha sido tão acentuado que suas ideias estejam perdendo força até mesmo dentro de sua instituição fundadora, que permanece bastante atuante.

problema que a FSF se dedicava há anos: construir um *kernel* licenciado sob uma licença livre para ser parte integrante de um sistema operacional livre. A FSF já tinha todo o resto da estrutura do sistema pronta, fruto de anos de esforços, e trabalhava no desenvolvimento de seu próprio *kernel*. Linus foi mais rápido e, usando a GPL como licença, adotou soluções tecnicamente mais eficientes, criando o Linux, parte essencial do sistema operacional.

O método de desenvolvimento adotado por Linus está delineado formalmente em *A Catedral e o Bazar*, livro escrito por Eric Raymond, em 1997. A obra é uma reflexão, elogio e uma descrição do que seria um modelo aberto de desenvolvimento, chamado "bazar". Trata-se, também, de uma alfinetada em Stallman e na FSF, acusados de adotar uma postura centralizadora na organização do trabalho coletivo do projeto GNU. A crítica de Raymond aparentemente é voltada ao modelo de desenvolvimento proprietário, mas também refere-se à FSF ao apontar que, até o trabalho de Torvalds, os códigos eram como se fossem "catedrais", monumentos sólidos construídos a partir de um grande planejamento central. Já o desenvolvimento adotado por Torvalds seria como um bazar, com uma dinâmica altamente descentralizada. Raymond aponta méritos em Torvalds não somente pela liderança no projeto Linux, mas por adotar um relacionamento com seus contribuidores no projeto diferente do até então adotado pelas empresas de software proprietário e pela própria Free Software Foundation. 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 frequentemente 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.” (Raymond, 1997).

A virtude desse novo método de Torvalds estaria, principalmente, na publicação frequente e precoce das alterações feitas no código-fonte. Assim, desenvolvedores de todo o mundo teriam a possibilidade de ler as alterações no código, realizar testes em máquinas diferentes e enviar sugestões de modificações a Torvalds. A essa prática Raymond denominou bazar e aponta suas raízes na cultura universitária dos anos 1960 e 1970.

Mas há mais no que diz Raymond com relação ao modelo Linux do que o elogio da astúcia e da técnica – embora o sucesso desta seja inegável –, há uma disputa de poder sobre quem representa e o que significa o movimento. 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 especial um conjunto de empresas da Califórnia que está tentando transformar o Linux em negócio. No site pessoal de Stallman, 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 estadunidense (os conservadores). Já Torvalds, além de ser politicamente bastante moderado e pragmático, tem uma identidade maior com a então nova geração de programadores então abaixo dos 40 anos, da qual Raymond faz parte. Essa geração, segundo Sam Williams em *Free as in Freedom* – livro que mistura notas biográficas de Stallman com a história do software livre – é mais energética e ambiciosa. Diz Williams: “With Stallman representing the older, wiser contingent of ITS/Unix²⁶ hackers and Torvalds representing the younger, more energetic crop of Linux hackers, the pairing indicated a symbolic show of unity that could only be beneficial, especially

²⁶ ITS/Unix são sistemas utilizados largamente por técnicos até a década de 1980. O GNU/Linux foi construído com uma arquitetura semelhante a desses sistemas.

to ambitious younger (i.e., below 40) hackers such as Raymond” (2002). Stallman representaria a velha geração, o discurso político dos anos 1970, sobrevivente à era Reagan nos anos 1980. Já Torvalds pôde representar os novos programadores, que ascenderam com a bolha da Internet do final da década de 1990 e com o ápice do neoliberalismo, e que hoje aspiram por empregos da nova indústria de tecnologia, com imagem alternativa (mas não anti-capitalista) das novas corporações de informação e comunicação.

Desde a popularização do trabalho de Torvalds, boa parte do tempo de Stallman tem sido gasta em pedidos para que todos se refiram ao sistema operacional, ao conjunto do software, como GNU/Linux e não apenas Linux. O projeto de Torvalds ganhou tanta repercussão que o sistema completo é mais conhecido como Linux. Stallman diz apenas querer que seu trabalho, e de toda FSF, seja reconhecido, já que, sem eles, não teria sido possível a existência do Linux. Dizer Linux ou GNU/Linux também tornou-se um marcador de maior afinidade com o grupo *free* ou com o grupo *open*.

O discurso politizado e o radicalismo de Stallman (que defende que todo software deve ser livre e que o software proprietário é “anti-ético”) não são atrativos para a nova geração de programadores e o são ainda mais indigestos para os empresários, mesmo os ditos modernos novos empreendedores da Internet. Raymond teve um papel decisivo na criação da alternativa mais ao gosto do paladar corporativo. Como dito 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 identificadas²⁷. Esse argumento de Raymond seduziu

²⁷ “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” (Williams, 2002)

executivos da Netscape, dona de um navegador de Internet que havia sido destruído pela ofensiva agressiva – e anti-competitiva, segundo tribunais dos EUA – da Microsoft com seu Internet Explorer. Em 1998, Raymond foi a peça chave no processo de convencimento dos executivos da Netscape para que usassem uma licença livre para o navegador – então comercialmente morto – de modo que a comunidade continuasse seu desenvolvimento. O código do Netscape, tornado livre, deu origem ao Mozilla Firefox, que pouco mais de cinco anos depois passou a rivalizar novamente com o Internet Explorer da Microsoft. O prestígio adquirido por Raymond (tanto pela liberação do código da Netscape como pelo livro *A Catedral e o Bazar*), somado ao do carismático Torvalds, foram essenciais para que o grupo *open* pudesse se estabelecer.

A confusão entre livre e grátis, que na língua inglesa têm o sentido referenciado pela mesma palavra, *free*, foi a justificativa formal para que surgisse o termo *open source*. frequentemente, Stallman procura, chegando a ser insistente, deixar claro que o *free* de *free software* não significa grátis, mas livre. Não há diferenças substanciais entre o que os termos *free software* e *open source* pretendem definir. Ambos estabelecem praticamente os mesmos parâmetros que uma licença de software deve conter para ser considerada livre e aberta. Ambos estabelecem, na prática, que o software deve respeitar aquelas quatro liberdades básicas que a FSF enunciou. Mas os defensores do termo *open source* 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”.

Ironicamente, o co-fundador da Open Source Initiative, junto com Eric Raymond, veio de um dos projetos de software mais bem-vistos pelo grupo *free*. 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 Foundation²⁸. Antes de ser uma contradição, esse fato é sinal de como as fronteiras entre os grupos políticos do software livre não são fixas. Embora existam as divisões, há também muitos valores em comum.

Cabe aqui uma pequena explicação sobre o que significa uma distribuição. Politicamente, elas são os mais importantes projetos de software livre, reunindo o maior número de colaboradores. Como o código do GNU/Linux é livre, ou seja, pode ser modificado e adaptado por qualquer um, esses códigos precisam ser agrupados em pacotes de software que obedeçam certos padrões, em sua forma executável, nas chamadas distribuições. Para se instalar um sistema livre completo e funcional com praticidade é preciso escolher alguma das 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 aos seus clientes. No entanto, existem também as chamadas distribuições da comunidade, grupos de usuários e programadores que empacotam os vários programas disponíveis com licenças livres de modo que formem um sistema completo, integrando o sistema operacional com diversas ferramentas de desenvolvimento, de escritório, jogos e outros. Exemplos de distribuições feitas por empresas são a Red Hat, a Novell/Suse e a Mandriva (empresa franco-brasileira fruto da fusão da brasileira Conectiva com a francesa Mandrake). Mas há também distribuições feitas por desenvolvedores independentes, como o Slackware e o Debian. Para receberem recursos e terem uma face institucional essas distribuições costumam organizar-se em fundações ou ONGs.

A distribuição Debian, cuja Definição Debian de Software Livre teve sua redação final feita por Perens, tem por princípio usar exclusivamente softwares considerados livres, alijando códigos com

²⁸ Em seus primeiros anos, o Debian foi financiado pela Free Software Foundation (Sanchez, 2007)

outras licenças²⁹. Ela tem, inclusive, o que chama de “contrato social”³⁰. A definição de *open source* usada pela Open Source Initiative foi emprestada da Definição Debian de Software Livre, inclusive com a mesma formulação, apenas sendo omitidas as referências ao Debian.

Porém, a definição de *open source* publicada pela OSI conta também, em cada item, com uma explicação, uma justificativa de sua existência, texto adicional que não existe na definição Debian. O exame do que foi adicionado ao texto original da definição Debian nos dá algumas pistas sobre as intervenções que o grupo *open* passa a fazer sobre quais são os novos valores a serem ressaltados pelo movimento software livre. 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 bazar de Linus Torvalds. A fundação estabelece um objetivo, uma razão para o item 3, sendo este permitir a continuidade do método de trabalho, baseado na revisão dos pares e no encaminhamento de soluções autônomas e de forma acelerada (“para que a rápida evolução se concretize”), sem a necessidade de autorização do autor anterior, que poderia frear ou retardar o processo.

Como mostra da ressignificação que está sendo operada pelo *open*, interessa comparar o

²⁹ Nos últimos anos, a Free Software Foundation tem colocado em questão alguns pedaços de software oferecidos pelo Debian, apontando-os como não-livres. Esse debate não deve ser considerado apenas do ponto de vista técnico-jurídico, mas também como uma disputa política.

³⁰ 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, consultado em 20/11/2008.

estabelecimento da mesma permissão na GPL, licença-modelo do grupo *free*. A possibilidade de alteração e distribuição da versão modificada já era algo permitido e incentivado, porém, com ênfase em outros fins que não a melhoria técnica. Não se trata, na GPL, de abdicar do controle, da autoria, da propriedade em nome do “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, acaba por abandonar a Open Source Initiative e retorna 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 deixa claro que *open source* e *free software* significam a mesma coisa, mas que a OSI não estaria enfatizado a importância da liberdade, o que considera um erro.

“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.”³¹

Perens certamente foi um dos sujeitos que mais tentou conciliar os ditos propósitos pragmáticos da OSI (em que se pode incluir tanto a expansão do uso de softwares livres e sua entrada forte no mercado tradicional de informática, quanto a melhoria mais acelerada da qualidade dos softwares) com

³¹ Mensagem trocada em lista de discussão e publicada em <http://lists.debian.org/debian-devel/1999/02/msg01641.html> O termo hacker é trabalhado em capítulo específico nesta tese.

a ideia de liberdade propagada pela FSF. Em 2001, logo após declarações do executivo da Microsoft, Craig Mundie³² – que criticou o que seria o caráter “viral” da GPL, o *copyleft*, a exigência de que todo software derivado de software licenciado pela GPL também seja GPL (se altero o software A, licenciado pela GPL, e produzo o software B, B também deve ser licenciado pela GPL) – Perens escreveu uma carta assinada conjuntamente por dez membros do movimento software livre, incluindo 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*, sinal da articulação política necessária. Na carta de Perens, há trechos com argumentação muito semelhante à desenvolvida por Stallman no texto “The GNU GPL and the American Way”³³. Segue um trecho da carta:

“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

³² 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. Disponível em <http://www.microsoft.com/presspass/exec/craig/05-03sharedsource.asp>.

³³ Diz Stallman, no texto: “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>, consultado em 20/11/2008

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, de Stallman e de Perens, a Microsoft é descrita como uma empresa que deseja “controlar”, seja o mercado, sejam os usuários. A empresa seria um empecilho ao livre fluxo do desenvolvimento tecnológico, um vetor de desaceleração. Ela força sua entrada e o controle do mercado ao “embrace”, ou seja, ao adotar padrões que outros já estão usando e “extend”, introduzir modificações reguladas com licença proprietárias, fechadas e secretas, que dificultam a adoção e a compatibilização dessas modificações por outros. Embora na carta de Perens 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. É a licença livre com efeito *copyleft* que garantiria que todo o esforço de melhoria do software, toda modificação introduzida e distribuída, seja entregue a todos. A GPL e o efeito *copyleft* não servem apenas ao propósito da FSF de manutenção das “liberdades”, mas também para garantir que todos os esforços acelerativos, todo desenvolvimento, esteja disponível a mais aceleração. A valorização da aceleração tecnológica é uma das ideias que unem os grupos *free* e *open*, embora haja diferenças de ênfase entre ambos.

Mas é principalmente na relação de oposição à Microsoft que, naquele momento, *open* e *free* encontram parte de suas afinidades. A empresa, pelo poder e lucros que acumulou, é a imagem perfeita da grande corporação monopolista originária do modo de comercialização pautado pelas licenças proprietárias e pelo capitalismo do século XX. Ao mesmo tempo, a grandeza da empresa é também símbolo daquilo que se tornou pesado e envelhecido, do passado a ser derrotado, a partir do qual se deve evoluir. Como vilã, a Microsoft oferece um contraponto fácil para qualquer corrente política do

software livre, que nela podem encontrar um bom conjunto de características negativas.

Chama a atenção também, na carta de Perens, a ordem das assinaturas, indício das relações de poder e prestígio. Em primeiro lugar, Perens, que 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, dos direitos autorais enrijecidos, do método catedral e da subordinação da aceleração aos interesses comerciais das empresa. Mais tarde, Raymond e alguns outros líderes do *open* vão criticar o efeito *copyleft* da GPL, que impediria uma melhor relação com as empresas, impedidas de se apropriarem do código livre. Porém, no momento da carta de Perens, mais importante é colocar-se contra a um ataque da Microsoft.

Richard Stallman diz não ver o subgrupo *open* 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.", diz. (Stallman, 2002: 55)

Inimigo ou parceiro eventual, 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, seus modos de ação e seu discurso político são tratados como caricatos e de maneira jocosa, principalmente quando obstaculizam a eventual colaboração de empresas capazes de investir na aceleração tecnológica e na adoção das ideias *open*. Com novas figuras proeminentes ocupando o cenário do movimento software livre, foi possível falar de abertura e do modelo desenvolvimento bazar proporcionado pelas licenças livres sem recorrer à figura incômoda de Stallman. Perens, na carta que marcou seu retorno à comunidade Debian, afirma que, pelo

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 *open source* 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.'", escreve (Stallman, 2002). 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." (Stallman, 2002).

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 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: tanto as falas demasiadamente políticas de Stallman incomodam a Raymond como a retórica empresarial 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*, procurou calar as as ideias do grupo de 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.”³⁴

À declaração de conteúdo forte de Raymond, que acabara de chamar de louco um dos dois

³⁴ “Let my software go!”. *Salon*. Disponível em http://archive.salon.com/21st/feature/1998/04/cov_14feature2.html. Acessado em 17/01/2010.

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 até em tom de brincadeira³⁵. 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 os derrotou. 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 diz procurar 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 ideia 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.”³⁶

Recusando-se a reconhecer que o movimento identifica-se com o espectro ideológico da

³⁵ 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>

³⁶ “Richard Stallman: "Software Livre não é pela direita nem pela esquerda”” http://webspace.webring.com/people/gu/um_6465/direita_esquerda.html, consultado em 12/11/2009.

esquerda, Stallman assume o sucesso do movimento *open source* nos EUA para mostrar, significando de maneira bastante tímida e conservadora, o que entende por direita e esquerda: “Eu não concordo com eles [direita e libertários], 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 Party³⁷, 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 campos do espectro igualmente repugnantes”³⁸.

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 ideias do *open source*. Diz ele, continuando o debate que teve com Stallman, publicado na *Salon.com*:

“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 ideias a que Raymond deu força com sua Open Source Initiative estão totalmente de acordo com sua visão. Ele diz defender os princípios *open source* pela eficiência que vê na prática,

³⁷ 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/>

³⁸ 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.”

pela qualidade do software gerado pela “seleção natural” que descreveu em *A Catedral e o Bazar*. O efeito “prático” tem mais relevância do que os princípios colocados. Assim como, ao assinar a carta de Perens, defendeu o caráter defensivo do efeito *copyleft* mesmo mostrando depois não concordar com ele, Raymond assume a negociação política necessária para angariar apoio à definição de *open source* da OSI. Continua, 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 um liberalismo de mercado evidente nas falas de Raymond: ele aceita que até mesmo o modelo *open* que defende deve provar sua força pela seleção do mercado. Ao contribuir decisivamente para a fundação do *open*, em processo que procurou ele próprio construir-se como figura pública, Raymond deixou claras suas convicções políticas, que ele diz serem calcadas num liberalismo clássico. É razoável supor que, nesse processo, Raymond tenha atraído não somente as empresas mas também ativistas e programadores com afinidade com sua visão política. Além disso, galvanizou uma determinada visão anti-tradicionalista e com olhos para um futuro de progresso tecnológico contínuo. A marca do discurso de Raymond não é a eliminação da desigualdade, a possibilidade que existe, no software livre, de que um usuário comum estude um código-fonte e possa interagir criativamente, participando do processo de criação de programas em relativa igualdade com qualquer outro esforço empresarial de produção. É a eficiência técnica, a velocidade de progressão de um método que

estabelece uma “seleção natural”, pelo qual o software “evolui”. Em textos do grupo *free*, por outro lado, dificilmente encontra-se alguma referência à distinção entre clientes ou usuários e programadores/desenvolvedores (Evangelista, 2005). 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?”³⁹

Algumas expressões merecem ser destacadas pois são as marcas desse discurso derivado da ideia de “seleção natural”. Está dito: “o software evolui/the software evolves”, como se estes fossem dotados de vida própria, e se os projetos puderem se desenvolver e competir entre si num ambiente de seleção natural (na Internet, disputando a atenção de milhares de programadores) haveria um progresso técnico, de qualidade. Também: “Nós na comunidade *open source* aprendemos que esse veloz processo evolucionário...”. E mais: “Há 20 anos esse *momentum* está sendo construído nas culturas técnicas que construíram a Internet...”. “Are you ready?”, pergunta o texto, como quem diz ameaçadoramente: evolua ou morra, o futuro é agora.

³⁹ O texto esteve na página inicial da OSI até 2007, sendo posteriormente, com a reforma do website, substituído por algo mais sucinto. Em 20 de novembro de 2008, o texto original ainda podia ser lido em um espelho do site original: <http://www.samurajdata.se/opensource/mirror/>

Elemento inerente ao processo evolutivo, a competição, por outro lado, é algo que, se acirrada, não é vista com bons olhos por Stallman. Ela é ruim quando retarda o movimento, quando serve ao propósito do lucro em lugar da aceleração, da melhora tecnológica. No Manifesto GNU, que escreveu ainda em 1985, antes da redação da GPL e como texto-convite aos desenvolvedores para produzirem software livre, 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.”⁴⁰

Enquanto para a OSI o mundo comercial é um aliado na construção de softwares *open source*, para Stallman seus objetivos lucrativos podem atrapalhar a iniciativa. O mercado é algo a ser controlado, regulado.

O elogio à velocidade

A valorização da velocidade e da aceleração é algo presente de uma maneira geral no movimento software livre, tanto no grupo *free*, cujo falante mais emblemático é Richard Stallman, quanto no grupo *open*, que teve como principal ideólogo de seus momentos iniciais Eric Raymond. A

⁴⁰ *Manifesto GNU* (1985). Extraído de: <http://www.gnu.org/gnu/manifesto.pt-br.html> Acessado em 04/08/2009.

fala acima de Stallman, produzida em 1985, antes de qualquer teorização mais clara sobre as virtudes do modelo bazar de desenvolvimento, dá conta de como acelerar, “correr mais rápido”, esteve entre os objetivos iniciais. O método para se acelerar, contudo, deveria ser a colaboração e não a competição desregulada por vezes presente no capitalismo. Nesse texto primordial, Stallman não se eximiu de apontar o que, para ele, era uma imperfeição do capitalismo desregulado.

Ao nomear e fazer seu elogio ao método bazar de desenvolvimento de software – tanto pelo livro *A Catedral e o Bazar* como pela criação de instituições que passaram a repetir seus argumentos – Raymond, porém, conseguiu deslocar novamente a argumentação em direção à validade da competição. Ela reaparece na metáfora do mundo natural, quando as fortificações (as licenças, a propriedade intelectual, a tarifa pela circulação), que impedem o livre fluxo dos códigos, tornam-se obstáculos à evolução, à aceleração do desenvolvimento. Muitos dos membros do grupo *open* (Raymond, inclusive) defendem atualmente modelos mais livres de licenciamento do que a GPL, semelhantes ao domínio público, afirmando que restrições como o efeito *copyleft* impedem uma maior adoção pelas empresas, que poderiam fazer o software evoluir ainda mais. Tanto a propriedade intelectual do software proprietário como direito autoral em sua forma “livre, mas com restrições colaborativas” obstaculizam. O primeiro porque exige tarifas para que a tecnologia circule, outro porque requer uma espécie de pedágio de reciprocidade, o compartilhamento da melhoria implementada de maneira que se torne não exclusivo.

Talvez caiba a comparação com as cidades de que fala Virílio em *Velocidade e Política*: “A burguesia extrairá seu poder inicial e suas características de classe menos do comércio e da indústria (que, como se sabe, não lhe eram específicos – conhece-se o papel crucial do monasticismo, da cavalaria, etc. no domínio dos bancos, das indústrias) do que desta implantação estratégica, estabelecendo o 'domicílio fixo' como valor (monetário, social) da especulação fundiária enquanto

venda e tráfico do imóvel (do imobiliário), deste direito de residir por trás das muralhas das cidades fortificadas: direito à segurança e à preservação em meio à perigosa migração de um mundo de peregrinos, compradores, soldados, exilados, deslocando-se aos milhões”. (1996; 24). Ao software proprietário interessa a venda da fortificação pura, das licenças; ao grupo *open* vale defender a GPL contra a fortificação proprietária, mas também sugerir modelos que possam levar a descontinuidades lucrativas no fluxo evolutivo, permitir que empresas tomem os códigos livres e lucrem com eles, sem necessariamente compartilhar as modificações. Como o objetivo final é a própria evolução tecnológica, o lucro das empresas pode ser interessante no sentido de ser meio para a arregimentação de trabalho tradicional, comprado no mercado, ou seja, mais emprego para técnicos especialistas em software livre.

Ao mesmo tempo, a ideia de aceleração, para o grupo *free*, permaneceu, pelo menos até bastante recentemente, imbricada, de forma subordinada, ao ideal de cooperação. Atrelada à defesa “liberdade do software”, ou seja, à permissão para que os sujeitos possam trocar colaborativamente códigos, ganhou força o objetivo de produzir um bem coletivo, softwares que possam ser utilizados por todos e para os quais toda contribuição, toda modificação, tenha ela sido feita por uma grande empresa ou por um simples aficionado, seja revertida a todos. Toda melhoria do software (evolução) deve ser direcionada à todos, o que também implica que nenhuma energia deve ser desperdiçada, nenhum esforço deve ser direcionado para fora do sistema de evolução acelerada.

Para se entender melhor a dinâmica acelerativa do sistema livre e proprietário talvez seja interessante retomar uma descrição de seus respectivos funcionamentos. O software proprietário é, oficialmente (salvo apropriações ilícitas de códigos livres), produzido completamente sob os auspícios e o planejamento de uma empresa. Os diferentes funcionários contratados ocupam-se da produção, escrita e integração dos códigos, que são de direito exclusivo do financiador da produção. A troca de

informações e códigos-fontes acontece de maneira controlada apenas entre pessoas autorizadas.

Dada as permissões instituídas pelas licenças livres, a dinâmica de produção não-proprietária acontece de maneira diferente. Em geral, o iniciador de um projeto coloca o código-fonte na Internet, tornando-o utilizável e modificável por qualquer um. Desenvolvedores interessados no projeto fazem suas alterações ou criando um novo projeto, com objetivos completamente diferentes, fazendo o chamado *fork*⁴¹; ou enviando sugestões e colaborações ao desenvolvedor inicial, que decide se as incorpora ao projeto ou as descarta.

O software proprietário utiliza tipicamente o modelo catedral, o primeiro exemplo; o software livre, o modelo bazar. No software proprietário, as licenças funcionam como fortificações, impedimentos jurídicos ao livre trânsito dos códigos. Elas servem ao propósito do lucro, são a maneira encontrada pelas empresas que se ocupam da comercialização de software de obterem retorno financeiro. É o que torna ilegal a transmissão (cópia) não autorizada do código, aquela não feita mediante pagamento do valor estipulado pelos detentores dos direitos. No software livre, todo o fluxo é permitido. E Raymond e o *open source* igualaram fluxo a evolução: foi descrito um processo em que a troca de códigos funciona como seleção natural. Postulou-se a ideia de que o fluxo em si – os milhares de olhos a inspecionar o código – é garantia de melhoria técnica e aceleração.

O *open source* deu relevância a uma nova prática de produção de software, que materialmente só se tornou possível em grande dimensão 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

⁴¹ *Forks* acontecem quando uma pessoa ou grupo decide dar um outro tipo de desenvolvimento ao software, às vezes constituindo nova comunidade em torno dele. Raymond (1998) afirma haver uma pressão social contrária aos forks, que ocorreriam apenas por grande necessidade. Os projetos não receberiam *fork* também em respeito e devido ao prestígio dos desenvolvedores originais. O que se verifica, contudo, é que muitos desses *forks* acabam acontecendo por razões políticas misturadas a justificativas de cunho prático.

melhorá-los. A relação mais flexível com a propriedade passou a ser justificada não pela crítica à privatização, monopólio do conhecimento e pela necessidade de uma regulação com princípios éticos, mas pela melhoria técnica, pela seleção natural estabelecida na Internet em que, quanto menos regras, melhor.

Na década de 1980, Stallman encontrou motivação para o movimento software livre quando práticas empresariais impediram-no de trocar código com seus colegas na universidade, quando viu sua prática cotidiana ser restringida por novas licenças de propriedade. Na época, o mercado de tecnologia da informação vivia um momento de transição, em que as empresas deixavam de oferecer o software gratuitamente, pré-instalado nos hardwares que vendiam. Surgia o mercado de software, baseado nos direitos autorais, tornando os programas de computador uma mercadoria à parte. Stallman tinha em mente resistir a esse processo, e buscou uma palavra forte na cultura estadunidense, representativa de direitos que ele afirmava estarem sendo violados, a liberdade de trocar informações – códigos – com seus colegas. David Harvey aponta como a palavra liberdade, esgrimada por um movimento político, representa a ameaça de cooptação pelo neoliberalismo. “Todo movimento político que considera sacrossantas as liberdades individuais corre o risco de ser incorporado às asas neoliberais” (2008, 50). Harvey fala especificamente dos anos 1970, época vivida intensamente por Stallman (Williams: 2002).

No começo dos anos 1970, quem buscava liberdades individuais e justiça social podia fazer causa comum diante do que muitos viam como um inimigo comum. Considerava-se que poderosas corporações aliadas a um Estado intervencionista dirigiam o mundo de maneiras individualmente opressivas e socialmente injustas. (...) Tomando os ideais de liberdade individual e virando-os contra as práticas intervencionistas e regulatórias do Estado, os interesses da classe capitalista podiam alimentar a esperança de proteger e mesmo restaurar a sua posição” (Harvey, 2008: 51-52).

Quase dez anos depois de dar nome ao movimento, Stallman usaria a palavra liberdade exatamente para afirmar seu pertencimento à cultura política dos EUA e afastá-lo mais uma vez das

acusações de ter inclinações comunistas. Em fevereiro de 2001, declarações do executivo da Microsoft, Jim Allchin, geraram grande repercussão. Allchin afirmou que o software livre ameaça a propriedade intelectual e disse 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.””⁴².

Em resposta a esse comentário, Stallman fala sobre as diferenças entre *free software* e *open source* – já que Allchin usou *open source* – e assume 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 é 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. O comitê ficou conhecido por elaborar uma lista de mais de trezentos profissionais da mídia, acusados de serem simpatizantes e propagandistas do comunismo. A eles foi negado trabalho pelas grandes empresas de comunicação.

“The Open Source Movement, which was launched in 1998, aims to develop powerful, reliable software and improved technology, by inviting the public to collaborate in software development. Many developers in that movement use the GNU GPL, and they are welcome to use it. But the ideas and logic of the GPL cannot be found in the Open Source Movement. They stem from the deeper goals and values of the Free Software Movement.

The Free Software Movement was founded in 1984, but its inspiration comes

⁴² “Open Source ‘stifles’ innovation”. Em http://www.theregister.co.uk/2001/02/16/open_source_stifles_innovation/. Acessado em 04/08/2009.

from the ideals of 1776: freedom, community, and voluntary cooperation. This is what leads to free enterprise, to free speech, and to free software. As in “free enterprise” and “free speech”, the “free” in “free software” refers to freedom, not price; specifically, it means that you have the freedom to study, change, and redistribute the software you use. These freedoms permit citizens to help themselves and help each other, and thus participate in a community. This contrasts with the more common proprietary software, which keeps users helpless and divided: the inner workings are secret, and you are prohibited from sharing the program with your neighbor. Powerful, reliable software and improved technology are useful byproducts of freedom, but the freedom to have a community is important in its own right.”⁴³

Ao afirmar que o movimento software livre representa sim os valores do *american way*, Stallman rediscute e ressignifica *american way*. Ao fazê-lo, procura dar à expressão um sentido coerente com os princípios do software livre, que estariam enunciados na licença GPL, ao mesmo tempo em que trata o software proprietário como algo que mantém seus “usuários indefesos e divididos”, oferecendo um certo sentido, por oposição, também ao software proprietário, como algo “não americano”. *Unamerican* seria a Microsoft, e não o software livre.

Os comentários de Allchin foram recebidos com surpresa por um dos membros mais ativos e articulador da fundação da OSI, o autor e editor de livros de informática Tim O'Reilly. Seu espanto parece ser justamente por Allchin ter usado o termo *open source* ao fazer as críticas, e não *free software*. Diz O'Reilly em artigo: “I was disappointed, because Allchin's comments ignored all of the reasoning behind the widespread change from the term “free software” to the term “open source.” (While there is a lot of overlap between the ideals of the free software movement and the open source movement, the two are not identical.)” O'Reilly, no entanto, afirma que mesmo Richard Stallman não seria contra a propriedade intelectual, ao contrário, a usaria para criar um tipo de propriedade que é oferecida como bem público, atividade que compara à caridade, que afirma não ser nada *un-*

⁴³ Disponível em <http://www.gnu.org/philosophy/gpl-american-way.html> Acessado em 17/01/2010.

*American*⁴⁴. Nesse mesmo texto, O'Rilley deixa claro que o que ele, Eric Raymond e outros fizeram foi fazer uma escolha pragmática, que permite maior inovação e sucesso econômico. Não se trataria de destruir a propriedade intelectual, mas de potencializar seus efeitos. O texto de O'Rielly é coerente com o conjunto das ideias do grupo *open* e reforça a imagem da Microsoft como empresa do passado ameaçada pelas novas tecnologias e processos de desenvolvimento acelerado do futuro. Esses processos levariam a mais inovação, à melhora técnica derivada da popularização do método bazar de desenvolvimento.

A chave para se compreender o sucesso do grupo *open*, principalmente se quisermos entendê-lo junto às empresas, talvez esteja em, ao lado de se perceber como esse grupo foi capaz de mobilizar de maneira mais clara argumentos em favor da evolução, perceber também a tensão entre lucro e aceleração. Enquanto para o grupo *free* é um imperativo moral e prático que toda melhoria do software seja revertida para todos, em que a aceleração é algo desejável, mas que deve estar subordinada a regras que evitem uma competição destrutiva, o grupo *open* trata os escapes de energia do sistema – as melhorias que se tornam privadas, não-livres, para serem melhor apropriadas lucrativamente – com maior permissividade, entendendo-as mesmo como indiretamente alimentadoras da produção, já que o lucro é um imperativo. No *open*, fala-se ostensivamente em evolução e melhoria técnica, porém permite-se que haja convivência entre o sistema livre e proprietário e permite-se mesmo que haja o uso de códigos livres em sistemas proprietários, tendo em vista o financiamento e a incorporação do modelo bazar no sistema produtivo predominante. Além disso, para o *open* é preciso calar os questionamentos com relação à propriedade, escamotear a política e canalizar os esforços para a velocidade da evolução, para a guerra na competição entre pessoas e entre códigos. O questionamento

⁴⁴ “A Response to Jim Allchin's Comments” Disponível em [http://www.oreillynet.com/manila/tim/stories/storyReader\\$167](http://www.oreillynet.com/manila/tim/stories/storyReader$167). Acessado em 12/11/2009.

do fundamento da propriedade dá lugar à guerra entre empresas: ao invés de objetivar um novo modelo de propriedade, o *open* coloca como primordial a derrocada da empresa lenta e envelhecida (Microsoft) pela moderna e ágil (as novas empresas baseadas na Internet, como na época a Sun Microsystems e hoje o Google). As grandes empresas, agora presentes, contribuintes e obtendo vantagens do sistema *open* de produção, reintroduzem as grandes marcas, o marketing tradicional, mas com roupagem moderna, aberta. Enquanto as distribuições não-comerciais pautam-se pela estabilidade do software, as produzidas por empresas privilegiam as novidades.

Cabe aqui um pequeno parêntese para tratar do caso de duas distribuições GNU/Linux envolvidas em crescente disputa, em que considero que a ideia de aceleração tem um papel relevante.

Como já dito, o Debian GNU/Linux é uma das distribuições mais antigas e tradicionais do software livre. É produzido a partir de uma estrutura não-comercial e altamente dependente de trabalho voluntário⁴⁵ e notabiliza-se pelo seu intrincado processo decisório interno, que procura conjugar meritocracia e democracia. Usa somente softwares considerados livres e boa parte de seus membros doa suas horas de trabalho por identificar-se com seus princípios políticos (Coleman, 2004; Sanchez, 2007). O Debian é uma distribuição reconhecida por sua estabilidade – ausência de erros, *bugs* – porém trabalha com um processo de integração de pacotes lento. Novas versões dos diferentes softwares que integram a distribuição demoram a serem incorporadas, por serem exaustivamente testadas.

Em outubro de 2004, Mark Shuttleworth lança o Ubuntu, anunciado como mais uma das diversas adaptações específicas baseadas no Debian⁴⁶. Diversos projetos de distribuições baseadas no

⁴⁵ Uma estimativa bruta sobre o custo de desenvolvimento do Debian 4.0 se dependente de trabalho regularmente remunerado aponta valores na casa dos US\$ 13 bilhões. “Impossible thing #1: Debian GNU/Linux”, disponível em <http://fsmsh.com/2771>. Acessado em 13/11/2009.

⁴⁶ Com relação às disputas entre free e open, lemos no website do Ubuntu: “Open source: Open source is a term coined in 1998 to remove the ambiguity in the English word ‘free’. The Open Source Initiative described open source software in the Open Source Definition. Open source continues to enjoy growing success and wide recognition; Ubuntu is happy to call itself open source. While some refer to free and open source as competing movements with different ends, we do not see free and open source software as either distinct or incompatible. Ubuntu proudly includes members who identify with both movements.” Disponível em <http://www.ubuntu.com/community/ubuntustory/philosophy>

Debian voltadas a públicos específicos já existiam, mas o Ubuntu contou com alguns diferenciais que o fariam tornar-se, em pouco tempo, a distribuição GNU/Linux mais adotada no mundo. Um deles foi a adoção de um agressivo marketing, que transformou o software em um produto com estética mais próxima a de outros sistemas operacionais do mercado, com uma atenção especial à embalagem e ao design do produto. Shuttleworth, além disso, investiu maciçamente na promoção, distribuindo CDs do produto sem cobrar por isso. O interessado no Ubuntu precisava apenas preencher uma ficha em um *website* com um endereço para entrega e em poucos dias, receberia em sua casa, de graça, de 5 a 50 CDs, conforme desejasse⁴⁷. Shuttleworth já era conhecido do mundo da informática por ser um dos milionários surgidos no boom das empresas de internet do anos 1990. Em 1999, ele havia vendido a empresa que fundou em 1995, especializada em certificação digital, por mais de US\$ 500 milhões. Calcula-se que seu investimento inicial no Ubuntu tenha sido de US\$ 10 milhões, feitos por meio da Canonical, empresa que fundou para vender serviços associados ao Ubuntu. Para escolher a lista desenvolvedores a contratar pela Canonical, Shuttleworth teria levado consigo, em viagem de turismo que fez à Antártica, seis meses de arquivos da lista de discussão dos desenvolvedores Debian⁴⁸. Pela lista, ele teria chegado ao primeiro grupo de funcionários de sua nascente empresa.

Porém, há ainda uma outra característica importante a ressaltar com relação ao Ubuntu. Diferentemente do Debian, o Ubuntu adota uma política de incorporação rápida de novos softwares. Seus desenvolvedores trabalham a partir da base Debian, e procuram introduzir nela versões mais atuais dos softwares. O Ubuntu lança atualizações do sistema operacional completo a cada seis meses. Contudo, a relação Debian-Ubuntu não é de mão dupla: pacotes Debian são facilmente incorporados no Ubuntu, mas as contribuições do Ubuntu são tecnicamente mais difíceis de serem incorporadas ao

⁴⁷ Atualmente o Ubuntu envia, por padrão, apenas um CD, também isento de qualquer custo, incluindo de entrega internacional. Pedidos de quantidades maiores estão sujeitos a aprovação.

⁴⁸ “Interview: Jeff Waugh”, *Linux Format*, n. 87, 2006.

Debian. Deve-se enfatizar, sem dúvida, a qualidade técnica conseguida pelos esforços da Canonical, já que o Ubuntu é um sistema bastante amigável, com pouquíssimos erros e de fácil manutenção pelo usuário, o que com certeza foi definitivo para seu sucesso. Mas quero apontar aqui a rápida incorporação de novidades também como um dos fatores, já que o Debian também é um sistema relativamente fácil de usar e de baixa manutenção. O que o distingue é a “evolução” mais lenta.

Do lado empresarial, o sistema do software livre tem sido entendido de maneira geral como um acelerador dentro de um ambiente competitivo. Presente no Fórum de Internacional de Software Livre de 2008, o executivo Luiz Fernando Maluf, da Sun Microsystems, deu entrevista à agência Reuters, que foi aproveitada pelo website Terra, um dos patrocinadores do evento. Intitulada “Software livre não é decisão ideológica, diz diretor da Sun”, a matéria mostra a expectativa com relação ao potencial do software livre como vetor para a aceleração tecnológica por parte das grandes empresas. O método bazar é associado à velocidade, inovação e a “sistemas abertos”, enquanto o oposto disso é ligado à imagem da Microsoft. E tudo não passaria de uma mudança “matemática”, de “modelo de negócio”. A lógica do próprio capitalismo e da competição levaria a essa transformação.

“Para a Sun Microsystems, uma das primeiras grandes companhias de tecnologia a apoiar a abertura dos códigos-fonte de software à comunidade de desenvolvedores, será muito difícil uma empresa de tecnologia sobreviver no modelo antigo de negócios, baseado em sistemas fechados e pagamento de royalties.

Luiz Fernando Maluf, diretor sênior de estratégias para governo da Sun nas Américas, afirma que "algumas pessoas acham que a opção pelos sistemas abertos é ideológica; estão completamente enganados: é um modelo de negócios, matemático".

"O que algumas pessoas não percebem é que existem dois modelos de negócios na área de tecnologia neste momento", afirmou à Reuters durante o 9º Fórum Internacional de Software Livre.

No caso do processo tradicional, baseado em registro de patentes, "a maior restrição é a velocidade de inovação", opinou.

O outro modelo envolve o que ele classifica como "economia de rede", onde todo o conhecimento é compartilhado em uma rede de pesquisadores para que uma empresa tenha acesso a inovações que sozinha não teria condições de

fazer.

Ele citou o caso da tecnologia Java, criada nas dependências da Sun e que conta hoje com algo como 30 milhões de desenvolvedores.

"Esse grupo gera inovação com uma velocidade enorme", afirmou Maluf. Além disso, por se tratar de um contingente tão grande, é possível envolver pessoas não tão especializadas, o que reduz o custo do desenvolvimento e acelera a chegada de cada novo produto ao mercado, explicou.

"Tempo de acesso ao mercado é algo vital em tempos de economia digital", afirmou o executivo à Reuters. No caso do sistema operacional Solaris, criado pela Sun, desde que ela decidiu abrir seus códigos-fonte para a comunidade, o ciclo de desenvolvimento caiu de seis meses para 37 dias.

"Os dois modelos são antagônicos na era da economia digital", reiterou. No caso do processo tradicional, ele afirma que a receita só dura o tempo do registro de propriedade intelectual.

Para ele, "vai ser muito difícil uma empresa de tecnologia sobreviver no modelo fechado". O reflexo pode ser visto, inclusive, na cotação das ações, acredita ele.

"Os acionistas costumam se basear em tendências", disse.

Ele citou o caso do Google como um exemplo da rapidez com que uma companhia pode se beneficiar da escolha pelo modelo aberto. "Quem era essa empresa três anos atrás?".

Questionado se, então, a Microsoft tinha sua sobrevivência em risco por conta da decisão de manter seus principais sistemas fechados, o executivo afirmou que não há alternativa.

"Duvido que a Microsoft mantenha a competitividade com o atual modelo", ressaltou.

Além da Sun, que começou a dar apoio aos softwares livres em 1981, empresas como IBM, Oracle e SAP hoje também dão suporte ao modelo aberto.⁴⁹

Embora apóie o *open source* há bastante tempo, só mais recentemente a Sun ofereceu alguns de seus principais softwares com licença livre. Parece ter encontrado uma maneira de fazer sem abdicar da possibilidade de lucrar. Assim, o software livre funcionaria como redução de custos, lugar onde a empresa obtém trabalho voluntário abundante. Além disso, o código livre seria um meio único para acelerar o desenvolvimento, até porque conta com um número de trabalhadores inimaginável para uma empresa.

⁴⁹ Disponível em <http://tecnologia.terra.com.br/interna/0,,OI2759209-EI11562,00.html> Acessado em 20/11/2008.

Conclusão

A aceleração deve ser entendida aqui mais como um ideal de crescente melhoria tecnológica do que uma prática de vida. Não se trata, no caso, de afirmar ou discutir se vivemos uma realidade acelerada, com uma percepção do tempo alterada, frenética e com uma consequente diminuição do espaço (Harvey, 1994). O ponto é entender a aceleração tecnológica como um valor bastante forte para a cultura tecno-científica onde nasce e constitui sua base o movimento software livre. Mesmo com as diferenças locais que se acumulam a partir da expansão global do movimento, em especial no Terceiro Mundo, a ideia de progresso e aceleração tecnológica permanecem como algo que, mesmo que não deva acontecer a qualquer custo, é algo desejável para a melhoria das condições de vida (Kanashiro e Evangelista, 2004). Aceleração na produção/evolução do software significa software de melhor qualidade e progresso tecno-científico.

Ao permitir a coexistência com o software proprietário, aproximar-se das empresas, enfatizar a evolução técnica e colocar, simbolicamente, como uma de suas metas a aceleração tecnológica, o grupo *open* tem se mostrado mais eficiente na tarefa de mobilizar mais trabalho e, em consequência, conseguir mais poder. Parece tratar-se tanto de oferecer uma melhor recompensa material aos trabalhadores recrutados como oferecer ideias que se encaixam melhor com a cultura política contemporânea. Aceleração, evolução técnica e a ideia de que a competição é a forma mais adequada para se extrair o melhor são conceitos caros à nossa sociedade atual. Ao mesmo tempo, a aproximação maior com as empresas cria condições objetivas e materiais para que haja mais desenvolvedores sendo remunerados para produzirem softwares livres, profissionalização que reduz a dependência de trabalho voluntário a ser desenvolvido nas horas vagas.

Em um ensaio intitulado “Nobody has to be vile”, Slavoj Žižek (2006) descreve o que ele chama de “liberal-comunistas”, que seriam os verdadeiros inimigos dos progressistas hoje. A partir da polarização inicial entre a Porto Alegre do Fórum Social Mundial e a Davos do Fórum Econômico Mundial, Žižek aponta o enfraquecimento da primeira e a migração de muitas de suas estrelas para a cidade suíça. Os maiores representantes dos liberais-comunistas seriam grandes empresas de tecnologia, como a IBM, Intel, Google, a Microsoft de Bill Gates e o especulador financeiro George Soros. O autor descreve um conjunto de valores desse grupo: dão valor a ser “smart”, dinâmicos e nômades se comparados à centralização burocrática; acreditam em diálogo e cooperação em lugar de uma autoridade central; em flexibilização em lugar da rotina; na cultura e no conhecimento em lugar da produção industrial, em interação espontânea e autopoiesis (auto-criação) em lugar de hierarquias fixas. Considerariam como conservadores e estariam em oposição não somente à direita autoritária, mas também ao que chamariam de velha esquerda e sua guerra contra o capitalismo.

O ícone desse “capitalismo sem fricção” seria, segundo Žižek, Bill Gates, cuja empresa seria comandada por *ex-hackers* “trabalhando por longas horas, aproveitando de bebidas grátis e em um ambiente verdejante”. Os liberais-comunistas, “geeks da contra-cultura que tomaram as grandes corporações”, procurariam mudar o mundo por meio da caridade e da ação prática. Avessos à retórica anti-imperialista, se engajariam em mostrar ação e não depender da ajuda do Estado. Em termos produtivos, tratar-se-ia não de produzir para o mercado mas em estimular formas de colaboração social.

Žižek cita os dez mandamentos do liberal-comunismo, que foram descritos pelo jornalista francês Olivier Malnuit para a revista *Technikart*:

1. You shall give everything away free (free access, no copyright); just charge for the additional services, which will make you rich.
2. You shall change the world, not just sell things.
3. You shall be sharing, aware of social responsibility.
4. You shall be creative: focus on design, new technologies and science.

5. You shall tell all: have no secrets, endorse and practise the cult of transparency and the free flow of information; all humanity should collaborate and interact.
6. You shall not work: have no fixed 9 to 5 job, but engage in smart, dynamic, flexible communication.
7. You shall return to school: engage in permanent education.
8. You shall act as an enzyme: work not only for the market, but trigger new forms of social collaboration.
9. You shall die poor: return your wealth to those who need it, since you have more than you can ever spend.
10. You shall be the state: companies should be in partnership with the state.

O maior incômodo de Žižek parece derivar das ações de caridade desses novos chefes do capitalismo global, cujas ações de impacto midiático obscurecem as desigualdades do sistema que lhes permitiu enriquecer. Porém, a maior virtude do texto está no que é apenas um esboço dos valores que ele chama de liberais-capitalistas. A imagem que ele projeta para a Microsoft de Gates se encaixa muito melhor nas empresas *open* como o Google, o lugar que considerável parte do movimento software livre elege como dos sonhos para trabalhar. Os dez mandamentos refletem bastante bem valores que funcionam para os dois grupos do software livre, mas que foram ressaltados com especial eloquência para o grupo *open*. Ao que tudo indica, diversas ideias do movimento software livre estão na fonte do que Žižek está chamando de liberal-comunismo. Mantem-se a ideia de acesso livre, central ao movimento, conjugada com a ideia de lucros ao se prestar serviços. Percebe-se também a valorização das novas tecnologias e da ciência, do fluxo livre de informações, do trabalho flexível e dinâmico. As novas empresas *open* agem como enzimas, para despertarem a colaboração social, buscando formas de lucrar com isso.

Veremos em seguida como o Fórum Internacional de Software Livre, maior evento da área e que acontece anualmente no Brasil, coloca em cena a disputa ideológica entre *free* e *open* e faz o choque e a síntese entre Davos e Porto Alegre. O evento nasce em sincronia com o Fórum Social

Mundial, em uma atmosfera de contestação do capitalismo de variadas intensidades. Na linha de frente da organização estão técnicos com passado sindical, identificados com movimentos de contestação do capitalismo e que veem nas ideias *free* uma bandeira similar, de enfrentamento daquele que era, no momento, o grande gigante da informática, a Microsoft. Ao longo do tempo, porém, o evento cresce, assim como se fortalece internacionalmente o *open* e as empresas que dão sustentação a essas ideias de abertura dos processos de produção em favor da aceleração tecnológica.

Cap. 3. Free e open do 9º Fórum Internacional de Software Livre

Neste capítulo e no seguinte, procuro fazer um relato etnográfico do Fórum Internacional de Software Livre (Fisl), tendo como ponto de partida a nona edição do evento, mas também relembro situações que acompanhei em anos anteriores. Nesta primeira parte, meu objetivo é mostrar como o Fisl, embora seja um evento com ampla agenda, abordando diversos assuntos, em grande parte de suas discussões e em seus corredores reproduz, de maneira aguda, o embate entre grupos *free* e *open*. Pelo relato, pretendo mostrar como a principal clivagem política do software livre no âmbito internacional, entre os grupos *free* e *open*, originada no final da década de 1990, permanece como referência essencial de uma disputa que dá termos para ao movimento. Embora haja diferenças internas dentro dos dois grupos, e apesar de essa divisão nem sempre estar referenciada em instituições, pode-se apontar a existência de duas concepções distintas sobre qual o objetivo e a razão de existência do software livre e de sua estrutura de produção de software de modo coletivo.

Interessa, também, perceber como certos fenômenos e tópicos de debate e participação política – como as reivindicações de outros movimentos sociais, o neoliberalismo, o papel do Estado, o mercado, etc. – são percebidos e influenciam o movimento software livre. É pela relação, estabelecida ou não, com certos movimentos sociais, certas empresas e com diferentes representantes políticos do Estado, que os grupos *free* e *open* também estabelecem e fazem ver suas diferenças como grupos, assim como as diferenças internas aos próprios grupos. Além das diferenças históricas e de concepções pré-existentes, é pela relação política estabelecida com Estado, empresas e movimentos sociais que novas

diferenças se produzem, algumas se tornam mais opacas e outras são enfatizadas e reforçadas. Ao estabelecer esse encontro, o Fisl coloca em cena as principais disputas políticas do movimento software livre brasileiro, ao mesmo tempo promovendo uma atualização de sua pauta e um reposicionamento de seus membros, por meio de novas alianças e distanciamentos.

Além disso, o Fisl permite que acompanhem a existência de um conjunto, até certo ponto homogêneo, de comportamentos, valores, prescrições e restrições que operam no movimento como um todo. Embora existam discordâncias que permitam afirmar a existência dos dois grupos distintos, ambos compartilham uma história e valores comuns, sentem-se parte de um mesmo movimento e perseguem uma agenda básica: impulsionar o uso de softwares livres. São essas proximidades que fazem com que, embora haja debates que se repetem ao longo da história Fisl, o mesmo tenha crescido ano a ano sem que se torne um encontro majoritário de apenas um grupo político. Registram-se, ao longo dos anos, momentos em que pequenos grupos afirmam publicamente a intenção de boicote ao Fisl, devido a acontecimentos em anos anteriores ou à presença anunciada de pessoas em instituições que desagradem a esse grupo. Porém, essa ausência eventual nunca se mostrou significativa a ponto de representar a retirada definitiva do *free* ou do *open* no evento e um consequente esvaziamento. Para isso, questões práticas – como o fato de o Fisl ter relevância tão grande para o movimento que se torne irresistível – e negociações políticas certamente pesam. Mas, além disso, minha hipótese é que, embora existam divergências, são os valores culturais comuns que levam à inexistência de um rompimento definitivo. Esses valores se tornam expressos nas normas, restrições, julgamentos e em uma compreensão compartilhada sobre o significado de tudo o que acontece nos dias do Fisl. Mesmo que sejam rejeitadas ou aceitas, com maior ou menor ênfase, as regras do Fisl – assim como os discursos – são compreendidos e aceitos como dotados de uma certa razão por todos os participantes do movimento.

No contexto brasileiro, a divisão entre *free* e *open* parece ser mais acentuada do que em eventos globais similares. Veremos como os militantes brasileiros mais identificados com o grupo *free* estabeleceram, em sua história, relações estreitas com outros movimentos sociais, o que contribuiu para que o movimento brasileiro adquirisse um perfil específico. Parte dos organizadores do Fisl procura estabelecer contato com outras causas sociais, especificamente aquelas que são atravessadas por questões tecnológicas, como a questão ambiental e da segurança alimentar, o que tem contribuído para que os próprios militantes vejam o software livre como um movimento social, que faz demandas ao Estado e apresenta propostas à sociedade. Nesse sentido, o Fisl teve um papel historicamente relevante, por servir também como instrumento de pressão política e de elaboração de políticas e contato com governos. Diferente de outros eventos mundiais, o Fisl é marcado pela grande presença de trabalhadores ligados ao Estado e pela participação notável, mesmo que em número proporcionalmente pequeno, de políticos em exercício de cargos públicos.

Como mencionado no capítulo anterior, é importante que parte considerável dos militantes do movimento software seja formada por profissionais com conhecimento técnico, capazes de aprimorar e criar softwares que sejam licenciados como livres. Além disso, o incremento no número de usuários dos softwares significa uma maior valorização do programa de computador no mercado e/ou maior força política para o grupo que o sustenta. Assim sendo, grandes empresas têm aumentado sua participação no Fisl e no movimento software livre de uma maneira geral, usando-os no recrutamento de profissionais. Este fato potencialmente altera o perfil do movimento, já que novos membros também veem a participação no software livre como oportunidade de melhoria profissional. A questão será melhor elaborada no capítulo seguinte. Contudo, este capítulo já traz elementos importantes para a discussão.

O Fisl

Realizado anualmente, o Fisl é o maior evento a reunir o movimento software livre brasileiro e, mundialmente, rivaliza em tamanho apenas com a LinuxWorld San Francisco⁵⁰. O Fisl é importante para o movimento brasileiro por ser um momento de encontro entre diversos indivíduos e grupos cuja ação, durante o ano, é tanto local quanto nacional, mas que raramente encontram-se presencialmente, comunicando-se eminente por meio da Internet, usando de listas de discussão por *e-mail*, *chats*, blogs, fóruns de discussão, sistema de comentários em sites de notícias sobre o tema, entre outros. Há outros eventos regionais durante o ano, que promovem o encontro de parte desses indivíduos, coletivos e instituições. O Fisl, porém, é o evento de maior porte e abrangência, tido por todos como o mais importante.

Dado o tamanho e a relevância do movimento software livre brasileiro, os debates e encontros que ocorrem no Fisl por vezes têm consequências que afetam o movimento globalmente. No Fisl, as principais lideranças nacionais se encontram, articulam atividades e comunicam-se com lideranças internacionais presentes, estabelecem alianças ou antagonismos e tomam contato com um grande contingente de pessoas, que podem dar base a novos grupos de pressão, assim como funcionam como termômetro para novos e antigos projetos e ideias.

Como já discutido nos capítulos anteriores, é importante ter em mente as variadas formas possíveis de apoio e alinhamento político no movimento software livre, assim como seus diferentes resultados. A divisão entre *free* e *open* não significa necessariamente a existência de militantes

⁵⁰ A partir de 2009, a LinuxWorld passa a ser chamada de OpenSourceWorld. Trata-se de uma conferência de negócios, realizada em diversos países do mundo, incluindo o Brasil. Registram-se comentários sobre uma recente queda do público, conjugada ou derivada de uma maior orientação do evento ao mercado corporativo. Ver http://www.linuxtoday.com/it_management/2009081700435NWEV A última edição estadunidense do evento, em São Francisco, Califórnia, em 2009, teria reunido duas mil pessoas. Ver http://www.eventsinamerica.com/events/opensource_world/ev4a54cff874309/

formalmente separados, atuando em instituições distintas, trata-se de um conjunto complexo de sujeitos, em permanente contato, cujo posicionamento político dentro do movimento é bastante nuançado e cujas filiações não são – não somente aos de fora como também aos próprios sujeitos – muitas vezes de fácil identificação. Um visitante ao Fisl que não conheça a história dessas correntes políticas, possivelmente não será capaz de relacionar certas atitudes e declarações que lá ocorrem e nem perceber as filiações aos grupos manifestadas pelos indivíduos. Essas filiações aparecem pelo meio do uso cotidiano e declarado de determinados softwares (em especial das distribuições); do emprego de certas palavras para referir-se ao movimento ou ao sistema operacional livre; no uso de certas imagens simbólicas (logotipos de empresas ou projetos, mascotes) em camisetas, adesivos ou como ilustração de sites; na referência jocosa ou elogiosa a determinados líderes do movimento; entre outros.

O resultado dessas filiações é complexo e vai além do apoio declarado a certos grupos. O movimento software livre não se resume a uma campanha pública em favor de licenças para programas de computador com regras mais flexíveis, dirigida ao Estado, às empresas e aos usuários de softwares. Entre suas atividades, e como meio para se alcançar sucesso nessa campanha, está a promoção dos softwares que se utilizam dessas licenças livres. Isso significa que indivíduos e grupos fazem campanha por softwares que pretendem ocupar espaço de mercado de programas de computador proprietários, produtos cujo licenciamento constitui a principal fonte de renda de diversas empresas. Ao mesmo tempo, empresas que procuram fazer dos softwares livres a base de seu negócio – ao oferecer serviços a eles agregados, por exemplo – de certa maneira entram em disputa por usuários e por eventuais colaboradores, que possam ajudá-la a desenvolver o produto com que lucram. Assim, entre os atores políticos que dividem espaço no movimento software livre, temos não somente militantes, que se colocam de acordo com suas diferentes concepções sobre o que é e para que serve o software livre, mas

também grandes corporações que buscam espaço comercial entre si para seus produtos e serviços. Ao engendrar, por sua natureza, a oposição a um determinado modelo de negócios para o mercado de software (a venda de licença de uso de programas de computadores), o software livre abre espaços a e é objeto de tentativa de instrumentalização de um determinado conjunto de empresas, diretamente interessadas na promoção de um modelo alternativo. O caminho aqui não é apontar algum tipo de “uso indevido” do movimento pelas empresas, mas reconhecer um processo dinâmico de influência cruzada.

Ao mesmo tempo, é preciso entender o público que prestigia o evento Fisl e que, de uma maneira geral, integra o movimento software livre, não somente como o militante de uma determinada causa, ou seja, a flexibilização do regime de propriedade dos softwares⁵¹. O movimento software livre deve ser entendido como lugar de socialização; aprendizado e inserção profissional; e construção de identidade política. Por envolver, diretamente, conhecimento técnico, empresas e produtos, o Fisl tem progressivamente se tornado espaço para o recrutamento de profissionais. Essa faceta do evento, como veremos, insere-se na disputa política que atravessa o movimento de uma maneira geral.

O surgimento do Fisl: entre movimentos sociais e partidos de esquerda

O Fórum Internacional de Software Livre (Fisl) é um evento que reúne, desde o ano 2000, grande parte do que se convencionou chamar de “comunidade software livre brasileira”. Dessa comunidade fazem parte uma gama complexa de indivíduos que qualificam a si mesmos principalmente de acordo com suas ocupações: desenvolvedores (que desenvolvem, modificam o

⁵¹ É evidente que há um conjunto complexo de expectativas dos militantes relacionadas a essa flexibilização do regime de propriedade. Alguns a entendem mesmo como uma quebra desse regime, outros a tratam como uma melhor adaptação ao modo como os códigos circulam pela internet. Ao optar por uma definição mais ampla, pretendendo incluir a totalidade dos grupos.

softwares), programadores (que oferecem instruções para que os softwares funcionem), usuários, funcionários de governo, políticos, estudantes de computação, jornalistas, ativistas sociais, empresários, etc. Ao longo do tempo, essa comunidade cresceu, superando principalmente o limite do conhecimento técnico, envolvendo cada vez mais usuários de nível intermediário e simpatizantes de algumas das ideias gerais do software livre. Concomitantemente, o movimento software livre também cresceu internamente, conquistando progressivamente a simpatia e/ou interesse de profissionais e estudiosos da computação.

Estive presente em todas as edições anuais do Fisl desde 2004, mas somente nona edição procurei fazer um acompanhamento mais sistemático, de caráter etnográfico. Nos anos anteriores, minha presença esteve ligada a meu trabalho como jornalista, como militante do movimento software livre, palestrante e pesquisador.

Ao longo dos anos, o evento consolidou uma determinada estrutura organizativa que mistura feira de negócios e exposições, congresso científico e fórum político de debates. Esse formato híbrido pode ser inicialmente explicado pela história do Fisl. Surgido entre funcionários públicos de tecnologia, ligados a sindicatos e movimentos de esquerda, o evento buscou sua base de público entre estudantes e profissionais da computação. Esses profissionais e estudantes convivem, geralmente, com empresas de todos os tamanhos, de onde retiram seu sustento (como empregados ou patrões) e que costumam estar presentes em eventos da área. Some-se a isso o fato de o Fisl ocorrer em Porto Alegre, no início do século XXI, período em que a cidade viveu uma grande efervescência política como local de realização das primeiras edições do Fórum Social Mundial (entre 2001 e 2003). O que em um primeiro momento pode parecer contraditório (a conjunção entre setores em certa medida anti-capitalistas e o espaço para as empresas), faz sentido dado o perfil dos setores mobilizados, refletindo conjuntamente o ambiente de eventos para estudantes, militantes políticos e empresários/trabalhadores. A persistência

desse formato híbrido ao longo dos anos, como veremos, pode ser entendida como resultado da continuidade de certos debates e divisões políticas, assim como pelo atendimento de demandas apresentadas pelos diversos públicos-alvo e de financiamento da estrutura material.

Tendo como pergunta principal de pesquisa a influência do movimento software livre no governo federal, principalmente após a posse de Luiz Inácio Lula da Silva, Aaron Shaw (2008) oferece uma visão interessante sobre alguns dos personagens que construíram as fundações do movimento software livre brasileiro e que participaram ativamente dos primeiros anos da organização do Fisl. Segundo Shaw, parte deles compartilhava uma história nos movimentos de esquerda do país e, quando o governo Lula atingiu o poder, levaram à frente um discurso radical, buscando politizar o papel do Estado desenvolvimentista em uma economia do conhecimento. Os membros do movimento software livre brasileiro possuiriam características únicas, se comparados a seus pares internacionais. A principal delas seria a orientação política, uma mistura de Neo-Marxismo com Socialismo.

Um dos indivíduos entrevistados por Shaw e que contribuem para que ele forme essa percepção sobre o movimento brasileiro é Mario Teza, bastante ativo na organização do Fisl até hoje. Teza, nascido em 1964, em Porto Alegre, aponta o início de sua identificação com a esquerda como tendo acontecido no final dos anos 1970, quando das greves que levaram à formação do Partido dos Trabalhadores. Logo quando inicia em seu primeiro emprego, na estatal Serpro (Serviço Federal de Processamento de Dados) da capital gaúcha, Teza entra para o sindicato e torna-se presidente da seção local da Federação Nacional dos Empregados em Empresas e Órgãos Públicos e Privados de Processamento de Dados, Serviços de Informática e Similares (Fenadados). Shaw prossegue escrevendo a história da relação de Teza com o software livre, relatando em particular a sua articulação com Marcelo Branco, um amigo de Porto Alegre e então diretor da estatal Companhia de Processamento de Dados do Rio Grande do Sul (Procergs), que resultou na criação do Fisl, além de

outros indivíduos com o mesmo perfil político e história de vida bastante semelhante: formação técnica em informática, mesma faixa etária, funcionários de empresas públicas e alguma relação com movimentos de esquerda e o PT. Nesse sentido, um depoimento de Teza⁵² colhido por Shaw é emblemático do significado que parte dos organizadores históricos do Fisl dão ao software livre, mostrando que, pelo menos para alguns eles, o software livre significava uma possível “transcendência do capitalismo” e um meio para superar as limitações naturais das lutas sindicais:

By 1989, the labor movement was in crisis – it's still in crisis! But let's put it this way, for some people, we weren't satisfied with the labor movement and beyond that with the democratization – the unions also entered into a system – a status quo, let's say. It didn't subvert the social order after the creation of democracy, and for many of the activists at that time this was not enough. We wanted to do more. And for many of us, software livre has enabled us to do more. We are able to take direct action, break paradigms. The labor movement is incapable of this – it raises salaries, but it's a whole corporatist thing, it's still very out of date. [The union] is a middle stage between the medieval guilds, the industrial revolution, and some other little bit of something modern – so-called modern – as well. In reality, it's very dated and it doesn't overcome capitalism. In as much as software livre, without perceiving it, begins to transcend, at least challenge capitalism, the ownership society, and intellectual property. ”

A partir de 1999, quando o PT chega ao governo do estado, Mario Teza, Marcelo Branco e Marcos Mazoni – então presidente da Procergs, Branco torna-se seu vice-presidente em 2000 –, fortificam ligações entre o PT, sindicatos de Porto Alegre, empresas estatais, movimentos sociais e setores interessados em informática, a partir de certas ideias do software livre. Em julho de 1999, Branco, Teza e o técnico da Procergs, Ronaldo Lages, organizam o primeiro encontro visando discutir o assunto software livre no auditório da empresa municipal. Fazem-se presentes por volta de 40 pessoas e o grupo passa a se chamar Projeto Software Livre – Rio Grande do Sul, denominação que será, nos anos seguintes, copiada por organizações de defesa do software livre no Brasil todo.

⁵² Essa declaração de Teza foi colhida em 2005 e confirma declarações com o mesmo tom colhidas por mim em anos anteriores.

Esse arranjo inicial contribuiu para dar ao software livre de Porto Alegre um perfil específico, ligado à esquerda. Já nessa época, os militantes porto-alegrenses procuram claramente aproximarem-se do grupo *free*, vendo nesse grupo, cujo representante mais saliente é Richard Stallman, maior afinidade de ideias. Ao que parece, essa aproximação com o *free* não era acompanhada com a mesma intensidade por outros grupos do resto do país.

Uma das iniciativas importantes no Brasil à época era a *Revista do Linux*, publicação editada pela empresa curitibana Conectiva, que comercializava, desde 1997⁵³, a primeira distribuição brasileira de software livre. Shaw cita a participação de Teza em entrevista concedida pelo então governador do Rio Grande do Sul, Olívio Dutra, para o quinto número da *Revista do Linux*, datada de maio de 2000⁵⁴. Nessa entrevista, é mencionado o planejamento para o que se tornaria a primeira edição do Fisl, onde Dutra foi recebido efusivamente pelo público. Na conversa de Dutra com a *Revista do Linux*, publicação patrocinada por uma empresa e não partidária de um posicionamento radical, já se percebe uma divergência sobre como Olívio e a revista chamam o sistema operacional livre: Olívio fala em GNU/Linux, enquanto a revista, nas perguntas, refere-se ao sistema como Linux, o que serve como marcador da distinção entre os grupos *free* e *open*. Em seu site pessoal, Teza mantém a transcrição de alguns depoimentos que deu relatando a história dos Fisl. Em um deles, ao comentar a participação de um profissional de Campinas no primeiro Fórum, ele toca explicitamente na questão do nome a usar para o sistema operacional, deixando claro como isso envolve um certo posicionamento. É a transcrição literal de uma fala, sendo mantidas as retificações que o sujeito faz ao perceber que disse algo impróprio.

Segundo: quem nos ajudou muito, por incrível que pareça, morava em

⁵³ Stulzer, Rodrigo. “Os primórdios do Conectiva Linux” em *ComCiência*. 2004. Disponível em <http://www.comciencia.br/200406/reportagens/18.shtml>

⁵⁴ “Governador do software livre”. *Revista do Linux*. Disponível em <http://augustocampos.net/revista-do-linux/005/index.html>

Campinas na época, o Eduardo Maçan. Então, como a gente debatia pela internet, ele tinha escrito um texto na Unicamp chamado... na época, ah! ele também chamava de gnu Linux de Linux, não chamava de gnu. O texto era “Linux na escola, no trabalho e em casa”. [...]. Bom, aí quando a gente fez o debate nesse evento a gente discutiu o seguinte: Nos 4 anos de governo o que podemos fazer. Resolvemos fazer um planejamento de como faríamos este projeto numa linha de tempo. Em julho, o que nós discutimos para vocês entenderem. O Linux, o Gnu Linux [corrige-se] explodiu no mundo, ele surgiu em 1991, deu um primeiro pique em 1992 e realmente a explosão foi provavelmente em 1994, fora do Brasil.”⁵⁵

Na entrevista para a *Revista do Linux*, Dutra, além de insistentemente repetir o nome GNU/Linux a toda menção a Linux feita pelo repórter, dá indícios de como a ligação de seu governo com o software livre advém de uma ideia de que, por meio dele, é possível enfrentar questões que vão além da liberdade dos usuários de software ou da qualidade do software produzido, como a inserção do país no mercado mundial de tecnologia, livrar-se da dependência de países estrangeiros e o acesso igualitário à tecnologia e às riquezas dela advindas.

Revista do Linux - Como foi que o senhor se envolveu com a questão do Linux? Qual a importância do projeto software livre para o Rio Grande do Sul?

Olívio Dutra - O meu envolvimento começou quando era deputado federal e atuava na Comissão de Ciência e Tecnologia da CUT. Tínhamos a preocupação de que a evolução científica e tecnológica proporcionasse melhorias na qualidade de vida para o conjunto da humanidade, em especial os excluídos, e não que servisse como mais um instrumento e acumulação de riquezas das elites.

RdL - [...] ...muitos países tiveram seus caixas dizimados por déficits monstruosos e o Brasil não foge à regra. Diante do empobrecimento dos Estados, como na América Latina, o Linux passou a ser uma alternativa possível de informatização do Estado. O senhor diria que o Linux é mera solução de emergência ou um solucionador de dependências de terceiros? Uma alternativa para a falta de recursos ou um caminho de independência tecnológica?

Dutra - O GNU/Linux é um dos sistemas que representa informatização de qualidade para o Estado, e não se deve confundir a implementação desse produto nas empresas públicas como uma solução temporal, advinda de uma crise financeira. Sabemos que a necessidade é a mãe da criatividade, mas esse software aberto tem uma história recheada de bons resultados, além do que os programas abertos, livres de fato, proporcionam acesso a métodos de uma

⁵⁵ Disponível em <http://wiki.softwarelivre.org/Pessoas/ComoOrganizamosOIForumInternacionalSoftwareLivre> . Acessado em 15/11/2009

elaboração tecnológica muito rica em experiência, possibilitando utilizarmos todo esse conhecimento a serviço do Estado e do cidadão, livrando-nos enfim da dependência tecnológica.

RdL - [...] O que muitos estranham é que até políticos como o senhor tenham se voltado para o assunto, e este é um fenômeno mundial, e que deixa a muitos perplexos. Porque o Linux hoje é assunto de Estado?

Dutra - Nosso governo tem uma identidade muito grande com esse tipo de projeto, [...] pelo GNU/Linux. Espero que muito em breve possamos encontrar soluções que viabilizem o acesso do cidadão aos microcomputadores também de forma gratuita, para que assim possamos ter uma sociedade em que seus participantes possam utilizar a tecnologia da informação em condições igualitárias.

RdL - Como o senhor vê este movimento mundial, de cunho solidário, como o Open Source (código aberto)? Acredita que ele trará quais benefícios à sociedade?

Dutra - Os benefícios são inúmeros, mas gosto sempre de citar que para nós o mais importante é podermos ter no Brasil o retorno à produção de software, mantendo no país a inteligência e o controle sobre a tecnologia da informação. Podemos, finalmente, ter um sistema operacional que respeite as realidades regionais, operando com base nas ideias das pessoas que com ele trabalham, permitindo que cada comunidade possa se manter protagonista da sua própria história na evolução e acumulação do conhecimento científico e tecnológico.

A fala de Dutra deixa clara a ligação com o grupo *free*, ao insistir no termo GNU/Linux e ao apontar que os “programas abertos” são também “livres, de fato”. Mas, além disso, há agregação de outras razões para a adoção dos softwares livres, como obstaculizar a “agregação de riqueza das elites”, o fim da dependência tecnológica por parte do Estado, um acesso igualitário à tecnologia (dada a gratuidade do software) e o desenvolvimento de soluções mais adequadas à realidade regional (devido à possibilidade de modificações no código). Trata-se de um conjunto original de argumentos, com influência do grupo *free*, mas também fruto de uma interpretação específica feita por movimentos sociais de esquerda, funcionários públicos e políticos que lidam com os problemas de países pobres.

Um exame das páginas publicadas na internet pelos organizadores do Fis1⁵⁶ confirma que a ideia

⁵⁶ Páginas que já não estão mais disponíveis regularmente, mas podem ser acessadas via serviços de armazenamento histórico da internet.

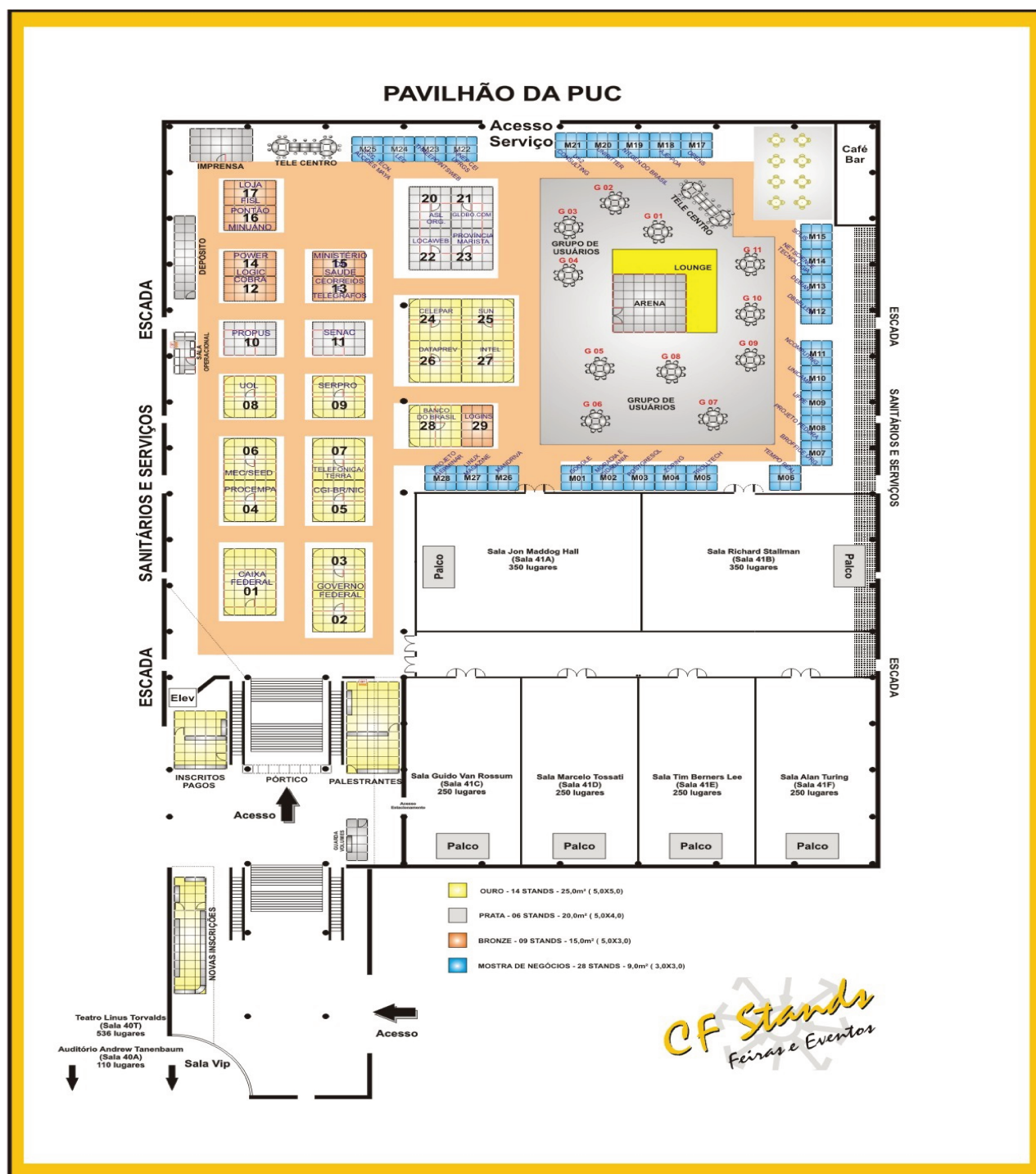
do software livre como fator de mudança social já estava presente nesses primeiros anos do evento. Tenta-se combinar o mundo dos negócios com objetivos de transformação da estrutura da economia. Uma das preocupações dos organizadores era impulsionar os negócios das empresas de software livre, vistas como portadoras, em si, de um modelo econômico alternativo. Uma das seções do antigo site do Projeto Software Livre-RS tinha o título “Negócios livres”. Nela, era possível encontrar o contato de empresas que trabalhassem com software livre em todo o país.

A importância de se estimular os negócios com as empresas de software livre é uma preocupação que persiste nas diversas edições do Fisl, estando ligada tanto à ideia de que isso levaria mais pessoas a “viverem de software livre” – “libertando-se” do “mundo do software proprietário” – como à noção de que não é saudável ao “ecossistema do software livre” estar excessivamente ligado a iniciativas estatais. Desde os primeiros anos, nota-se a importância da estrutura estatal para a promoção das ideias do software livre, exemplificada pela clara interconexão entre o Projeto Software Livre-RS, organizador do Fisl, com o governo do estado do Rio Grande do Sul. As páginas, tanto do Fisl em suas primeiras edições, como do PSL-RS, funcionavam em um domínio .rs.gov.br, ou seja, estavam endereçadas em um registro que pertence exclusivamente à administração estadual. Essa forte influência do governo estadual e municipal no evento foi substituída, mais tarde, quando da saída do PT do governo gaúcho e porto-alegrense, por forte influência do governo federal, a partir do governo Lula. Após 2003, o governo federal passou a contribuir mais consistentemente com o evento, oferecendo os patrocínios básicos que garantiram a realização do evento em condições mínimas. Além disso, funcionários públicos, ligados neste segundo momento ao governo federal, continuaram colaborando com a organização. Contudo, não se trata necessariamente dos mesmos indivíduos, e estes estão menos ligados à estrutura interna de organização do Fisl (até por não estarem no Rio Grande do Sul) do que os colaboradores iniciais. Em 2003, a organização do Fisl tornou-se autônoma do PSL-RS,

fundando uma ONG regularmente formalizada (a ASL.org) para gerir a organização do evento. Ao mesmo tempo, cresceu e diversificou-se o patrocínio oferecido pelas empresas privadas.

O Fisl em 2008

Em sua nona edição, o Fisl teve a seguinte estrutura física de distribuição espacial, bastante semelhante a, pelo menos, a dos quatro anos anteriores. Uma ala foi destinada a estandes de patrocinadores, bastante semelhante a de qualquer feira de exposições, com anúncio de produtos e distribuição de brindes. Nela misturam-se, como patrocinadores, ministérios federais, empresas públicas (federais, estaduais e municipais); pequenas, médias e grandes empresas privadas; e entidades, formalizadas ou não, que colaboram com o evento ou com o software livre. Estas, compartilham o espaço da “mostra de negócios” com empresas que adquiriram a menor cota de patrocínio. Envolvidos pela “mostra de negócios” e pelos patrocinadores principais (divididos nas categorias ouro, prata, bronze), localizam-se os “grupos de usuários”. Os “grupos de usuários” são coletivos que agrupam-se por motivos geográficos (grupos de estados distantes do Rio Grande do Sul ou países próximos como Uruguai e Argentina, que muitas vezes fretam um ônibus para viajarem ao evento); ou participarem de projetos em comum (usuários de uma determinada distribuição ou de certo software; ativistas envolvidos em certo projeto de inclusão digital ou de popularização de tecnologias livres; membros de projetos governamentais de inclusão digital). Envolvendo um dos lados desses *stands* ficaram as salas, de diferentes tamanhos, onde ocorrem as palestras.



Há duas categorias distintas de palestras, as propostas pelo público e as propostas pela organização. Contudo, todas são apresentadas nas mesmas salas, não havendo nenhuma distinção entre as sessões “oficiais” e as do público. As palestras propostas pela organização em geral envolvem

palestrantes internacionais convidados ou autoridades públicas, brasileiras ou não. As propostas pelo público envolvem uma gama ampla de indivíduos envolvidos de alguma forma com o software livre: desenvolvedores, usuários entusiastas, profissionais de empresas, acadêmicos, jornalistas, educadores etc. Meses antes do evento, o palestrante apresenta o resumo de uma proposta. Não há nenhum pré-requisito de formação técnica ou escolar feito ao candidato a palestrante para isso, embora seja um dos itens avaliados. A proposta deve encaixar-se em alguma das trilhas definidas pelo Temário, um grupo de trabalho da organização responsável pelas palestras. As trilhas de 2008 listadas na programação foram: Negócios (Produtos/Serviços), Tópicos Emergentes; Desenvolvimento: PHP; Desenvolvimento: Python; Desenvolvimento: Ruby; Governo e Software Público; Hardware e Sistemas Embarcados; Kernel; Admin; Ecossistema do Software Livre; Educação e Inclusão Digital; Desenvolvimento: Ferramenta/Metodologia; Desenvolvimento: Java; Desenvolvimento: Perl; Jogos e Multimídia; Desktop; Casos/Soluções; Desenvolvimento: Banco de Dados; Eventos Comunitários; Javali; ApyB; Fórum KDE; Organização; WSL; Hora Ginga. Dessas trilhas, nem todas estavam disponíveis para inscrição. A trilha Organização, por exemplo, serve apenas para a sessão de abertura e encerramento. Já a trilha Hora Ginga abarcou apenas sessões sobre um dos softwares que compõe o sistema de TV digital brasileiro e foi proposta por membros do governo federal, patrocinadores do evento. Nas trilhas regulares, os trabalhos a serem apresentados são selecionados pelo público, em um sistema em que é possível a qualquer um se cadastrar como avaliador. Nos fóruns anteriores, os avaliadores eram convidados pela organização, sendo recrutados principalmente entre os palestrantes dos anos anteriores. Após insistentes críticas ao longo dos anos, e de questionamentos duros sobre as razões que levavam à exclusão ou escolha de determinadas palestras, optou-se por abrir o processo de avaliação a qualquer interessado, não sendo necessário convite prévio.

Outra mudança sensível aconteceu nas trilhas: o número foi bastante aumentado, com a abertura

de espaço para palestras mais técnicas (as trilhas do ano anterior foram: Desenvolvimento e Banco de Dados Web; Admin; Comunidade e Filosofia; Ecossistema do Software Livre; Educação e Inclusão Digital; Gênero; Jogos e Multimídia; Desktop; Casos; Eventos Comunitários; WSL - International Track; WSL - Trilha Nacional; WSL - Software Livre na Universidade; Javali; Oficinas; Organização). No sistema *web*, que permite a visualização das palestras do evento, é possível selecionar um modo de visualização em que apenas as trilhas selecionadas são exibidas. Também é possível, pelo sistema, selecionar a visualização de palestras “técnicas” e “não-técnicas”. Longe de ser uma divisão fortuita, veremos que isso reflete (ou é reflexo de) um comportamento de parte do próprio público. Algumas pessoas circulam especificamente pelas palestras técnicas, evitando as de conteúdo “filosófico” e usando o evento como um congresso técnico/educativo. Outros procuram exatamente os assuntos não-técnicos, em que estão incluídas as discussões sobre direito autoral, patentes, gênero, inclusão digital, uso de software livre na esfera governamental e empresarial e história do software livre. O termo nativo mais frequente para esses assuntos é “filosofia”, palavra que originalmente expressa as motivações para o uso e a construção dos sistemas livres. A palavra é utilizada não somente em português, mas tem origem nas primeiras publicações da Free Software Foundation sobre o assunto. Parte do público do Fisl também se refere a essas discussões como “políticas”, em uma classificação que às vezes é – noutras não – pejorativa.

Mas, em comparação ao evento do ano anterior, o de 2008 teve uma mudança mais significativa. Ou melhor, um retorno significativo. Desde sua terceira edição, o Fisl é realizado no Centro de Eventos da PUCRS. Porém, a partir da sétima edição, foi transferido para o Centro de Eventos da Federação das Indústrias do Rio Grande do Sul, um espaço para feiras empresariais, porém mais distante do centro da cidade. A mudança teria ocorrido devido a um aumento da quantia cobrada pela PUCRS. De imediato, alguns setores – principalmente aqueles mais avessos ao contato com as

empresas – demonstraram insatisfação, enquanto outros animaram-se, pois viram uma oportunidade de aumentar o contato entre as empresas e o software livre. Realizar o Fisl em um espaço empresarial seria uma sinalização do quanto o software livre é amigável aos negócios. Ao mesmo tempo, por ser mais distante do centro da cidade e por possuir uma pior infra-estrutura de serviços (principalmente alimentação e transporte), a mudança despertou também críticas práticas.

Em 2008, após negociações com a PUCRS, o evento retornou ao prédio da universidade. A importância do retorno vai além da saída de um espaço empresarial. Em suas duas primeiras edições, em 2000 e 2001, o Fisl, ainda um evento de porte médio, reunindo pouco mais de duas mil pessoas, aconteceu no Salão de Atos da Universidade Federal do Rio Grande do Sul (UFRGS), mudando-se para a PUCRS apenas em 2002, quando cresce consistentemente. Em 2001 e 2002, contudo, aconteceram as duas primeiras edições do Fórum Social Mundial, utilizando esse mesmo espaço da PUCRS. A ligação entre os dois fóruns passa pela presença de alguns membros como organizadores dos dois eventos, pela semelhança dos nomes e pelo apoio do governo municipal e estadual a ambos. Mudando-se para a PUCRS, o Fisl acabou significando, para muitos, uma extensão dos debates do FSM sobre tecnologia. Além do mais, ambos os fóruns colocam-se de modo claro em oposição a uma estrutura maior, mais poderosa e já estabelecida (o neoliberalismo ou o software proprietário/Microsoft) e a favor da construção de “alternativas”, sejam elas para o sistema sócio-econômico ou para o sistema operacional dos computadores.

O advogado Tim Ney, da Free Software Foundation, esteve presente já na primeira edição do FSM. No ano seguinte, Richard Stallman esteve em uma mesa que discutiu tecnologia e comunicação, com grande audiência e repercussão. Deve-se dizer, contudo, que essa ligação simbólica entre os dois eventos é algo que já foi mais forte no passado e que, hoje, é algo presente apenas para alguns setores do movimento e para parte dos organizadores (aqueles com mais afinidades com os movimentos

sociais). Para outros, mesmo organizadores, a volta à PUCRS é apontada como benéfica apenas por razões práticas. Vejamos, como exemplo, o comentário de um dos membros do temário sobre o assunto, publicado em seu blog, intitulado *Mundo Open Source*⁵⁷:

“A volta para a PUCRS: a volta para a PUCRS para mim foi uma das melhores coisas do fisl 9.0. A PUCRS, além de ser melhor localizada, possui um acesso muito mais simples com muitas linhas de ônibus e lotações que passam por ali o dia inteiro. Além disso, a PUCRS tem uma variedade muito grande de restaurantes e lanchonetes que agradam os gostos e bolsos de todos os participantes do evento. Ponto para a ASL!!!”
(<http://mundoopensource.blogspot.com/2008/04/fisl9-ano-que-vem-tem-mais.html>)

A escolha do nome para o blog (*Mundo Open Source*) feita pelo autor, é forte indício de que o mesmo não tem forte ligação com grupo *free*. No entanto, ele se mostra feliz com a mudança a localização, facilidade de transporte e estrutura para alimentação.

Hackers, políticos e o público

Para melhor descrever o público do evento, vou dividi-lo em quatro categorias. Essa não é uma divisão nativa – embora use em parte seus termos – nem tampouco implica em posicionar rigidamente os indivíduos nessas categorias. O objetivo é apenas oferecer um referencial sobre a origem e a motivação dos presentes.

Burocratas: São os funcionários dos governos (municipal, estadual ou federal) ou de empresas públicas. Profissionalmente, realizam funções técnicas e/ou administrativas. Apenas uma pequena parte está envolvida diretamente no desenvolvimento de software. Em sua maioria são gerentes ou

⁵⁷ “fisl9 - Ano que vem tem mais...” Disponível em <http://mundoopensource.blogspot.com/2008/04/fisl9-ano-que-vem-tem-mais.html> Acessado em 04/08/2009.

administradores de sistemas. Parte está envolvida com programas de inclusão digital. Normalmente estão no evento com todas as despesas pagas pelos seus empregadores, o que implica ficarem parte do tempo no estande de quem os emprega. Estão presentes mais nas sessões que discutem políticas de adoção de software livre em âmbito governamental e nos debates sobre a filosofia do software livre, embora não rejeitem as sessões técnicas. Quando necessário, usam terno ou roupa social, mas preferem vestir jeans e camiseta. Têm entre 25 e 50 anos.

Empresários: São donos ou funcionários de pequenas e médias empresas. Frequentam quase que exclusivamente as sessões técnicas, embora também tenham interesse em mesas que debatam políticas governamentais – onde buscam espaço para futuras prestações de serviço ou apresentam aos burocratas demandas de suas empresas. Têm bastante conhecimento técnico e estão no evento ou com recursos próprios ou de seus patrões. Usam terno ou roupa social. Têm entre 20 e 45 anos.

Ativistas: Em geral têm pouco conhecimento técnico mas, se o têm, são autodidatas. Parte tem formação técnica de nível médio e universitária em ciências humanas. Estão ligados a projetos de inclusão digital ou que envolvam arte (música, artes gráficas) em software livre. Usam bermuda e camiseta, também com motivos políticos. Frequentam as sessões que discutem a filosofia do software livre, novas regras de propriedade intelectual, inclusão digital e política de governo. Estão no evento com recursos próprios, hospedados na casa de amigos e tendo viajado de ônibus. Parte têm ou já teve envolvimento com o movimento estudantil. Têm entre 18 e 30 anos.

Nerds: São, em geral, estudantes de computação. Frequentam principalmente as sessões técnicas. Aceitam as mesas sobre a filosofia do movimento, embora tenham uma visão bastante estrita sobre o tema. Vestem bermuda e camiseta, em geral com referência a personagens da cultura pop, piadas envolvendo conhecimento técnico ou projetos de software livre. Estão no evento com recursos

próprios, e muitos vêm em caravanas de diferentes estados. Viajam e andam pelo Fisl em grupo. Estão interessados em aprender sobre tecnologia e em contatos profissionais. Têm entre 18 e 25 anos.

Essas quatro categorias, grosso modo, podem ser posicionadas em relação aos grupos *free* e *open*. Não significam correspondência direta verificável necessariamente em casos individuais, mas permitem entender melhor a divisão geral. Nerds e empresários costumam manifestar maior rejeição à presença de políticos e partidos no Fisl e não fazem grande esforço em ligarem o software livre a outras lutas sociais. Ao contrário, os nerds frequentemente manifestam sua rejeição aos políticos, enquanto os empresários, embora tenham contato profissional com os políticos, procuram manifestar-se como apartidários. Já os ativistas e os burocratas, ou envolvem-se diretamente em outras lutas sociais ou não manifestam rejeição à interconexão delas com o software livre. Ambos têm também rejeição mais fraca à presença de políticos no evento.

Muitas vezes essa divisão burocratas/ativistas versus nerds/empresários aparecerá mascarada na subdivisão entre um público mais ou menos técnico, embora esse conhecimento mais avançado não seja um fato verificável. Pessoas com maior ou menor conhecimento técnico se espalham por todas as categorias e, além disso, o que parece existir mais concretamente é a preferência por determinados softwares ou linguagens de computador de acordo com os grupos⁵⁸.

Dentro da própria estrutura organizadora do evento essa divisão é operada na classificação informal dos membros entre “hackers” e “políticos”. De acordo com um informante, nessa divisão a qualificação de maior prestígio é “hacker”, assim sendo chamados aqueles que, para o grupo, teriam conhecimentos mais técnicos. Porém, o que se verifica é que, mais do que conhecimento, é necessário um determinado posicionamento público e político para se merecer esse adjetivo de prestígio na

⁵⁸ A linguagem Java, por exemplo, criada pela empresa Sun Microsystems, é bastante usada pelos nerds, além de ser a especialidade do representante da OSI no Brasil. Já o Twiki, software para construção de páginas web colaborativas, é largamente utilizado por membros do governo federal e por militantes do Projeto Software Livre Bahia, bastante identificado com outras causas político-sociais.

estrutura da organização. Os “hackers” têm uma postura pública austera, até mesmo reservada e, quando participam de um debate público – que quase sempre acontece por meio de e-mails em de listas de discussão – esse debate costuma envolver a discussão de características técnicas de determinados softwares. O trabalho profissional do “hacker” (de onde retira seu sustento) quase nunca envolve diretamente governos e sua relação com ocupantes de cargos oficiais (deputados, vereadores, etc.) é distante. Já os “políticos” da organização do Fisl são os que conversam e convidam as autoridades presentes no evento. Articulam o apoio financeiro e ocupam mais fortemente o papel de porta-vozes do Fisl e do próprio movimento⁵⁹. Por isso, os “políticos” são constantemente criticados, em especial pelos nerds – essas categorias são de uso geral, não restringem à organização do Fisl –, que apontam uma frequente contradição entre falar e fazer. Os “políticos” são acusados de falarem muito mas produzirem pouco, pois nunca estão envolvidos no “codar”, em escreverem software e participarem de grupos de desenvolvimento de programas. O “hacker” é uma categoria hierarquicamente mais elevada que o “político”, que é visto sempre com maior desconfiança (por eventualmente querer “se aproveitar do software livre para outras causas”). Os “políticos” efetivamente trabalham muito mais na organização (conseguindo apoios, negociando com o movimento, conversando com a imprensa), mas os “hackers” são figuras mais respeitadas pela comunidade. Produzir código e ter conhecimento de programação são fatores muito importantes para se obter prestígio dentro do movimento de uma maneira geral. Contudo, não é possível fazer uma relação automática e progressiva (mais unidades de conhecimento não significam mais unidade de prestígio), trata-se de algo também mediado por uma atitude pública de distanciamento ou de relação fria com a política partidária tradicional. Mario Teza e Marcelo Branco,

⁵⁹ Murillo (2009) utiliza os termos “téc” e “ativistas” como referência a “hackers” e “políticos” do modo como trato aqui. Encontrei o uso desses termos em conversas com membros da organização do Fisl, porém, o “téc” estaria em um nível hierarquicamente inferior ao “hacker”. O “téc” seria alguém mais jovem, muitas vezes – mas não necessariamente – com menor conhecimento técnico e mais ativo nos trabalhos gerais da organização do evento. “Ativistas” parece-me ser uma versão mais atenuada de “políticos”, porém não encontrei seu uso em específico.

por exemplo, ambos considerados “políticos”, nos oferecem bons exemplos sobre o funcionamento da reputação dentro da comunidade. Ambos possuem conhecimento técnico aparentemente equivalente e marcaram suas trajetórias pelo envolvimento com movimentos sociais e pelos primeiros esforços de organização do Fisl. Contudo, Branco é muito mais criticado por setores do movimento, ao que tudo indica por seu estilo pessoal. Está sempre disponível para entrevistas e costuma dar declarações fortes. Já Teza, embora também assuma um papel proeminente e dê declarações consideradas politizadas, porta-se de maneira mais discreta e procura ser mais um articulador interno, agindo de maneira mais pragmática. Com isso, suas ligações à esquerda – que são, de um certo ponto de vista, até mais fortes que as de Branco – acabam sendo melhor aceitas.

Para o movimento software livre, a categoria “hacker” é algo essencial (discuto o termo, seu sentido para o movimento de forma ampla e suas implicações, em capítulo separado) e congrega qualidades como criatividade, curiosidade, extrair prazer no trabalho e conhecimento técnico. É a distinção máxima que alguém pode receber dentro de um movimento que se considera “de hackers”. Ser hacker é parte da identidade do movimento software livre, é algo que se refere não somente a pessoas mas a uma atitude com relação à vida e ao mundo. Fora da estrutura contrastiva da organização, no software livre brasileiro de uma maneira geral, os ditos “políticos” do Fisl podem ser vistos e se declararem “hackers” – embora, ao se auto-identificarem, sejam recebidos internamente com certo ceticismo e ironia. Mas na estrutura da organização e do movimento eles são vistos como “políticos”.

Nas imagens, as filiações

Durante o Fisl, nos corredores e palestras, toda a carga política dessa disputa, em que *free* e *open* são categorias fundamentais, está presente na grande quantidade de símbolos de projetos (de softwares, de empresas, de iniciativas comunitárias, de iniciativas pela inclusão digital) que circulam em camisetas, cartazes, estandes, etc. Meu objetivo aqui não é discutir cada um desses símbolos exaustivamente, mas mostrar como eles estão inseridos em um sistema de significados que aponta, entre outros, para filiações políticas. O logotipo de um projeto de software não tem apenas um sentido imediato, mas está ligado à história política daquele projeto dentro do movimento. E usar o símbolo de uma distribuição ou software, e não de outros, em geral diz algo sobre o posicionamento político de quem o faz. Ao mesmo tempo, a distribuição dos símbolos guarda certa coerência e alguns não podem ser misturados.

O vestuário é algo importante, e pode marcar desde o simples pertencimento ao movimento como a filiação a determinado grupo. Durante meu trajeto ao evento, por exemplo, enquanto esperava o voo, já pude perceber participantes do Fisl apenas pelo vestuário e mesmo destino. Um deles vestia uma camiseta com a frase: "Existem apenas 10 tipos de pessoas no mundo". Trata-se do início de uma piada cuja formulação completa é "Existem apenas 10 tipos de pessoas no mundo: as que entendem códigos binários e as que não entendem".



A piada só faz sentido para aqueles que sabem que 10 (um e zero) significa dois em código binário. Esse tipo de humor, como já dito, é bastante frequente na comunidade. Contudo, não parece ser usado igualmente por todos, mas sim por aqueles mais identificados com os setores técnicos. Já as camisetas com símbolos e mensagens especiais são o item de vestuário mais visto pelos corredores do evento. Usar determinada camiseta significa marcar-se como: usuário de uma determinada distribuição (Debian, Red Hat, Ubuntu, Slackware etc); apoiador de certa entidade ligada ao software livre (um dos diversos Projeto Software Livre do Brasil, por exemplo); frequentador de certo evento (Congresso Internacional de Software Livre, que se realiza em São Paulo; ou o Encontro de Software Livre da Paraíba). Todas têm, em maior ou menor grau, algum significado político, de adesão ou proximidade a

certo grupo, com posições razoavelmente determinadas sobre que licença de software é mais ética, o valor ou o prejuízo de uma maior aproximação com as empresas, quem são os vilões e quem são os mocinhos no mundo do software.

Dois websites, de projetos diferentes ligados ao software livre, nos servem para ilustrar essa distribuição política dos símbolos.

O primeiro é da Rede Três Mosqueteiros Cooperativas de Software Livre, site wiki (sistema de publicação em que qualquer usuário cadastrado pode fazer alterações na página. Essas alterações são registradas e são recuperáveis) construído para integrar cooperativas de software livre. Iniciado em 2006, ano em que o Fisl abrigou mesa de debates com Paul Singer que discutiu o tema “software livre e economia solidária”, a Rede foi formada para a integração e troca de experiências entre as cooperativas. O uso de tecnologia twiki (tipo específico de wiki) para a construção do site e a imagem que segue abaixo são indicativos da participação líder da Colivre, cooperativa baiana bastante identificada com uma visão mais “política” do software livre, na iniciativa.



Os símbolos presentes na imagem marcam de forma consistente certas filiações. O pinguim, símbolo do Linux, é o símbolo máximo e mais popular do software livre. Embora Linus Torvalds, seu

criador, tenha divergências com Richard Stallman, o reconhecimento do caráter inovador do processo de produção descentralizada que Torvalds utilizou para o Linux é bastante geral. A imagem do pinguim é o símbolo mais constante em qualquer iniciativa de software livre e funciona como identificador básico. É interessante apontar que um desses pinguins usa uma camiseta com o desenho de um gnu, representando o projeto de Richard Stallman. Como vimos, Stallman insiste sempre para que o sistema operacional seja chamado de GNU/Linux. Dois outros projetos de software estão ainda representados.

Na asa esquerda de um dos pinguins, nota-se uma espiral vermelha, símbolo do Debian. O Debian é uma distribuição produzida pela comunidade – e não por uma empresa – e que se notabilizou por seu processo bastante horizontalizado e sistematizado de produção, além de sua aderência bastante estrita na incorporação de somente softwares livres. O Debian possui um contrato social⁶⁰, uma definição própria de software livre, e uma constituição, que define a estrutura organizacional do projeto e o processo de tomada de decisão. Como discutido em capítulo anterior, o Debian tem sido a distribuição preferida por militantes sociais que fazem uso de software livre⁶¹.

Ao fundo, um dos pinguins segura uma esfera contendo a imagem de uma pegada. É o símbolo do Gnome, uma das interfaces gráficas mais utilizadas nos sistemas livres. Diferentemente do que é usual em sistemas como o Windows, no sistema livre é possível utilizar um pequeno conjunto interfaces gráficas, que fazem a comunicação com o centro do sistema operacional (transformando a interação com a imagem em comandos invisíveis ao usuário). Essas interfaces são desenvolvidas por grupos diferentes de desenvolvedores, em projetos distintos. Tão popular quanto o Gnome é o KDE, com o qual existe uma certa rivalidade entre os usuários. Quando foi lançado, em 1996, o KDE fazia

⁶⁰ Os cinco itens do contrato social Debian são: “O Debian permanecerá 100% livre; Nós iremos retribuir à comunidade software livre; Nós não esconderemos problemas; Nossas prioridades são nossos usuários e o software livre; Programas que não atendem nossos padrões de software livre”. Para cada um dos itens há uma explicação dos motivos. O último item refere-se à política de aceitação desses softwares não-livres. (http://www.debian.org/social_contract)

⁶¹ Existe uma camiseta com dizeres que unem, humoristicamente, o Debian e a política. “apt-get install anarchism”, são seus dizeres, sendo apt-get install o comando para instalação de softwares no Debian.

uso de alguns softwares não-livres, o que levou a uma controvérsia aguda no movimento. Mais tarde, esses softwares acabaram sendo lançados também com uma licença livre, mas a imagem do produto continuou, de alguma forma, ligada ao episódio. O Gnome foi lançado justamente em reação ao uso de softwares não-livres pelo KDE, associando-se desde então, a uma alternativa mais livre. O Gnome é a interface padrão (embora outras possam ser instaladas pelo usuário) da distribuição Debian. Ironicamente, a licença livre adotada pelo conjunto de softwares que permite a construção de programas de interface gráfica como o Gnome pode ser usada para o desenvolvimento de softwares proprietários, enquanto a licença usada atualmente pelos softwares de construção gráfica usados no desenvolvimento do KDE – e que eram o objeto da controvérsia no passado – não. A imagem de “mais livre” para o Gnome e “menos livre” do KDE, no entanto, persiste.

Para completar, há na imagem os pinguins que utilizam itens do vestuário feminino, marcando o caráter não exclusivamente masculino da iniciativa (o que é algo importante, pois o mundo da computação e do software livre é eminentemente masculino) e os objetos identificados com a Bahia e a cultura negra de um modo geral, como o chapéu rastafári no pinguim ao fundo e o berimbau, logo à frente.

Por contraste, é interessante contrapor a essa figura uma outra, que está no cabeçalho do blog *Nerdson não vai à escola*. Seu autor, Karlisson Bezerra, é um jovem de Natal, Rio Grande do Norte, e intitula-se “desenvolvedor web, ilustrador e programador nas horas vagas”. “Quadrinhos feitos de nerds para nerds”, diz Bezerra. O personagem principal, Nerdson, é seu alter-ego.⁶²

⁶² Bezerra descreve assim o personagem principal de seu blog: “Nelson, mais conhecido como Nerdson, é um carinha nos seus vinte e poucos anos, que tenta levar uma vida pacata, faz faculdade de computação, mas acha que aprende mais em casa ou no trabalho, por isso mantém uma visão pessimista sobre o atual sistema de ensino, mesmo que em certos momentos esteja apenas exagerando. Trabalha numa empresa de desenvolvimento de softwares com seus colegas Beta e Lilo, gosta de ler, programar em C, C++, Python, Shellscript e PHP, jogar videogames e participar de eventos de informática, além de desenhar os quadrinhos do Libman & APIboy. Curte bandas de heavy metal, punk e progressivo, como Iron Maiden, Ramones e Pink Floyd. É um grande fã de Linus Torvalds, o criador do Linux.”



Nesta imagem, misturam-se os personagens do blog, citações a jogos clássicos de vídeo game e símbolos de empresas ou iniciativas ligadas ao software livre. Os dois únicos símbolos que se repetem comparados à primeira imagem são o do kernel *Linux*, na figura do pinguim, e a pegada que representa a interface gráfica *Gnome*. Ao lado do pinguim nota-se um pequeno diabo, que representa o projeto *FreeBSD*. O *FreeBSD* é um sistema operacional livre derivado do *BSD* (*Berkeley Software Distribution*). Sua licença é altamente permissiva e iguala-se, na prática, ao domínio público. Por não impedir que o código por ela regulado seja incorporado em softwares proprietários, ela é criticada pela *Free Software Foundation*. Ao mesmo tempo, os grupos mais identificados com a corrente *open* do software livre apontam um pioneirismo do grupo de *Berkeley* na ideia de software livre, em lugar da iniciativa de *Richard Stallman*, já na década de 1980, quando do início do projeto *GNU*.

Há, na figura, um conjunto de animais que simbolizam vários projetos de software, como o elefante (que pode simbolizar tanto o banco de dados *Postgre* quanto a linguagem *PHP*), o rato (símbolo da ferramenta gráfica *Gimp*), entre outros. Segundo o autor, foi dada a preferência por elementos “caricaturizáveis”. “Eu teria colocado a logo do *Ubuntu*, mas ela não se encaixa em nenhum lugar ali. Eu nem uso o *Suse*, mas coloquei o camaleãozinho dele ali no asfalto, só porque é um desenho”, disse ele, quando lhe perguntei sobre a escolha dos elementos. Pela resposta podemos perceber, apesar de sua afirmação em favor do que considera “caricaturizável”, a preferência por usar na ilustração os softwares que ele mesmo utiliza, como o *Ubuntu*. Há também símbolos de empresas,

como o M da cadeia de fast-food McDonald's – virado de ponta cabeça. Ao lado, lê-se “use W”, como em um outdoor. O W é símbolo do software livre para blogs Wordpress, o mesmo usado no blog de Bezerra. Um pouco mais à esquerda, ao lado da Torre Eiffel, um prédio ostenta o letreiro da empresa Google. Abaixo, ao lado do camaleãozinho citado por Bezerra, está o Amigooogle⁶³, um dos personagens das histórias do blog.

Ao se analisar as imagens, o objetivo não é classificar seus autores como aderentes ou não a determinada corrente política do movimento software livre. Elas, sim, oferecem indícios sobre as escolhas de cada um, contudo trata-se de um processo muito mais complexo que requereria um outro procedimento de pesquisa. Mais interessante neste momento é mostrar como essas escolhas de símbolos, que estão presente no cotidiano de expressão dos membros do movimento, podem ser interpretadas – mesmo que à revelia de seus autores – como manifestações políticas sobre o que é o software livre, para que serve, quem dele faz parte – ou é aliado – e qual versão da história de seu surgimento deve ser endossada. Vestir um pinguim com uma camiseta estampada com um gnu e assim, por meio de imagens, dizer GNU/Linux, como pede Stallman, significa apontar para uma ideia de software livre. Já unir Google e o pinguim significa reforçar uma outra imagem do software livre, mais integrado ao ambiente empresarial. Não há contradição nisso, ambas as representações são aceitas como próprias.

No Fisl, é possível acompanhar um verdadeiro desfile desses símbolos. Em geral, as composições são como a da primeira figura analisada, da Rede Três Mosqueteiros Cooperativas de Software Livre, em que há um conjunto mais restrito de elementos, sendo fixado um posicionamento político específico. Espaços e o próprio público, por meio de sua vestimenta e acessórios, constroem-se

⁶³ Bezerra descreve assim o personagem: “O Amigooogle é uma instância física do Googlebot, que faz parte de um novo serviço chamado “Personal Google Friend“. Oferece pessoalmente os serviços do Google, e está sempre pronto para buscar seus...livros desaparecidos. O Amigooogle é feito de plástico, e tem grande admiração pelo Marvin, do *Guia do Mochileiro das Galáxias*.” (Ver <http://nerdson.com/blog/sobre/>)

pelo arranjo desses símbolos, manifestando apoio político aos grupos estabelecidos.

Ao que tudo indica, o uso desses símbolos, principalmente incorporados ao vestuário, serve a um duplo propósito. Em um nível geral, serve como elemento identificador dos membros do movimentos software livre como aderentes à causa, tanto para o público externo, capaz de reconhecer símbolos mais populares como o pinguim e o nome Linux, como para aqueles com alguma familiaridade com o mundo da computação, que reconhecem um maior número desses símbolos e os associa ao software livre. Já para os integrantes do próprio movimento software livre, os símbolos servem como indicadores de filiações a certas correntes políticas do próprio movimento. Funcionam, inclusive, em combinação, operando como sinalizadores de nuances no posicionamento político⁶⁴.

Certamente há uma relação entre os símbolos criados, popularizados e apropriados pelo software livre e as logomarcas de empresas. O pinguim, símbolo do software mais identificado com o software livre junto ao público geral, acabou por se tornar um mascote, não muito diferente de alguns adotados por grandes corporações. Os projetos de software, mesmo não se constituindo juridicamente como empresas, adotam logotipos, que funcionam como ícones de identificação. Porém, é pelo significado político que possuem esses projetos que essas imagens passam a ser utilizadas como ferramentas de expressão de ideias e posicionamento dos indivíduos.

Na abertura, as autoridades fazem o choque entre *free* e *open*

São 14h do dia seguinte ao de minha chegada e me encaminho para a sala de abertura do

⁶⁴ Nesse sentido, é possível relatar o caso do membro da Free Software Foundation Latin America, Alexandre Oliva. Oliva é, também, funcionário da Red Hat, identificada com o grupo *open*. Durante o evento Oliva circula com um chapéu vermelho, símbolo da Red Hat, e uma camiseta da FSFLA. Assim, marca a si mesmo como aderente aos princípios do grupo *free*, porém sem compartilhar das ideias anti-corporativas de parte do grupo *free*.

evento. No caminho para a sala onde acontecerá a solenidade, lembro-me de uma frase dita a mim por Mário Teza, que "a abertura é a hora deles [dos políticos, das autoridades] e o encerramento é a hora nossa [da comunidade]". De fato, a abertura é um dos momentos formais do evento, quando discursam as autoridades federais, estaduais e municipais e os principais patrocinadores, notadamente executivos de estatais. Os membros da organização do Fisl que sentam-se à mesa, nesse dia abandonam a camiseta do evento e trajam terno. Essa formalidade, no entanto, não significa necessariamente o apagamento das divergências entre os grupos *free* e *open*. Ao contrário, muitas vezes, como foi o caso de 2008, são as autoridades que trazem à tona as diferenças políticas até de maneira mais contundente do que num debate entre membros do movimento, cujas posições já são suficientemente públicas.

Além das falas na mesa, diversos outros itens (cênicos, de vestuário, comentários da plateia) que fazem parte da abertura (como também do encerramento) podem ter sentido e serem melhor entendidos a partir da distinção entre *free*, ainda que ressignificado à moda brasileira, e *open*. Além disso, abertura também é interessante por permitir a políticos e autoridades que estabeleçam conexões entre diversas questões sociais (exclusão social, educação, autonomia tecnológica nacional, etc.) e o software livre. O público reage aceitando ou rejeitando essas conexões.

Em 2008, a opção foi usar uma sala diferente da habitual: o auditório mais importante e luxuoso da PUC-RS, com cadeiras fixas e estrutura de teatro. Nos anos anteriores, mesmo quando o evento aconteceu também na PUC-RS, a opção foi usar a maior sala disponível, embora não sendo a com melhor estrutura. A principal consequência da mudança para uma sala menor foi deixar parte da audiência de fora. Quem chegou pontualmente não pôde entrar e um telão foi disponibilizado.

Antes da abertura, sobre o palco ainda vazio, está uma longa mesa onde os convidados se sentariam. Ao fundo, um painel gigante com o nome de todos os patrocinadores do evento. Comparado a anos anteriores, é o maior número de patrocinadores, tendo surgido especialmente neste ano as

empresas de comunicação: Globo, UOL e Terra. Segundo um dos organizadores, haveria duas explicações. A primeira, refere-se a um maior esforço de contato com a mídia e busca de patrocínios por parte dos captadores de recursos. Mas, além disso, teria havido uma espécie de "efeito Campus Party": "Ele tem um proposta diferente da do Fisl mas, por ter ocorrido em São Paulo, durante uma semana, e ininterruptamente, com diversas empresas e soluções, serviu para que as questões de tecnologia e internet tivessem bastante visibilidade". O Campus Party é um evento realizado na Espanha desde 1997, e que reúne diversas manifestações culturais em torno das tecnologias de informação e comunicação, como games, blogs e celulares. Patrocinado pela Telefônica, teve uma primeira edição brasileira em 2008, com boa repercussão na imprensa e grande presença da comunidade software livre. "Acho que isso serviu para estabelecer um *link* entre os eventos, para que as empresas de comunicação maiores descobrissem a importância e a irradiação do tema software livre", diz meu informante. O principal organizador do Campus Party Brasil foi Marcelo Branco. Técnico em telecomunicações, Marcelo Branco, como dito, é um dos pioneiros do movimento software livre em Porto Alegre. Ligado ao Partido dos Trabalhadores, participou ativamente das primeiras edições do Fórum Social Mundial. Mais tarde, após mudanças no governo, deixou o país e trabalhou para o governo da Catalunha, na Espanha. Os contatos para o Campus Party Brasil podem ser atribuídos em parte a essa experiência no exterior.

À frente da grande mesa, em cima do palco, chama a atenção uma pilha de feno ou mato. "Deve ser um protesto do Movimento Sem Terra", comenta alguém na fila de cadeiras atrás da minha. Embora com tom jocoso, o comentário não é absurdo. Desde 2004, o Fisl vem cooperado com comunidades quilombolas do Rio Grande do Sul, recolhendo para elas dinheiro a ser usado na compra de "sementes livres", não geneticamente modificadas. A pessoa mais ligada ao projeto é Mario Teza. Teza afirma que a iniciativa tem também por objetivo trazer a questão para o movimento software livre, mostrar que

se trata da mesma luta, que há paralelo entre um movimento contra as licenças proprietárias de software e o movimento contra as patentes sobre a vida e contra empresas como a Monsanto. Segundo ele, em 2004, a organização ficou sensibilizada com o alto índice de mortalidade infantil de uma comunidade indígena de Mato Grosso do Sul, caso de grande repercussão nacional. Pensaram em ajudá-la mas, na mesma época, aconteceu uma forte seca no Rio Grande do Sul, que afetou comunidades indígenas e quilombolas do estado, especialmente aquelas que haviam plantado sementes transgênicas, prejudicando o plantio do ano seguinte. O efeito negativo, segundo ele, teria sido menor para os que se ativeram às sementes não transgênicas. Sobre o assunto, o entrevistei para uma matéria jornalística, em 2005. Reproduzo a seguir parte desse texto, que dá conta da proximidade dos movimentos – questão que lhe fiz – e de eventuais resistências a essas ligações existente dentro do movimento software livre.

"O objetivo agora é criar uma cadeia produtiva livre, em que os agricultores não sejam obrigados a pagar os royalties abusivos cobrados pelas transnacionais dos transgênicos. No próximo ano, as comunidades beneficiadas contribuirão, com o fruto de seu trabalho, para fazer crescer ainda mais o Banco de Sementes Livres. “Não podemos ver reproduzido na agricultura o monopólio como no mercado de software”, afirma Teza. “Queremos liberdade para o código genético, assim como queremos que sejam livres os códigos-fonte dos programas de computador”, completa.

(...) Segundo Mário Teza, a lógica é a mesma, a indústria é conivente com o uso ilegal porque este, no futuro, gerará mais lucros a ela. “Veja o caso da soja. No primeiro ano, a Monsanto cobrou uma certa quantia pela saca colhida. No ano seguinte, esse valor está em negociação. Até onde isso vai?”, afirma.

Teza acha que o foco das campanhas contra os transgênicos está errado e, por isso, ainda não é bem compreendida pela comunidade software livre. Para ele, é preciso mostrar que os transgênicos são produzidos porque o objetivo é obter uma patente sobre a espécie e, assim, controlar os agricultores e a produção. “O ponto não é dizermos que faz mal para a saúde ou para o meio-ambiente – argumentos para os quais nem os movimentos sociais nem a indústria podem exhibir provas conclusivas. Temos que mostrar que o que está em jogo é a autonomia da produção.”, afirma

Para isso, Teza imagina que, além dos debates, é preciso incentivar ações práticas, afirmativas, algo que é característico do movimento software

livre. (...)

No ano que vem, a ideia de Teza é combinar o Banco de Sementes Livres com discussões que mostrem a ligação entre as tentativas de apropriação dos códigos da informática e a apropriação sobre os códigos da vida. O debate promete."

Esse plano de realizar debates sobre a questão nunca foi levado a cabo. A conjunção entre debate e prática de que fala Teza se mostrou mais fácil do lado da ação. Pautar o debate público do Fisl com um tema lateral ao software se mostrou mais complicado do que manter uma ação social que apenas insinua uma ligação entre os dois movimentos, enquanto a imagem mais forte é a da filantropia ou algo próximo à responsabilidade social das grandes empresas. Em 2007, quando perguntei a ele sobre a iniciativa, Teza disse haver resistências ao Banco de Sementes Livres dentro da organização do Fisl. De qualquer maneira, em ação similar que também poderia ser rotulada sob o mesmo chapéu de responsabilidade social, neste ano o Fisl declarou-se neutro em emissões de carbono, adotando práticas de mitigação para os gases estufa emitidos.

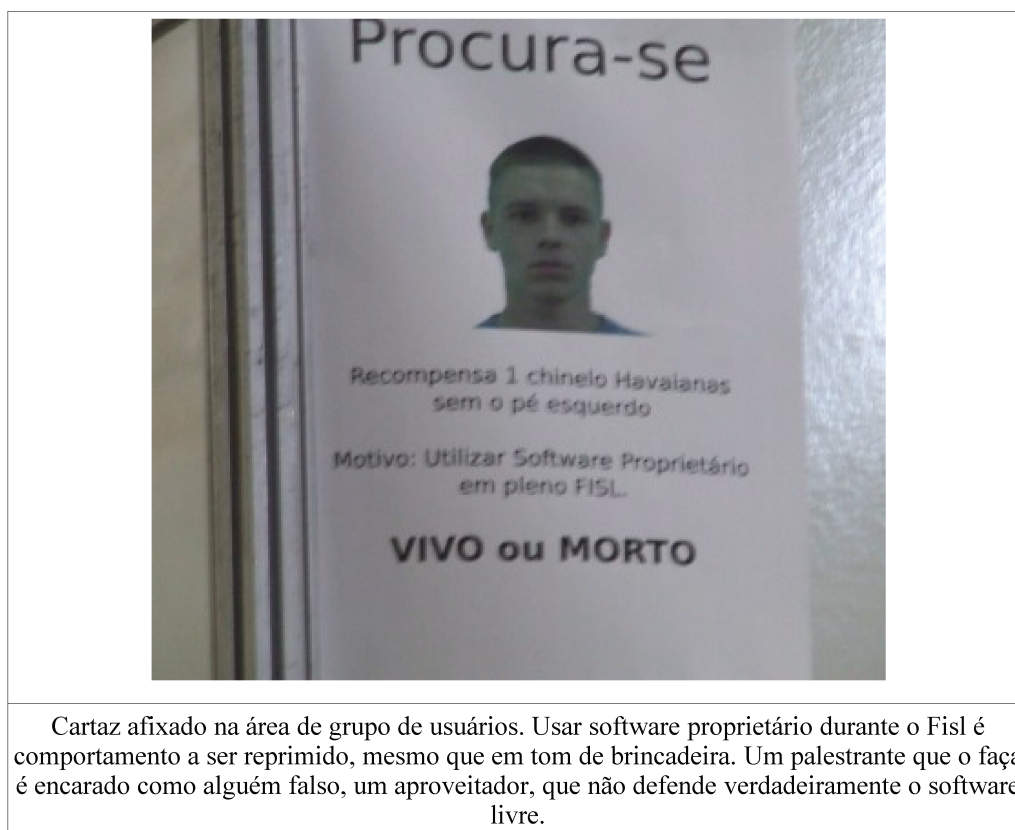
Antecedendo o início da sessão, algumas pessoas atravessam os corredores carregando sacos de sementes. "É um protesto do MST", repetem. No palco, dois telões mostram propagandas institucionais da Caixa Econômica Federal, que falam em responsabilidade social e tem como personagem senhoras de idade, nada muito diferente de outras propagandas do mesmo estilo, em que empresas procuram demonstrar preocupação social.

No chão, em frente ao palco, um garoto negro monta uma bateria que parece ser artesanal. Embora haja negros no movimento software livre, a presença talvez seja comparável à existente nas universidades: não correspondem à divisão populacional e apenas alguns procuram marcar uma identidade étnica. Em todas as quatro categorias de público descritas (burocratas, ativistas, nerds, empresários), a maioria é de brancos. Isso muda com os "incluídos digitais", jovens da periferia que

passam a usar software livre por meio dos programas governamentais de inclusão. Estes já se vestem de maneira diferente, os que são negros manifestam identidade étnica e parecem ter um interesse maior pela política. No evento, os “incluídos” aparecem em pequeno número, andam com seus grupos e, apesar de frequentarem também as sessões técnicas, estão em maior número que os nerds nos debates mais políticos, principalmente os relacionados às licenças livres para a cultura e sobre inclusão digital.

"É muito cacique pra pouco índio", comenta alguém sentado na fileira de trás. Ao meu lado, um engravatado, com um tripé à sua frente, usa um Macintosh. Ao notar o formato do arquivo de vídeo que será exibido em sequência pelo telão, ele comenta comigo, sem que eu houvesse puxado a conversa: "Duro é num Fisl é renderizarem o vídeo em WMV". WMV é a sigla para Windows Media Video, formato de arquivos proprietário desenvolvido pela Microsoft. Sua crítica é porque há formatos livres para arquivos de vídeo e o uso de um formato proprietário é visto como uma contradição com o evento. Uma das maiores dificuldades das distribuições livres é que necessitam, muitas vezes, incluir um tocador de arquivos de mídia que execute formatos proprietários, para garantir que o usuário possa visualizar todos os arquivos que recebe, e muitas vezes esses tocadores incluem software proprietário, o que afetaria a “pureza” do sistema. O irônico é que o autor da reprimenda usava um computador com sistema operacional proprietário, o MacOS. Produzido pela Apple, o Macintosh é um equipamento visto como de grande qualidade técnica. Seu sistema operacional, que incorpora códigos livres cujas licenças permitem interação funcional com software proprietário e alteração da licença original, é reputado como de boa qualidade e é muito menos mal-visto do que o Windows. Mas o que causa estranhamento no comentário é que aquele é um tipo de reprimenda que cairia bem vindo de um “radical” *free* que, mesmo que fosse dono de um equipamento da Apple, teria substituído seu sistema operacional por um outro, livre. Para boa parte do público do evento, o reclamante estaria em situação mais criticável do que a crítica que faz. A figura abaixo é um exemplo do tipo de restrição, ainda que

em tom de brincadeira, que há no Fisl quanto ao uso de sistemas proprietários.



Em seguida começa o evento, com exibição do vídeo sobre o projeto Arroz Quilombola. Após o vídeo, um garoto anuncia a apresentação de um grupo que mistura rap com música tradicional. A música fala de negros, África, escravidão e é tocada com garrafas d'água e uma bateria velha. É estranho porque aquela apresentação parece estar deslocada do resto do evento. Por um lado, encaixa-se nas preocupações sociais que se refletem em debates como o da inclusão digital. Por outro, soa artificial para um ambiente de negócios. E parece ser com esse sentimento misto que o público recebe aquilo: não rejeita, mas ao mesmo tempo não se identifica. “Livre de transgênicos, livre de agrotóxicos”, cantam. O vínculo entre as causas não é evidente, automático. As palmas, ao final, acabam sendo mais fortes do que o esperado.

Em meio à música, chegam as autoridades. Ocupam a mesa que está no palco. "Autoridades

chegam com meia hora de atraso", alguém comenta atrás. Quando a música se encerra é distribuído um folheto com uma receita para o arroz quilombola. A sala está completamente lotada.

"Esse aí não traiu o movimento...", diz alguém em tom jocoso. Não consigo identificar o autor da frase nem a quem ele se dirige ou refere-se, mas a expressão "traiu o movimento" é algo muito comum naquele ambiente e para o software livre como um todo. A ideia de compromisso, que implica em uma postura pública e em hábitos cotidianos com referência principalmente ao uso de determinados softwares (ou não-uso de alguns, os proprietários) é algo muito comum. Trair o movimento pode significar desde uma ação individual de repercussão geral, como o endosso ou a colaboração com aqueles que são construídos como "inimigos" do movimento (o maior deles é a Microsoft), como pequenas ações privadas, como trocar arquivos em formato proprietário, ou usar software proprietário em âmbito doméstico. Ou ainda utilizar arquivos em WMV, como me alertou o usuário Macintosh.

São então anunciadas as autoridades. A primeira é o vice-governador do Rio Grande do Sul, Paulo Afonso Feijó. Em seguida, Sady Jacques, então coordenador geral da ASL. Depois, Roberto Requião, governador do Paraná. Só depois os outros. Todos estão de terno, menos Requião. Executa-se o Hino Nacional. Todos se levantam, mesa e plateia.

O primeiro a falar é Jacques. Jacques tem história de vida semelhante à de Branco e Teza: sindicalista, funcionário público e ligado ao PT. Ele ressalta o crescimento do software livre, explica as trilhas do evento, fala de mudanças na organização. Aponta a alteração no processo de seleção de palestras. Jacques fala também no Ginga, software livre utilizado na TV digital. Parte do movimento software livre brasileiro envolveu-se no debate sobre o padrão da TV digital, opondo-se à adoção da modulação do sinal com padrão japonês. Desde antes dessa disputa em torno do padrão se configurar, alguns pesquisadores já estavam envolvidos no projeto do Sistema Brasileiro de TV Digital (SBTVD) e seus esforços acabaram canalizados no *middleware* (uma camada de software intermediária) Ginga.

Porém, parte do movimento permaneceu alheia a esse debate, entendendo-o como não relativo ao software livre. É difícil afirmar se essa menção ao Ginga estaria presente no discurso de abertura não fosse o apoio dado ao Fisl em 2008 pelo Ministério do Planejamento, que hospeda o software do projeto, numa iniciativa nomeada como Portal do Software Público Brasileiro. Jacques aponta o que chama de diversidade do software livre, "onde todos têm espaço".



Em seguida, é a vez do representante Marista falar. Os Marista são um grupo católico que administra a PUC-RS, onde o evento foi realizado, entre outras instituições educacionais pelo país. Ele fala em geração de "oportunidade, principalmente para os jovens das periferias" e em reciclagem de computadores. "A descoberta da solidariedade das relações. O conhecimento não como mercadoria, mas como evolução para a sociedade".

O próximo é o representante da Secretaria de Logística e Tecnologia da Informação do Ministério do Planejamento. Enquanto fala, algumas pessoas começam a ir embora. Já são 17h, o

evento está atrasado e a grade normal de palestras recomeça.

O seguinte é o representante dos vereadores de Porto Alegre. Lembra que a prefeitura apóia o evento desde o início. Fala que outros lugares querem levar o Fisl de Porto Alegre. O vereador justifica porque as instituições de Porto Alegre usam muito "código aberto". Fala em vocação de Porto Alegre e de aplicativos livres usados pela prefeitura. Combina termos e fala em "software aberto", ou seja, software livre e código aberto.

Marcos Mazoni vem em seguida. Como dito, ele esteve entre os organizadores das primeiras edições do Fisl quando, no período entre 1999 e 2002, foi presidente da Procergs. Assim que Roberto Requião assumiu o governo do estado, Mazoni tornou-se o comandante da empresa estadual de informática do Paraná (Celepar). No Fisl de 2008, Mazoni estava como chefe do Comitê de Software Livre do governo federal e presidente nacional do Serpro. Em seu discurso, Mazoni lembra inicialmente de sua participação no primeiro Fisl – quando, segundo ele, se esperava por volta de mil pessoas e mais de 1,8 mil acabaram aparecendo. Depois, lê mensagem do presidente Lula, que foi convidado oficialmente para estar lá. É um discurso em que se fala de participação e justiça social, e certamente não foi elaborado pelo presidente, mas por alguém que parece ser do movimento, talvez o próprio Mazoni. A fala junta as “quatro liberdades” do software livre com ações do governo como o programa Gesac, que instala parabólicas para captação de sinal de Internet em escolas do país e oferece serviços de informática (o Gesac foi herdado do governo Fernando Henrique Cardoso e teve seus cargos ocupados por pessoas bastante identificadas com o software livre); os Pontos de Cultura (centros de cultura digital, capitaneados pelo Ministério da Cultura, que usam software livre para programas de multimídia e que também contaram com envolvimento de parte do movimento software livre). Mazoni/Lula também lembra que Porto Alegre é a cidade-sede do FSM e "que sempre deu espaço ao software livre".

Em seguida, fala o governador do Paraná, Roberto Requião. Adota um discurso bastante combativo, mais do que muitos dos líderes tidos como mais radicais do software livre. Acompanhei sua presença na abertura do Fisl de 2005 e o tom foi o mesmo. O discurso de 2008 foi republicado em site do governo do Paraná. Marco em negrito as expressões que me chamaram mais a atenção. A maioria também foi anotada por mim no momento do discurso.

"[...] Mais uma vez, manifesto a minha satisfação por fazer parte deste movimento de cultura livre, cujo avanço seguro e valente no mundo todo deve ser comemorado.

Não quer dizer que vencemos, que derrogamos todos os empecilhos, os tantos e fortes embaraços.

Pelo contrário. **Insidiosos, solertes, com cartas e seduções multiplicando-se em mangas, coletes e bolsos, os senhores dos sistemas proprietários vão continuar fazendo de tudo para que a nossa liberdade de acesso, de criação e de uso da rede seja inibida, restrita, vigiada, reprimida, desestimulada.**

Contudo, e apesar de tudo, avançamos. Ousaria até mesmo dizer que **entre as frentes de luta abertas contra a dominação global e o avanço açambarcador do mercado, a frente do software livre foi a que obteve melhores resultados. De tal forma que pode servir de exemplo e estímulo a outros combates.**

É gratificante poder comemorar avanços nessa já longa jornada por **um outro e possível mundo.**

Vejo neste auditório muitos rostos jovens. É provável que os jovens predominem no **movimento**. Mas vejo também cabelos grisalhos ou brancos como os meus. É, nós, os mais velhos, sabemos como é estimulante, rejuvenescedor acumular vitórias, ampliar conquistas, ganhar terreno. Ainda mais a **nossa geração, veterana de tantas e tão duras provações.**

Melhor ainda. Estamos avançando exatamente na frente do conhecimento, da produção, democratização, universalização do conhecimento. Justo o campo **cujo domínio pelos países imperiais** teve sempre como resultado a nossa **submissão, o nosso atraso, o nosso subdesenvolvimento**, a nossa pobreza, a nossa dor.

Conhecer para se libertar.

Permitam agora que cante a minha aldeia.

No Paraná, não temos dúvidas quanto as nossas escolhas. **Temos um lado, claramente definido** e transformado em política de Governo.

Todo o planejamento estatal, todo o estímulo e indicação de investimentos, todas as ações públicas têm as marcas de nossa opção **pelos mais pobres, pelos trabalhadores, pelos pequenos agricultores, pequenos comerciantes e**

empreendedores. Por aqueles, enfim, que o mercado relega à margem ou quer absorver como simples engrenagens do consumo.

Logo, coerentemente, no Paraná, o uso e o desenvolvimento do software livre faz parte das decisões estratégicas do nosso Governo.

Assim como acontece com a nossa **resistência à tentativa de controle da agricultura brasileira pelas multinacionais produtoras de sementes geneticamente modificadas**, a nossa opção pelo software livre é um **enfrentamento àqueles que querem monopolizar a tecnologia da informação.**

Não há diferença entre a manipulação dos genes das sementes de soja, milho e o bloqueio dos códigos-fonte dos programas de computador.

Naquele e neste caso, o que se pretende é o controle do fluxo e distribuição de riquezas através do controle do conhecimento.

No Paraná, estamos rompendo, estilhaçando esse outro **grilhão com que nos querem acorrentar à dependência.** Tem sido uma experiência gratificante.

[...]

Isso sem falar no maravilhoso mundo que a informática abre para as nossas crianças e jovens. **Não há emoção tão forte que se compare ao ver lá no mais remoto, escondido, humilde município, crianças viajando pela internet, descobrindo, aprendendo, crescendo, incluindo-se no universo.**

[...]

Fazendo contas, é possível dizer que, desde a implantação do software livre, em 2003, até o momento, **deixamos de contribuir com Bill Gates et alia coisa de 180 milhões de reais.** Recursos que investimos no desenvolvimento tecnológico do Estado, na capacitação de nossos profissionais e na modernização de nossa empresa de informática pública, a Celepar.

[...]

Com tudo o que avançamos, nossas possibilidades são ainda imensas. Cada vez menos **dependentes dos sistemas proprietários**, estamos consolidando a nossa **autonomia.**

[...]

Em breve, esse sistema de gestão hospitalar estará à disposição de todas as unidades de saúde do Paraná e do Brasil. Eis aqui um modelo de compartilhamento que somente uma **tecnologia solidária** poderia proporcionar. Outro programa pelo qual temos tanta estima, é o programa de Inclusão Digital. Os nossos centros Paranavegar já somam 120 unidades, espalhados notadamente em localidades de menor Índice de Desenvolvimento Humano, logo as que mais precisam de acesso à informação e à comunicação para **superar a exclusão social e a desigualdade.**

Falei da emoção da luz que emana do computador em uma remota escola de um distante município. **Não é menor a emoção ver crianças e adultos em assentamentos rurais, em aldeias indígenas, em comunidades quilombolas**

reunidas em torno dessas maravilhosas máquinas e suas infinitas possibilidades. O programa Paranavegar permite que isso aconteça.

(...)

Reiteradamente, em meus pronunciamentos, tenho falado sobre a **contradição entre Mercado e Nação. A oposição entre os interesses nacionais e a transnacionalização da economia, que nos configura como meros fornecedores de produtos primários, de commodities, e como consumidores de produtos acabados.**

O software livre põe-se hoje como uma das armas mais poderosas para a construção e consolidação de nossas nações, da Nação Brasileira, da Nação Argentina, da Nação Chilena. Da nossa identidade Latino-americana. Da independência Latino-americana.

O conhecimento é chave do desenvolvimento. Os sistemas proprietários são condicionantes, são amarras, **equivalem-se aos ordenamentos reais do tempo colonial, que restringiam, que manipulavam, que escorchavam, que submetiam.**

Logo, esse Fórum Internacional ganha uma dimensão, uma amplitude que **ultrapassa os limites do debate técnico**, para se firmar como um espaço de construção da nossa própria cidadania.

[...]

Contem com o Paraná, como nós contamos no início do nosso Governo com o grupo de gaúchos capitaneados pelo Marcos Manzonni, que nos possibilitou a montagem deste sistema maravilhoso que viabiliza de forma extraordinária o nosso Estado."

Metade da audiência aplaude de pé, efusivamente. Outra metade, permanece sentada, mas também aplaude. Esse tipo de discurso certamente não é unânime, mas é bem recebido por parte significativa do movimento brasileiro. Requião conseguiu agregar em sua fala diversas dicotomias, bastante extremadas, exageradas, mas que encontram eco em especial na comunidade software livre da América Latina⁶⁵. Nunca acompanhei, nem conheço registro, de um discurso como esse em eventos similares em países desenvolvidos. Software livre *versus* software proprietário; liberdade *versus* dominação; pobres *versus* ricos; nação *versus* mercado; autonomia *versus* dependência; pequenos

⁶⁵ Em dissertação de mestrado, analisei projetos de lei – propostas brasileiras e a lei aprovada pelo Peru – voltados à adoção preferencial de software livre pelo setor público. Aponto que a relação de dependência e subordinação dos usuários de software proprietário e as empresas donas do software, em especial, é entendida como similar à relação de dependência entre países periféricos e centrais. (Evangelista, 2005)

versus grandes. Requião cita ainda a frase-tema do Fórum Social Mundial: um outro mundo é possível. Produz um sentido bastante claro de software livre, em que este se encaixa em uma disputa entre projetos políticos de esquerda e direita. Ao mesmo tempo, o coloca não como um novo modelo de negócio, mas como anti-mercado. Por meio do software livre, Requião reencontra-se com um discurso nacionalista de esquerda, em que os países periféricos se encontram atrasados e empobrecidos devido ao domínio de um colonizador externo que impede a nação de agir de acordo com seus próprios interesses. Insere-se aí a tecnologia, em sua versão solidária, como emancipadora, iluminadora, com infinitas possibilidades e como meio de superação de desigualdades.

Em seguida, fala o vice-governador do Rio Grande do Sul, Paulo Afonso Feijó, para fechar a cerimônia. Improvisado, seu discurso é uma reação clara e até certo ponto agressiva ao que foi dito por Requião. Há um evidente mal-estar. Feijó começa falando em competitividade do parque tecnológico gaúcho, que seria maior que o de outros estados do Sul. Sua provocação é recebida com risos e algumas palmas. "Software não é questão de ideologia mas de liberdade de escolha". A frase, um aparente pedido de neutralidade, é uma tomada de posição. "Liberdade de escolha" é expressão corrente entre os defensores do software proprietário, que trocam o sentido de liberdade do software livre, apontado que os indivíduos devem ser livres para escolherem seus softwares e, assim, questionando a ideia de que o software proprietário é algo antiético – ideia presente no grupo *free*. O vice-governador diz ser empresário de software e fala sobre como vê o Estado: o governo não cria riqueza, mas se apropria do que os empresários criaram. "Bom governo é aquele que menos ocupa espaço e não é notado". Segundo ele, em sua empresa produz-se tanto software livre como proprietário, ao gosto do freguês. O ponto de vista do vice-governador é bastante diferente do de Requião. Não há dominantes e dominados, mas estados e empresários em competição. Quem vence é quem consegue dar mais liberdade aos empresários, diminuindo impostos, para que exerçam seu potencial competitivo e produzam

crescimento, de modo a que enriqueçam a sociedade como um todo.

O discurso de Feijó foi recebido com aplausos rápidos, de apenas parte da plateia. A contraposição entre Feijó e Requião gerou tal incômodo que, logo após o fim do discurso do vice-governador, boa parte do público foi embora. A organização previa ainda a assinatura simbólica de alguns acordos e o anúncio de iniciativas de patrocinadores governamentais, mas tudo acabou ofuscado pelo episódio.

A grande mídia que esteve no evento parece não ter percebido a divergência pública entre as autoridades. De fato, para o público não habituado com o debate que envolve os softwares livres expressões como “liberdade de escolha” perdem o sentido de posicionamento político que é entendido por quem pertence ao movimento. Nesse sentido, é interessante trazer o registro sobre o acontecimento feito por um blog. Mantido por um membro do movimento estudantil de estudantes da computação, o posicionamento é pró-Requião e exagera na boa recepção que teve o discurso do governador paranaense. Não encontrei nenhum registro pró-Feijó, o que de forma alguma significa ausência de identificação de parte do movimento software livre com suas ideias ou, pelo menos, uma recepção ruim dado o modo como elas foram expostas.

"Figura já conhecida do FISL, o governador do Paraná Roberto Requião participou mais uma vez da cerimônia de abertura do evento (...). Ovacionado pelo público durante a sua de cerca de 15 minutos, o Requião destacou a política agressiva de software livre desde a sua primeira gestão e que tornou o estado do Paraná uma referência mundial em políticas públicas de software livre. (...) Requião foi aplaudido 3 vezes durante o discurso e aplaudido de pé ao final do seu discurso.

O mico da abertura ficou por conta do vice-governador do RS que, durante sua breve fala citou ser dono de uma empresa de software proprietário (putz, haja cara-de-pau!) e que, em indireta ao discurso que acabava de terminar, preferiu dizer que governo bom é governo que não aparece e terminou sem falar absolutamente nada sobre a política de software do RS. Melhor calar a boca logo para não falar mais besteira...

Tipo, talvez seja hora de o FISL pensar em mudar de estado..." (Post no Educativre (<http://educativre.wordpress.com/2008/04/18/abertura-do-fisl-tem->

Longe de ser um evento fortuito, debates como o entre o vice-governador do Rio Grande do Sul e o governador do Paraná repetem-se todo ano no Fisl, seja de forma clara - como na abertura da nona edição - seja em palestras paralelas ou nos corredores do evento. Ele é, ao mesmo tempo, uma transposição e um acirramento do debate entre as correntes *free* e *open* do movimento. É transposição por sustentar-se em bases semelhantes: de um lado, uma assumida utopia que envolve a construção de novas relações sociais, de trabalho e de independência; de outro, um pretenso pragmatismo que prega a convivência e a complementariedade entre os dois modelos, esforçando-se por mostrar-se alheio a o que considera questões políticas ou ideológicas – não-técnicas, embora a discussão sobre economia e mercado seja bem aceita. Ao mesmo tempo, trata-se de um acirramento, por dar ao debate uma configuração específica e rara em outros lugares do mundo, em especial nos países ricos. Mesmo que alguns representantes públicos mais evidentes do grupo *free* também recusem uma filiação direta a partidos ou grupos políticos – assim como fazem de forma mais evidente os *open* –, eles se fazem presentes de maneira clara e, no Fisl, levam o debate interno do movimento para além das questões de direito autoral, patentes e desenvolvimento tecnológico, abarcando também as implicações do software livre para a justiça social, igualdade de oportunidades, desenvolvimento econômico local e autonomia nacional. A acentuada presença de militantes sociais e sindicalista na implementação do movimento software livre no Brasil, em especial em Porto Alegre, conformou uma visão específica sobre os propósitos do movimento. Esta permite que militantes de outros países, integrantes do grupo *free*, extrapolem os habituais limites de suas atuações.

Em 2006, por exemplo, a Microsoft tentou estar presente no evento pela porta dos fundos, associando-se a um patrocinador do Fisl que, mais tarde, revelou-se patrocinado pela Microsoft. Esse

patrocinador, uma pequena empresa de jornalismo sobre tecnologia, montou uma mesa de discussão no meio do Fisl, articulando para que os debatedores fossem: um representante de Microsoft; e uma empresa de software livre que, no mês seguinte, anunciaria trabalhos para a Microsoft. Como tudo aconteceu em uma tarde 21 de abril, dia de Tiradentes, Richard Stallman, palestrante em outras sessões, interrompeu a mesa em altos brados, gritando de maneira jocosa: "libertas quae sera tamen". Embora seja difícil imaginar que falas como a de Requião pudessem ser repetidas por Stallman de maneira completa (há semelhança mas não coincidência entre as duas posições), o presidente da Free Software Foundation usou a lembrança de um movimento de libertação colonial para provocar representantes Microsoft, que acabou associada a um país colonizador. Sobre o acontecido, a revista *Veja* publicou a seguinte entrevista com Stallman, em que ele comenta a presença da Microsoft:

POR QUE O SENHOR FEZ ESSE PROTESTO?

Não há espaço para a Microsoft em um evento daquele tipo. Software livre é uma questão de liberdade, enquanto o da Microsoft é distribuído de forma a subjugar o usuário. As pessoas me disseram o que significava a frase e, como era o dia da comemoração, me pareceu apropriada. Existem muitas semelhanças entre a colonização eletrônica e o sistema colonial antigo.

POR EXEMPLO?

O sistema colonial recruta elites locais para conseguir subjugar o resto da população. Ao fornecer cópias grátis de seus softwares, que não são livres, para escolas, a Microsoft está usando a escola para criar uma futura dependência tecnológica na sociedade.⁶⁶

O encerramento

A sessão de encerramento pode ser entendida como complementar à sessão de abertura, por isso opto por falar sobre ela antes de abordar o restante do evento. A frase de Mário Teza (“agora é a nossa vez”), faz crer que o encerramento é a hora de se ficar à vontade, de se fazer o que se quer. A abertura é

⁶⁶ “Programa livre” em revista *Veja*. Disponível em <http://veja.abril.com.br/030506/gente.html>, acessado em 15/11/2009

dedicada a se retribuir o apoio das autoridades, a dar espaço àqueles que de alguma forma viabilizam a realização do evento daquele ano para que liguem o software livre a suas agendas políticas. Já o encerramento é quando o evento se volta para dentro, reforçando as alianças internas e submetendo-se ao juízo do público, ainda que indiretamente. É a hora de outras retribuições: às figuras da comunidade, de preferência às que são quase uma unanimidade. O ambiente é descontraído, mas também mais crítico, com mais intervenções da plateia e menor respeito ao protocolo. Também é o momento de se apresentar e comemorar os números do evento, que tem crescido a cada ano.

A sala – a mesma onde aconteceu a cerimônia de abertura – está cheia. Muita gente já foi embora, principalmente os burocratas e os empresários. A plateia é mais jovem, com grande número de estudantes, principalmente aqueles que vieram em caravanas de outros estados e que só vão embora, em bloco, quando o evento realmente se encerra.

No palco, John Maddog Hall abre a sessão – que ainda não se trata do encerramento propriamente dito. Maddog é figura histórica do software livre mundial e do Fisl. Quando ainda trabalhava na Digital Equipment Corporation, no início da década de 1990, conheceu o trabalho de Linus Torvalds no kernel Linux e conseguiu doações de equipamentos para que Torvalds fizesse seus primeiros testes. Mais tarde, tornou-se diretor executivo da Linux International, associação sem fins lucrativos de empresas e entidades destinada ao apoio do software livre. Maddog vem ao Fisl desde as primeiras edições e significou um aval e uma imagem de relevância internacional ao evento. Suas palestras são reconhecidas como divertidas e entusiasmantes.

Ele vai exibindo seus slides, com um jeito calmo e um tanto sarcástico. “A pessoa mais importante do software livre é você”. “O sistema que muita gente chama de Linux o RMS chama de GNU/Linux” é um cutucão na eterna reprimenda de Richard Stallman. Embora a brincadeira tenha um fundo de verdade, é um tanto exagerada, principalmente no Brasil – não são poucos os que falam em

GNU/Linux, Stallman com certeza não está sozinho. Fala sobre o verão em que esteve em Florianópolis, citando um evento local. Elogia o churrasco e a bebida, enquanto alguém da plateia grita: “*banana power!*”. Maddog explica: “Isso é uma parte de banana e oito de rum”.

Surge da plateia o questionamento sobre porque Linus Torvalds não vem ao Brasil. O pedido se repete pelo menos desde o V Fisl, em 2004, quando Maddog gravou um vídeo com centenas de pessoas dizendo: “*Linus, we love you. Please come to Brazil*”. Das grandes personalidades internacionais do software livre, de fato, Torvalds é a que nunca veio ao Brasil. Maddog fala da fobia de Torvalds de falar em público. Conta que, em uma reunião com quarenta pessoas ele deixava a sala a todo tempo para vomitar.

Enquanto Maddog responde à brincadeira de alguém da plateia que o chamou de Papai Noel (pela barba branca e a barriga redonda), o mestre de cerimônias do evento, vestido de pinguim, desce as escadas dizendo “*I love Maddog*”. O visual de Maddog lembra bastante o de Richard Stallman, mas não se trata de imitação de um modelo e sim de uma padrão para uma mesma geração de programadores dos anos 1960 e 1970. Ainda com ligações tímidas com o ambiente corporativo, esses profissionais de barbas e cabelos longos, cresceram no ambiente universitário. Diferem dos da década de 1990, barbeados e de cabelos curtos, como Torvals e Eric Raymond.



Foto: Cristiano Sant'Anna/indicefoto.com

Mestre de cerimônias vestido de pinguim, com plateia ao fundo durante o encerramento do Fisl de 2008.

Alguém na plateia pergunta a ele, em inglês, sobre o futuro do “*open*”. Ele responde usando os dois termos, *free* e *open*, e ajusta seu discurso exatamente entre as duas categorias: faz uma crítica à natureza das empresas – “Eles não fazem um software melhor porque eles são uma companhia para o lucro, eles não colocam mais engenheiros para melhorar, pois isso custa” – e um elogio à melhoria técnica do software livre – “Quem faz um software melhor são os consumidores”.

Com o fim da apresentação de Maddog, o mestre de cerimônias vestido de pinguim sobe ao palco. Faz piadas, diz que “finalmente tem mulher neste evento”, distribui brindes de patrocinadores.

Mario Teza fala sobre as Sementes Livres. Um cheque de R\$ 17 mil é entregue ao representante dos quilombolas, que devem comprar um engenho de beneficiamento para o arroz que produzem. Teza é visto como um dos “políticos” do movimento, até por incentivar iniciativas como essa. Mas, como dito é objeto de respeito e reverência que parecem superar o estigma negativo do “político”. Embora

seja um ótimo articulador, não corre atrás de holofotes, algo que impossibilita com que seja lido como alguém que usa o software livre em benefício de uma agenda política que estaria fora dos limites do movimento. Teza trabalha nos bastidores e não projeta a si mesmo como um líder, embora o seja. Contudo, o elo entre software livre e a luta contra as patentes sobre a vida nunca ganhou força de fato em discussões no Fisl. A própria FSF, que se posiciona firmemente contra as patentes de software, já rechaçou uma aliança com outros movimentos que não se resume às bandeiras específicas do software livre.

O mesmo enredo de todo ano é repetido: fala-se do número de participantes (7 mil e 400, um salto, visto que nos últimos três anos o número manteve-se estável na casa dos 5 mil); os países presentes são citados, as caravanas, o número de empresas patrocinadoras. Quem fala é o coordenador da ASL, Sady Jacques. Em entrevista concedida por ele em 2007, a propósito do Fisl 8, para a revista *Computerworld*, é interessante o perfil de catalizador de negócios que Jacques dá ao Fisl, mesmo sendo ele, pessoalmente, alguém com ligações políticas fortes. Ao se posicionar publicamente como representante do Fisl, assume o software livre como um movimento, mas dá aos primeiros anos do evento que coordena um caráter de celebração e não de demanda política. Usa o termo amadurecimento para referir-se às mudanças do Fisl ao longo dos anos, reforçando a imagem de que, na idade adulta, deve haver uma relação equilibrada com o mundo dos negócios – e não de confrontação juvenil. As posturas críticas ao capitalismo dentro do movimento são classificadas pelo software proprietário e pelo grupo *open* como utopia não-realista, de pouca sustentação na vida adulta (Evangelista, 2005). Na fala de Jacques, o software livre (ou de código aberto, como ele também se refere) aparece como objeto de interesse de *players* do mercado. Estes se aproximam porque o desenvolvimento colaborativo oferece vantagens, porque o modelo de negócios mudou. Vale ressaltar que se trata de uma revista muito voltada para empresários e profissionais de informática, e Jacques parece usar a argumentação

pronta a esse público:

COMPUTERWORLD – Desde a primeira versão do Fórum Internacional de Software Livre, oito anos atrás, o que mudou para a edição atual?

Sady Jacques – Podemos dizer que houve um amadurecimento no processo. Iniciamos naquela época tentando criar um espaço de interlocução, um movimento que envolvia usuários, desenvolvedores, universitários, enfim, um conjunto de pessoas que estavam começando a desenvolver software livre e não tinha espaço mais organizado para fazer uma celebração. Esse espaço vem cumprindo a função desde então e, mais recentemente, vem procurando dar conta de uma série de demandas que o relacionamento com o conceito de software livre acaba construindo, como questões sobre o que fazer com o código desenvolvido e como torná-lo economicamente viável. Esse amadurecimento produz resultados práticos. Antes tínhamos em fase incipiente um sistema operacional para desktops e hoje temos uma série de opções. O código aberto se aprimorou, está mais competitivo. E é por essa competitividade que podemos conversar de forma mais objetiva sobre os resultados.

CW – Pode-se entender então que o fórum está mais profissional e tem sido encarado por muitas empresas como centro gerador de negócios?

Jacques – Acredito que sim. Para as grandes empresas as oportunidades nascem justamente da percepção desses movimentos de popularização do código aberto. A partir da demonstração de interesse do mercado. Por seu lado, esses players têm visto que cada vez mais o segmento de software livre se caracteriza como uma alternativa viável de negócios.

CW – Como você avalia a aproximação de empresas tradicionalmente conhecidas por serem avessas ao software livre com outras entusiastas do modelo, como no acordo da Microsoft com a Novell?

Jacques – É natural que haja uma percepção da importância que o software livre vem tendo nos negócios. Acredito que seja natural, da mesma forma, esse novo tipo de interlocução, embora isso não seja sinal de que essas empresas estejam concordando com o software livre. Acho que nesse momento as empresas estão começando a pensar nessa possibilidade, não no encerramento de um ciclo de um software para outro, mas uma visão de desenvolvimento mais colaborativo, que vai desembocar em um modelo de negócios focado nos serviços.

(...)” (http://computerworld.uol.com.br/mercado/2007/04/11/idgnoticia.2007-04-11.7594990900/IDGNoticiaPrint_view)

Jacques coloca uma interlocução possível entre empresas e software livre, porém sem que as empresas necessariamente concordem com o software livre. Embora o software livre possa ser útil a elas, há algo a se discordar – ou concordar. Falando a uma revista voltada aos negócios na área de

informática, Jacques, mesmo sendo um “político”, articula a imagem mais palatável possível ao gosto da publicação em que o software livre é sinônimo de novo modelo economicamente viável, mas deixa escapar que há, no software livre, algo que possa desagradar a, pelo menos, algumas empresas. E não poderia ser diferente, pois a inexistência de discordância (e de polarização com a Microsoft) significaria a equivalência do software livre a nada.

Em seguida ao comunicado de Jacques, é feito o anúncio das equipes vencedoras da “Arena de Programação”, quando se pode acompanhar mais dessa integração entre o Fisl e as empresas *open*. Diz o texto de anúncio da competição no site do Fisl:

“Imagine um grande aquário, e dentro dele, ao invés de peixes, programadores, programadoras, computadores, desafios e prêmios. Assim é a Arena de Programação do Fórum Internacional Software Livre (FISL). A Arena tem como missão promover o encontro de membros da comunidade tecnológica para uma disputa baseada em habilidades técnicas individuais e em grupo, e acontece antes, em fases remotas, e durante os dias do FISL, em uma sala-aquário com acesso restrito no meio do centro de eventos. São escolhidos projetos de Software Livre que são usados como estudo de caso para a Arena”.

A Arena é ideia recente, e teve sua primeira edição em 2007. Em 2008, foi realizada com o apoio da Nokia. A representante da empresa está no palco, e diz que “encontraram o que queriam: plataformas abertas”. Os grupos que participaram da Arena criaram um leitor de arquivos no padrão ODF para o *tablet* Maemo (espécie de computador portátil) da Nokia. O ODF é um padrão livre de arquivos. No início de 2008, o movimento software livre envolveu-se em uma intensa disputa com a Microsoft em torno de padrões de arquivos. A Microsoft desejava, e conseguiu, ver seu padrão OOXML de arquivos ser classificado como um padrão livre pelo sistema ISO (International Organization for Standardization), o que facilita a adoção de documentos produzidos por seus softwares por governos. O objetivo do movimento software livre era mostrar que o único padrão livre era o ODF, e que a negação dessa verdade era o objetivo de um intenso *lobby* corporativo. Empresas

como a Nokia se interessam por padrões alternativos aos controlados pela Microsoft, como o ODF, por conseguirem melhor acesso ao mercado a partir deles.

A Arena de Programação é uma iniciativa tipicamente ao gosto do público nerd, com fortes elementos da ideologia *open*: competição em princípio igualitária (pois todos possuem as mesmas informações para a tarefa); integração com as empresas; competidores postulantes ao mercado de trabalho; e demonstração pública de virtuosismo técnico.

Os prêmios aos vencedores são produtos da Nokia. Os grupos comemoram bastante. A representante da Nokia fala de um vídeo feito sobre a Arena de Programação que será exibido na Finlândia, no Nokia Fórum. Ela diz que a Nokia voltará o ano que vem, “com muito mais pessoas, com finlandeses, alguns famosos”. A referência é a Linus Torvalds, finlandês. Soa como blefe de alguém que pretende se mostrar poderoso, se lembrarmos da fobia de público que contou Maddog.

Com o fim da apresentação, Sady Jacques passa a falar do próximo Fórum Social Mundial, previsto para acontecer em Belém, no ano seguinte: “O Fisl convida participantes para construírem o Laboratório de Conhecimentos Livres, que foi referência em anos passados em debates sobre o conhecimento livre”⁶⁷. Ao citar o LabLivre, Jacques afirma que a iniciativa ocorre em conjunto com o Ponto de Cultura Minuano, projeto da ASL com apoio do governo federal⁶⁸.

Em seguida, Alexandre Oliva, principal representante da FSFLA (Free Software Foundation

⁶⁷ Estive pessoalmente envolvido com a construção do primeiro Laboratório de Conhecimentos Livres, em 2005. A partir da percepção de que era preciso fortalecer a ideia dos conhecimentos livres e estimular o debate sobre patentes e direitos autorais no FSM, tratava-se de um esforço de articulação entre iniciativas que comungavam desses questionamentos para que se encontrassem no FSM de Porto Alegre naquele ano. A iniciativa teve sucesso e ganhou repercussão, principalmente pela visita do então Ministro da Cultura Gilberto Gil e de autores reconhecidos como John Perry Barlow (da banda Grateful Dead) e do advogado Lawrence Lessig.

⁶⁸ O Pontão de Cultura Digital Minuano é um projeto da Associação Software Livre.Org em convênio com o Ministério da Cultura dentro do Programa Cultura Viva com a missão de promover o desenvolvimento humano sustentável através do compartilhamento de gestão e conhecimentos. Seu objetivo é promover o uso de ferramentas de comunicação e produção cultural em software livre para os integrantes dos Pontos de Cultura, Casas Brasil, Rádios Comunitárias, Escolas, Projetos de Economia Solidária, Movimentos Sociais e comunidades assemelhadas, possibilitando a autogestão de seus projetos e incentivando o trabalho colaborativo em redes. - Ver <http://www.minuano.org/?q=node/12>

Latin America), anuncia uma cartilha para crianças sobre DRM que traduziu. DRM significa *Digital Restrictions Management*, e refere-se a dispositivos de hardware e software que tentam impossibilitar o uso de material sob direito autoral sem autorização dos titulares. O problema é que esses dispositivos afetam também o compartilhamento de códigos livres, e a FSF e suas equivalentes continentais (FSF India, FSF Europa e FSFLA) promovem uma campanha de esclarecimento e protesto contra o DRM. Oliva pagou do próprio bolso 500 cópias da tradução da cartilha e as levou ao Fisl. Ele anuncia que a organização do Fisl gostou da iniciativa e que vai patrocinar mais 10 mil cópias impressas.

Nos últimos anos, a FSF tem caminhado no sentido de evitar laços políticos que a identifiquem com radicalismos à esquerda, enquanto aprofunda sua qualificação de qualquer item de software não-livre como anti-ético. Oliva personifica esses anos recentes da FSF. É um técnico, com formação universitária em computação, mas muito apegado a uma leitura estrita dos ideais fundadores do software livre, avesso à interpretação à esquerda feita por alguns. Já manifestou aversão a qualquer extrapolação dessas ideias e caminha com cuidado na associação entre software livre e políticas partidárias. Está entre o nerd e o ativista. Afirma que “as patentes são um problema social” e é missão da FSFLA “...levar à frente o software livre como movimento social”. Trabalha para a Red Hat, uma das primeiras empresas a comercializar uma distribuição de software livre, um sistema operacional completo. Por sua filiação profissional e sua reconhecida contribuição em termos de código ao software livre, Oliva nunca poderá ser classificado como um “político”, embora estabeleça ligações institucionais com uma organização mais simbólica do grupo *free* e melhor aceita pelos “políticos”, a FSF. Entre *free* e *open*, Oliva talvez seja o líder que mais emite sinais aparentemente confusos: suas palestras são cheias de referências nerd⁶⁹ (filmes de ficção científica, piadas com temas técnicos) e já

⁶⁹ Takhteyev (2007) fala sobre o sentimento de pertencimento de desenvolvedores brasileiros com uma cultura *nerd* ampla. Essa cultura incluiria, por exemplo, a familiaridade com uma grande variedade de jogos de computador e de heróis de histórias em quadrinhos. Kinney (1993) aponta que o nerd descrito por filmes e show de televisão para adolescentes é o jovem desengonçado, inteligente, tímido, não-atraente, socialmente marginalizado, com cabelo e roupas fora de moda.

manifestou publicamente profunda aversão à extrapolação dos ideais do software livre a outras lutas sociais; ao mesmo tempo, procura seguir estritamente os princípios da FSF, recusando-se a instalar e usar qualquer tipo de software proprietário em seu computador, e participa ativamente de campanhas públicas de pressão – como a movida contra órgão do governo, como a Receita Federal, que exige uso de software proprietário para o envio de arquivos de declaração de imposto de renda.

Em seguida, quem assume o microfone é Fernanda Weiden⁷⁰, filiada à Free Software Latin America e funcionária do Google. Ela fala do forte e recente patrocínio das empresas de mídia ao Fisl (Globo.com, UOL e Terra tiveram estandes). Cita em especial o contato que a organização do evento teve com a Globo e como a empresa de comunicação usa software livre em várias iniciativas, como na votação do Big Brother Brasil. A reação é péssima, surgem várias vozes reclamando. “Traz a Microsoft também”, grita alguém do palco. Mário Teza percebe que o tema foi apresentado de maneira equivocada e intervém. Assume o microfone e, em tom mais alto, diz que quando a Microsoft abrir seus códigos ela estará no Fisl. É vaiado. “Não queremos o dinheiro deles, eles podem nos dar todo dinheiro do mundo, mas quando abrirem o código poderão vir”, diz. Fica uma sensação de mal-estar, semelhante à réplica do vice-governador do Rio Grande do Sul, Feijó, ao governador do Paraná, Requião. A Microsoft foi construída, ao longo dos anos, como grande inimiga do software livre. E este como aliado dos movimentos sociais – mesmo que indiretamente, por simples associação, dada sua posição marginal em relação à fatia de mercado do software proprietário. Assim, o ambiente do software livre e do Fisl é significado como algo impróprio à Rede Globo, maior emissora do país e

⁷⁰ Fernanda Weiden é administradora de sistemas para o Google, em Zurique, Suíça. Membro do Projeto Software Livre -RS, participa da organização do Fisl desde os primeiros anos e é fundadora do Projeto Software Livre Mulheres. Assim como Oliva, Weiden mantém um posicionamento avesso à política, característico da FSFLA, da FSFE e dos anos mais recentes da própria FSF. Ela afirmou, em mensagem registrada por Murillo (2009): “Ligar o Software Livre com movimentos sociais diversos é coisa da cabeça de quem os liga. Os criadores do SL não pregam isso e não se interessam por isso. Eles querem que o SL dê certo. Se isso vai acontecer moral ou imoralmente na visão de outras pessoas, isso não é problema nosso [...] não sei de onde as pessoas tiram que o SL deveria ou alguma vez foi contra capitalismo [...] desculpe, mas pra mim o SL e especialmente a GPL é a tradução do capitalismo na tecnologia (Fernanda “Nanda” Weiden, 30/04/2008, lista ASL.org).

vista como parceira das restrições à liberdade ocorridas no regime militar. A impropriedade se agrava com a citação do *Big Brother Brasil*, programa de forte apelo popular e que consiste em câmeras em vigilância constante a participantes de um *reality show*.

Teza foi hábil em sua intervenção. Usou o respeito que tem para assumir a questão e defender um diálogo institucional dentro do FisI. Mas o mal-estar ainda persistiu, enquanto Jacques falava de um acordo com Cuba, recebido com palmas fracas. Não parece ser uma aversão ao acordo, mas um efeito residual do acontecido. Teza dá prosseguimento e fala então de uma cooperação com a Nasa para o FisI 10. O objetivo dos organizadores é atingir a marca de 10 mil inscritos na décima edição do evento. “Vai ter muita coisa louca, um *link* com a estação espacial. Tudo o que planejamos fazer em 2001 [primeira edição do FisI] faremos agora”. Nesse momento, o mal-estar parece já ter se dissipado. Não tanto pelas falas, mas porque o assunto parece já ter sido digerido.

Toda a organização sobe ao palco para uma foto com Maddog. Teza tenta puxar um coro: “10 mil no FisI! 10 mil no FisI!” Quase ninguém o acompanha e uma pessoa da plateia tira sarro: “Sozinho! Sozinho!”. Em anos anteriores, no próprio encerramento, Teza já puxou o coro “software livre! software livre!”, sendo acompanhado por um auditório lotado, num momento realmente catártico. Talvez as palavras de ordem de 2008 tenham sido complicadas e menos consensuais do que o tradicional “software livre”. Talvez seja um reflexo do mal-estar de minutos atrás. Talvez o momento do movimento não seja o mesmo. Provavelmente um pouco de tudo.

Conclusão

A distinção entre os grupos *free* e *open*, analisada no capítulo anterior, oferece a base para que

possamos entender os enfrentamentos, alianças e tomada de posições que acontecem no Fisl. Os símbolos vestidos e utilizados pelos participantes, por exemplo, originam-se em grande parte de grupos internacionais, que podem ser posicionados a partir dessa divisão.

No entanto, percebe-se que o discurso, em especial do grupo *free*, ganhou coloração própria quando reinterpretado por militantes brasileiros. A ideia de cooperação, colaboração, solidariedade e construção de um conjunto de softwares que fosse uma alternativa para o enrijecimento das regras de propriedade intelectual, ganhou outra força ao aportar em um país subdesenvolvido de industrialização parcial. Técnicos, muitos ligados ao serviço público, e com passado ligado aos movimentos de esquerda, entenderam o movimento software livre também como uma resposta ao domínio das grandes empresas de informática e ao saque de riquezas promovido pelos países desenvolvidos. No horizonte, enxergou-se o software livre até como fator de transformação e superação da economia capitalista.

Foi assim que políticos de alguma forma identificados com a ideia de resistência à dominação e exploração externa incorporaram o software livre em seu repertório de propostas, somando-o a planos de independência nacional. Setores discordantes sobre essa interpretação da origem do subdesenvolvimento brasileiro ou descartaram o software livre como algo viável, ou mobilizaram argumentos típicos do grupo *open*, apontando sua melhor qualidade técnica como derivada do processo aberto de produção. Foi o que pudemos acompanhar no embate ocorrido na abertura do evento, com Roberto Requião defendendo o software livre como alternativa “à dependência de sistemas proprietários” e o vice-governador Feijó contra-argumentando em favor de uma decisão técnica e pela convivência entre o modelo livre e o proprietário. Se submetida a entidades internacionais que agrupam membro do grupo *free*, como a Free Software Foundation, a fala de Requião não seria tomada como própria, mas sim como uma extrapolação indevida do que são os objetivos do movimento software livre. Porém, no contexto do Fisl, é entre os simpatizantes do grupo *free* que esse tipo de interpretação

ganha apoio. Requião conecta, de modo que chega a ser caricato, a causa do software livre à ideia de independência das potências estrangeira, desenvolvimento local autônomo e distribuição justa e equânime da renda e da propriedade. Igualmente – embora tenha sido, na ocasião, talvez enfático demais – o tipo de discurso pragmático e que não se opõem ao mercado é o mesmo que é sustentado por simpatizantes do grupo *open*.

A cerimônia de abertura do Fisl, além de ter sido emblemática por trazer esse enfrentamento entre *free* e *open* à moda brasileira, em seus discursos mostra como o uso de certas palavras representa a afirmação de filiação dos sujeitos a esses grupos. Os falantes com menor experiência, políticos com contato lateral com o movimento, alternam termos como “código aberto”, “software aberto”, “código livre”, como se estes fossem sinônimos; enquanto os membros do movimento usam termos “open source” ou “GNU/Linux” como marcas de filiação de suas ideias. O mesmo ocorre com as imagens, usadas à exaustão em camisetas e cartazes no Fisl, e que significam apoio geral à causa do software livre, mas que marcam fortemente o apoio aos variados grupos, entre os quais a distinção é política.

Vimos também que o movimento software livre, em especial o grupo *free*, requer de seus membros uma certa pureza, uma adequação entre defender o software livre com argumentos teóricos e extirpar da vida cotidiana o software proprietário. O palestrante do Fisl que utiliza software proprietário em sua apresentação é logo desacreditado pelo público. O membro do movimento que usa o sistema operacional Windows durante o evento – ou mesmo apenas o mantém instalado, em um setor separado, no disco de seu computador – é censurado pelos companheiros. Se a organização do evento utiliza um arquivo de vídeo em formato proprietário, acaba sendo objeto de crítica pelo público. Os participantes procuram até mesmo retirar o adesivo do sistema Microsoft Windows que vem colado na maioria dos notebooks, substituindo-o por diversos adesivos alusivos ao software livre. É por possuir um conjunto até certo ponto estrito de regras de comportamento que podemos ouvir, nos corredores do

Fisl, frases como: “Esse aí traiu/não traiu o movimento”. Esse conjunto de restrições e recomendações funcionam de modo a, por um lado, estabelecerem as divisões entre aqueles que pertencem – e dialogam com essas normas – ao movimento e aqueles que não pertencem. Ao mesmo tempo, o modo como são interpretadas – se são tidas como “exageradas” ou pertinentes – indica posicionamentos em relação às diversas subdivisões existentes.

Essa cobrança por integridade, por comprometimento, está presente também no encerramento, quando o anúncio da presença da Rede Globo causa reação negativa no público. A empresa de comunicação, por seu histórico, mas também por sua posição dominadora no setor de comunicações, é associada ao gigante da indústria da informática e empresa inimiga do movimento, a Microsoft. Não é o que acontece com outras empresas, ascendentes e com uma imagem inovadora, como o Google e a Nokia. A empresa finlandesa, aliada do movimento em causas como a do padrão livre de arquivos ODF, é recebida com naturalidade, sendo sua presença até mesmo um sinal de que o evento não é anti-capitalista. Não apenas as pessoas são interpretadas por sua relação favorável ou contrária ao software livre, mas também empresas, lidas como sujeitos que escolhem entre o bem e o mal (aqui cabe lembrar o slogan “Don't be evil” do Google)⁷¹. Com um histórico associado ao regime autoritário, distante de práticas de democratização da informação e de transparência, a Rede Globo é vista como oposta aos ideais do software livre, importando pouco o quanto ela de fato usa de códigos livre e com ele contribui.

Ao mesmo tempo, um evento visto por muitos como radical, como o Fórum Social Mundial, é anunciado no mesmo palco, na mesma cerimônia, separados apenas por alguns minutos. A abertura é o momento em que o Fisl, de uma maneira geral, integra melhor esses aspectos aparentemente

⁷¹ A frase “Don't be evil” é citada frequentemente, em listas de discussão, como slogan corporativo do Google quando as atitudes da empresa estão em questão. A frase consta no prefácio do Código de Conduta da empresa disponível em <http://investor.google.com/conduct.html> Acessado em 04/08/2009.

contraditórios. O faz apoiando-se em figuras que consigam transitar, sendo respeitadas, pelos dois grupos principais, *free* e *open*. John Maddog Hall, por exemplo, ao mesmo tempo que ostenta um visual semelhante ao de Stallman e faz críticas à natureza das empresas, é próximo de Linus Torvalds e atua próximo ao mundo corporativo. Outras figuras mostram o mesmo trânsito entre os grupos como Alexandre Oliva, Mario Teza e Sady Jacques. Jacques, embora seja ligado a movimentos sociais, é capaz de utilizar um discurso próprio a uma revista como a *Computerworld* para promover o Fisl, enfatizando o papel de catalisador de negócios do evento. Teza, outro organizador do evento próximo aos setores políticos e aos movimentos sociais, assume uma postura sempre conciliadora, objetiva e pragmática na condução das coisas do Fisl. Embora trabalhe pela sua visão sobre a importância do software livre, procura não confrontá-la com iniciativas de outra natureza; de certa forma sua visão e projetos competem livremente com outros de uma maneira não destrutiva, procedimento que se encaixa com as ideias gerais do movimento. Já Alexandre Oliva, embora trabalhe para uma grande empresa e demonstre reservas à aliança com outros movimentos sociais, assume uma postura pública de crítica radical ao uso cotidiano software não-livre, muito semelhante à de Richard Stallman. É, por isso, respeitado pelo grupo *free*. Ao mesmo tempo, sua capacidade técnica como desenvolvedor, seu conhecimento da computação e suas restrições às interpretações do movimento software livre como em alguma medida anti-capitalista lhe rende o trânsito com grupo *open*. Além disso, ressalte-se que a performance pública de Oliva e Hall, por exemplo, que usam de um determinado tipo de humor ao falarem em público – auto-irônico, nonsense e recheado de piadas internas - os aproxima do gosto geral da audiência do evento.

Embora seja inequívoca a existência de dois grupos no movimento software livre, a cerimônia de encerramento deixa claro como há unidade entre eles. O compartilhamento desse mesmo tipo de humor, presente nas performances públicas de Oliva e Hall e em várias situações acompanhadas

durante o evento, sinaliza isso. Assim como a valorização – em variados graus de intensidade – da tecnologia como capaz de resolver variados problemas sociais. Os membros do movimento, sejam eles do grupo *open* ou *free*, compartilham da ideia de progresso técnico da humanidade e tem, em geral, visões otimistas sobre o impacto da tecnologia na sociedade. No máximo, o que há são restrições ao que seria como o tipo errado de tecnologia, fechadas. Porém, as tecnologias tidas como livres, como o software livre e a internet, seriam democratizadoras e promotoras de uma evolução qualitativa da humanidade e do ambiente social.

No capítulo anterior, discuti o texto de Žižek, que fala do “capitalismo sem fricção”, fruto da síntese das ideias do Fórum Social Mundial, de Porto Alegre, e do Fórum Econômico Mundial, de Davos. O Fisl, integrando progressivamente *free* e *open*, e apresentando uma alternativa tecnológica cada vez mais pertencente ao capitalismo, tem sua unidade também alicerçada nessa síntese, na ideia de que é preciso haver foco no desenvolvimento tecnológico, acelerando-o. Os dez mandamentos do liberal-comunismo, citados por Žižek, incluem valores básicos ao movimento software livre como tem se desenhado no Fisl. Entre eles estão o modelo de capitalismo baseado na prestação de serviços, a transparência, a educação permanente e o trabalho flexível que, para existir, deve se misturar com o lazer. No próximo capítulo, ao examinar o crescimento do grupo *open* e as novas conexões do movimento com o mercado, esses pontos ficarão mais evidentes.

Cap. 4. Nerds e geeks

No capítulo anterior, argumentei sobre a relação entre *free* e *open* no contexto do Fórum Internacional de Software Livre, apontando como esta acaba funcionando como condutora dos principais conflitos políticos do evento. Ao longo dos quase dez anos em que o Fisl é realizado, a dinâmica dessa disputa vem sofrendo alterações. Como já dito anteriormente, os embates entre os grupos estão diretamente relacionados à maior valorização em termos de mercado, mas também em termos políticos, de determinados projetos de software. Ao palestrarem em eventos como o Fisl e debaterem publicamente a partir de sua visão sobre o software livre, os líderes dos projetos e dos grupos políticos buscam apoio dos entusiastas, competindo por eles enquanto militantes, mas também como possíveis trabalhadores dedicados ao desenvolvimento, teste e melhoria dos softwares.

Mudanças no mercado de tecnologia da informação, com o fortalecimento das empresas que baseiam seu negócio na venda de serviços e não de licenças de software, têm levado a uma crescente presença destas em eventos como o Fisl. O que, em tese, significaria uma “vitória” do movimento, dado ser esse modelo de negócio – a venda de serviços – o defendido pela maioria de seus membros como “mais justo”, leva também a transformações no próprio movimento. A ideia, aqui, não é fazer uma análise histórica, nem exaustiva, mas mostrar como empresas como Nokia e Google, entre outras, presentes na nona edição do Fisl, representam um pólo de atração para novos membros do movimento

software livre, que passam a integrá-lo a partir de motivações que são mais direta e explicitamente profissionais, individualistas e fundamentalmente ligadas ao grupo *open*. Não se trata de, ingenuamente, assumir a inexistência dessas motivações antes da presença mais forte dessas empresas no evento, mas de apontar para o reforço de uma certa perspectiva sobre qual o propósito, a utilidade, o valor e o papel do software livre no cotidiano, no mercado e como movimento social, perspectiva essa que se fortalece em detrimento de outras.

Pretendo discutir, a partir de situações que presenciei no 9º Fisl, um pouco do discurso envolvido na competição por colaboradores/apoiadores, assim como busco afirmar a existência dessa competição em si. Persigo mais especificamente uma determinada categoria de público do evento: os nerds. Nos últimos anos, têm se tornado mais frequentes e intensas as expressões de auto-identificação com esse termo por certa parte do público. E é justamente esse público que parece estar sendo atraído ao Fisl pela presença das empresas que transmitem uma imagem *open*, bem como pela possibilidade de adquirirem conhecimentos e relações pessoais que levem esses indivíduos a, no futuro, estabelecerem relações de trabalho com as grandes companhias líderes de mercado.

É também a partir dos nerds que exploro o que se poderia entender como “o outro lado” desse mesmo processo de tentativa de arregimentação de colaboração técnica e apoio público. Figura símbolo do grupo *free*, Richard Stallman, em diversas falas, é explícito ao afirmar que o reconhecimento público – seja da qualidade e engenhosidade de projetos de software desenvolvidos, seja da contribuição política ao movimento – leva a uma maior facilidade na incorporação de mais força de trabalho para o desenvolvimento de novos projetos de software. Stallman, pela sua trajetória de vida e imagem pública, funciona como exemplo negativo para a carreira profissional dos nerds. É por essa percepção da figura de Stallman – em parte compartilhada por outras categorias de público do evento – que procuro estabelecer uma distinção com o perfil de comportamento profissional que é ofertado, e às

vezes imposto, aos nerds. Aqui, não discuto caminhos profissionais reais que seriam oferecidos aos nerds, mas procuro entender a oposição que é oferecida simbolicamente aos indivíduos como sendo exemplar do grupo *free* e do grupo *open*, mesmo que a atribuição dessa diferença aos grupos seja, quando muito, apenas eventual.

Vale então, neste momento, buscar elementos que ajudem a compor a imagem pública do próprio Stallman. Ela é importante não só para se perceber esse jogo que envolve o reconhecimento da contribuição ao software livre como um todo mas também para se ter indícios acerca do que “diz” o grupo *free* sobre o futuro profissional dos militantes do movimento. No Manifesto GNU, por exemplo, Stallman já se vê levado a responder a questão "Os programadores não irão morrer de fome?"⁷², colocada a ele por aqueles que duidavam não só da viabilidade prática da ideia de software livre como questionavam sobre que perspectiva profissional se ofereceria aos programadores em um mundo futuro, com amplo uso de softwares livres. Escreve Stallman:

“O motivo pelo qual os programadores não irão morrer de fome é que ainda será possível para eles serem pagos para programar; somente não tão bem pagos como o são hoje. (...) Provavelmente a programação não será tão lucrativa nas novas bases como ela é agora. Mas este não é um argumento contra a mudança. Não é considerado uma injustiça que caixas de lojas tenham os salários que eles tem hoje. Se com os programadores acontecer o mesmo, também não será uma injustiça.” (Stallman, 1985)

Nesse texto e em sua prática de vida, Stallman oferece a perspectiva de ganhos financeiros humildes e de uma banalização das habilidades profissionais dos programadores, algo bastante diferente do que parece ser valorizado pelos nerds e que claramente se opõe ao que lhes é oferecido pelas grandes empresas. Em lugar da imagem de um profissional com status social, com ganhos que o coloquem no topo da pirâmide econômica e habilidades especiais, posto que raras e alcançáveis por

⁷² Nessa mesma resposta, Stallman afirma que o software livre leva a uma outra organização do modelo de negócios em torno do software, sem porém se deter em como seria esse modelo.

poucos, Stallman compara-os, ainda que indiretamente, a caixas de lojas que, se não “morrem de fome”, não são profissionais bem pagos.

Stallman é a figura mais ativa, desde os primeiros anos do movimento, a exercer o papel de recrutador de colaboradores/apoiadores para os projetos de software tomados como prioritários pela Free Software Foundation. A figura pessoal de Stallman é vista, pelos nerds, de maneira até certo ponto contraditória. É, por vezes, ironizado, sendo tratado em certas situações com desdém. Mas há também admiração e respeito. Isso acontece, como veremos, pelos feitos técnicos e intelectuais de Stallman ao longo de sua vida, que se somam aos seu comportamento público, em alguns momentos tomado como inconveniente, e ao seu posicionamento político pouco conciliador. Embora Stallman tome frequentemente atitudes tidas como excêntricas, estas são interpretadas como manifestações de seu espírito hacker misturado com a contra-cultura dos anos 1970.

Richard Stallman: de líder a motivo de piada

Richard Stallman é, certamente, a figura mais relevante para o software livre enquanto movimento social. Foi ele quem criou o termo, deu as bases para as outras licenças, tendo escrito a mais importante delas, a GPL, e iniciou o esforço coletivo para a construção de um sistema operacional que fosse totalmente livre. Linus Torvalds, o criador do *kernel* Linux, é uma figura mais conhecida publicamente. Stallman esteve na primeira edição do Fisl, em 2000, e a excursão pelo Brasil que realizou naquele ano teve papel importante na divulgação de sua visão particular sobre o software livre. Esteve em Porto Alegre e no Fisl em outros anos, sempre gerando polêmica e atraindo atenção pública. Ausente em 2008, mesmo assim é um personagem referencial para comportamentos, atitudes,

linguagem, visual e ideias.

Já no avião que me levou a Porto Alegre pude ouvir comentários sobre Stallman. Perto de meu assento, na fileira ao lado, sentam-se três pessoas que fazem o mesmo trajeto que o meu, de Campinas a Porto Alegre. Aparentam ser estudantes, são todos homens com menos de 25 anos, e conversaram a viagem toda sobre assuntos do Fisl. A eles juntam-se mais três, também homens jovens e, ao que tudo indica, estudantes. Um, aparentemente mais velho, fala de uma recente visita de Richard Stallman à USP. Conta, eufórico, das excentricidades do líder da Free Software Foundation. Diz que Stallman queria hospedar-se em um apartamento que tivesse um determinado tipo de papagaio e que só comia acompanhado por, no máximo, três outras pessoas. "Virou estrela!", retruca um dos estudantes, enquanto outro fala da diferença de atitude de Stallman hoje e no passado. "Bom era a época que ele implementava", aponta um terceiro, referindo-se a quando Stallman se dedicava mais aos trabalhos técnicos.

Em seguida, é lembrado o episódio da cobrança por autógrafos, ocorrido no Fisl 7. Incomodado com o assédio dos participantes do Fisl 7, que lhe pediam autógrafos e poses para fotos, Richard Stallman, na ocasião, passou a pedir contribuições à sua entidade em troca de seu autógrafo, inclusive fixando um preço. A atitude gerou revolta em muitos participantes, com comentários críticos crescentes. Um grupo, então, resolveu ironizar a atitude e promoveu o leilão de um antigo autógrafo do presidente da FSF, sendo a quantia arrecadada entregue diretamente a ele. Acompanhei o desenrolar do episódio pois, na época, colaborava com um website chamado *Cobertura Wiki*, criado para reunir relatos de participantes do Fisl 7 em textos escritos coletivamente na Internet. Um dos textos do site, que foi objeto de sucessivas edições por diferentes participantes, descreve o episódio. É importante frisar que, dentre as diversas correntes políticas do movimento, as pessoas que colaboraram com a *Cobertura Wiki* tendiam a serem mais simpáticas do que críticas a Stallman, à FSF e ao grupo *free*.

"Leilão de autógrafo é levado na esportiva por Stallman"

Em atitude contra a cobrança, por Richard Stallman, de dinheiro em troca de fotos e autógrafos, manifestantes promoveram, na tarde de sábado, 22 de abril, leilão de assinatura dada pelo fundador da FSF a Leonardo Vaz, do Openbsb-RS. A peça caligráfica foi arrematada por R\$ 33,00 e um saco de moedas (0,01; 0,05; 0,10; 0,25...), num total de R\$ 30,33 -- quantia que foi dada, pessoalmente, ao próprio Stallman no estande da FSFLA [Free Software Foundation Latino América], ao som de "Glória, Glória, Aleluia!". Os manifestantes entregaram, também, a peça de autógrafo arrematada, com a sugestão de que continue sendo usado em leilões.

Confrontado com o mau humor dos manifestantes, Richard Stallman pediu desculpas, e assegurou repensar seu comportamento."⁷³

Desde o ano 2000, Richard Stallman tem feito visitas regulares ao Brasil, nunca hospedando-se em hotéis, mas sempre na casa de algum responsável pelo evento que participa. Sua dificuldade de relacionamento pessoal deu origem a muitas histórias, que são aumentadas e se modificam quando repassadas. De um de seus hospedeiros, ouvi reclamações sobre a falta de higiene de Stallman, que comia com as mãos e teclava em seu inseparável *notebook* com os dedos engordurados, vez ou outra colocando na boca as pontas de seus longos cabelos. De outra pessoa que o recebeu, ouvi que Stallman é alguém difícil, que nunca puxa uma conversa e que não procura ser simpático. Quando, em listas de discussão, o assunto é a personalidade de Stallman, é frequente alguém referir-se a um leve autismo que o acometeria. Ao mesmo tempo, quando o entrevistei, ele portou-se como um perfeito entrevistado, falando pausadamente e preocupando-se em deixar claras suas ideias mesmo a jornalistas pouco experientes no assunto. Ele definitivamente domina os códigos tradicionais de comportamento e, quando não o faz, é por sua própria vontade.

Dentro do movimento, assim como no senso comum, é forte a percepção de que grande inteligência – ou mesmo genialidade – e um certo grau de irreverência e excentricidade são atributos relacionados. O próprio Stallman, de certa forma, reforça essa imagem excêntrica em suas

⁷³ 2006. Disponível em <http://wiki.softwarelivre.org/CoberturaWiki/Post20060422183048>. Acessado em 19/01/2010.

apresentações já há muito tempo. Em muitas de suas falas públicas em defesa do software livre, Stallman executa uma performance em que se veste como um sacerdote (veja imagem abaixo), chama a si mesmo de Santo iGNUcius e exorciza o software proprietário de computadores. Ao mesmo tempo que ironiza o rótulo de radical que lhe é imposto (o software livre sendo comparado a um movimento religioso fundamentalista), a performance reforça o rótulo.



Na sua palestra mais tradicional, antes de colocar o halo em sua cabeça, Stallman explica suas divergências com as outras correntes políticas do software livre. Ele repetiu partes dessa palestra em muitas oportunidades no Brasil, entre elas em algumas edições do Fisl e na segunda edição do Fórum Social Mundial. É uma apresentação preparada para aqueles que tomam o primeiro contato com o movimento. Destaco alguns trechos de uma de suas falas pelo mundo, esta realizada na Australian National University, em abril de 2004⁷⁴.

⁷⁴ Transcrição disponível em <http://hi.baidu.com/techofchaos/blog/item/9324202ef5d296301e308925.html> Acessado em 19/10/2010.

Inicialmente ele fala sobre a relação entre GNU e Linux, enfatiza que Linux é apenas o kernel e pede crédito pelo trabalho do grupo GNU afirmando, como vimos no capítulo anterior, o caráter político do trabalho desse grupo e do caráter “apolítico” do Linux e de Linus Torvalds. Ao adotar uma “filosofia apolítica” e ao considerar todas as licenças igualmente legítimas, Torvalds seria equivalente à Microsoft.

"After there was a complete GNU system with Linux that you could get to run, people started thinking that it was Linux. But before that point, our software spread the philosophy and our philosophy help spread the software because when the people read this, if they agree, they will be motivated to develop more free software and add to GNU.

However after people started using essentially the GNU system with Linux added, and called it Linux, it no longer led then to the philosophy associated with GNU – the philosophy of free software. Instead of that, the people read the philosophy that was associated with the name Linux. **The apolitical philosophy of Linus Torvalds who thinks that all software licences are legitimate and it is wrong ever to violate them. So his views on this are more or less the same as Microsoft's.** Now he of course has the right to promote his views but I object to our work becoming the main basis for promoting his views because it is attributed to him directly by labeling the GNU system as Linux. And that is why I ask people to call the system as GNU/Linux. Give us equal mention. We need it. **We need it not just because it is fair but because it will help people recognize what we have done so they will think about what we are asking them help us do.** Our work is not finished. People will sometimes give me advice which in other circumstances might have been wise. They would say, it looks bad to ask for credit. And so they say, when the people call the system Linux, smile to yourself and take pride in a job well done. This would be very wise advice if it were true that the job is done. We made a great beginning. We have developed more than one free operating system in our community and many free application programs. **But there are many application programs we still have to develop.** We have developed free operating systems used by 10's of millions of users. But there are 100's of millions of users of proprietary operating systems and even the people using free operating systems often use proprietary programs on top of that. So we have a tremendous amount of work to do."

No parágrafo acima, ele menciona um ponto importante: o reconhecimento é um caminho necessário para que mais pessoas contribuam com o software livre. O trabalho ainda não está

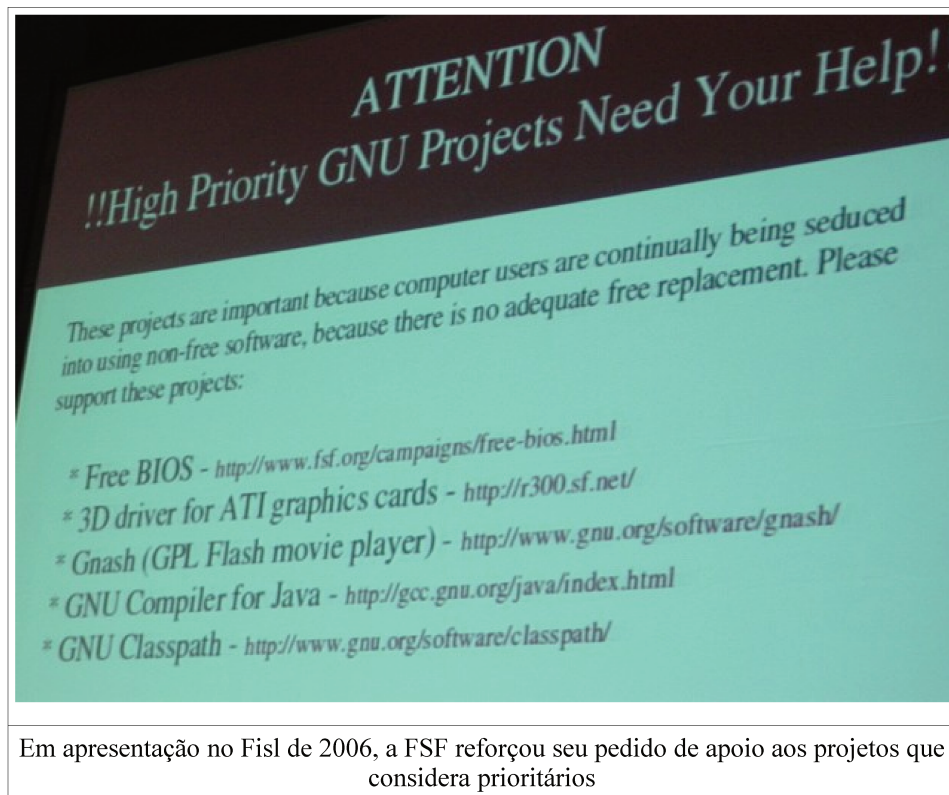
terminado, ainda existem muitos usuários de softwares proprietários e, mesmo entre os usuários de software livre, alguns ainda usam alguns softwares proprietários. Aqui fica claro como o reconhecimento leva também a uma facilidade no recrutamento de trabalho para projetos específicos.

"(...)

The use of flash websites is a major problem for our community. People are working in free software for playing flash. And now it more or less handles just the display of things but it doesn't handle reading input. If you see a website using flash, complain. Complain to the site developer saying you are excluding people who believe in maintaining their freedom. Please get rid of the flash from your site."

Acima, ele questiona os websites em flash, formato de arquivo de propriedade da empresa Adobe/Macromedia. Em 2004, ainda não havia sido criado o YouTube, website de vídeos que surge em 2005 e que popularizou definitivamente o flash, tornando-o o formato mais usado para a exibição de vídeos pela Internet. Mas já nessa época, Stallman demonstra sua preocupação com o formato, que necessita de um software proprietário para a criação de seus arquivos, e de um outro software proprietário (um *plug in*), para que seja corretamente visualizado nos navegadores da Internet. A FSF trabalha desde dezembro de 2005 no projeto Gnash, que objetiva a criação de um tocador (*plug in*) livre para para os arquivos em *flash*. O projeto Gnash está no topo da lista de prioridades da FSF e, de acordo com informações do site da FSF em outubro de 2008, é capaz de executar corretamente vídeos do YouTube. Contudo, ainda necessita de desenvolvimento, pois não é capaz de visualizar corretamente os arquivos criados para as versões mais atuais do *plug in* (versões 8 e 9). Esse é o problema mais frequente dos projetos de software livre que pretendem criar programas de visualização de arquivos em formato proprietário: os donos do formato criam constantemente versões mais novas a partir de especificações que mantêm secretas. Os projetos livres levam algum tempo para descobrirem/adivinharem essas especificações e, conseqüentemente, para atualizarem seus softwares e

para fazê-los funcionar corretamente.



ATTENTION
!!High Priority GNU Projects Need Your Help!!

These projects are important because computer users are continually being seduced into using non-free software, because there is no adequate free replacement. Please support these projects:

- * Free BIOS - <http://www.fsf.org/campaigns/free-bios.html>
- * 3D driver for ATI graphics cards - <http://r300.sf.net/>
- * Gnash (GPL Flash movie player) - <http://www.gnu.org/software/gnash/>
- * GNU Compiler for Java - <http://gcc.gnu.org/java/index.html>
- * GNU Classpath - <http://www.gnu.org/software/classpath/>

Em apresentação no Fisl de 2006, a FSF reforçou seu pedido de apoio aos projetos que considera prioritários

"(...)

... In 1998, some of them started another way of talking about free software where they call it opensource. **And with this different name, they have associated a different set of ideas.** They don't say that this is a matter of the freedom that every user should have. In fact, they would often say that they recommend a development methodology which they say will generally produce more powerful and reliable software. And that may be true. I hope it is true. It would be nice if freedom provides as a byproduct, security of software. But it is a terrible mistake, I think, to focus all the attention on these short term practical benefits and ignore freedom itself. The danger is, then people would fail to defend their freedom when it is threatened as they wouldn't recognize what it is. So if you imagine two people, one who is convinced by the opensource philosophy and another who is convinced by the free software philosophy. And you show these people a powerful, reliable, convenient, non-free program. What are they going to say ?

The opensource guy would say - "I am surprised you were able to do such a good job without letting the users study the code and find the bugs for you. But I can't argue with the facts. It seems a powerful and reliable program". And he will probably use it. Where as the free software person will say - "I don't care how powerful and convenient it is if it takes away my freedom. I wouldn't pay

such a high price for that convenience. I am going to get to work on a free replacement for this program right away before anybody else get tempted to use that program".

One person would give up his freedom when ever you can offer him convenience in its return and the other would fight for his freedom. And if enough of you fight for your freedom, freedom may prevail."

No trecho acima, Stallman afirma sua divergência e da Free Software Foundation, nos moldes do que vimos no capítulo anterior, com o grupo *open source*. Há um ponto a se ressaltar nessa divergência, que se liga à priorização da FSF ao projeto Gnash. Somente alguém do *free software*, que toma o software proprietário como essencialmente anti-ético, pode apontar o Gnash como projeto prioritário. A Adobe, empresa que produz o *plug in* do flash, oferece gratuitamente um *plug in* compatível com os sistemas livres, e que funciona razoavelmente bem (melhor que o Gnash). Esse *plug in*, porém, é um software proprietário. Para o *open source*, não é o caso de se dispender grandes esforços em torno de um software alternativo. Se essa alternativa existir, ótimo, possivelmente esse software, dado seu processo de desenvolvimento livre e de possível evolução, será de melhor qualidade. Mas enquanto tal alternativa não existe, que pragmaticamente se use a proprietária. Já para o *free*, o uso de tal programa não-livre, que satisfaz a necessidade imediata dos usuários e assim debilita seu impulso de buscar uma alternativa ou de fazer pressão por uma solução livre, é algo ruim.

Está clara na fala de Stallman sua oposição à associação que o grupo *open* fez entre os softwares livres e um conjunto diferente de ideias. E o exemplo que ele dá, mostrando como reagiriam de forma diferente duas pessoas convencidas ideologicamente pelas duas ideias distintas, retrata bem o efeito prático dessa divergência. Stallman não espera somente que o *free* se recuse a usar programas que não sejam livres, mas vê neles um possível desenvolvedor de uma alternativa.

"(...)

Now people sometimes have accused me of having a holier than thou attitude. I think that is not actually true. I don't criticise and condemn people just because

they don't stand up for free software strongly as I do. As long as what they are doing is good, I will say what they are doing is good and I might suggest somethings they could do.

However, I do have a holy attitude because I am a saint. It is my job to be holy. I am saint iGNUcius of the church of Emacs. I bless your computer my child. Emacs started out as a text editor which became a way of life for many users because it could do almost everything without exiting Emacs. And ultimately a religion. We even have a great schism between two rival versions of Emacs. And now we have a saint too. Fortunately no gods. In this church, instead of gods, we have an editor.

To be a member of the church of Emacs, you must recite the confession of the faith. You must say, there is no system but GNU and Linux is one of its kernels. The church of Emacs has certain advantages compared with some other churches. To become a saint in the church of Emacs does not require celibacy. However, it does require living a life of moral purity. You must exorcise the evil proprietary operating systems that posses what ever of the computers under your control and install in all of them a holy free operating system instead. And then, only install free software on top of that. If you make this commitment to live by it, then you too would be a saint and you may eventually have a halo if you can find one because they don't make them any more.

Sometimes, people ask me if it is a sin in the church of Emacs to use the editor Vi. It is true that Vi-Vi-Vi is the editor of the beast. But using a free version of Vi is not a sin but a penance. And sometimes, people ask me if my halo is really an old computer disc. This is not a computer disc. It is my halo. But it was a computer disc in a previous existence."

O comportamento excêntrico e até um pouco anti-social é bastante bem aceito na comunidade.

Com frequência, é associado e naturalizado como um comportamento típico dos aficionados em computação e ciências exatas, do *nerd* e do *geek*. Há uma valorização daquele que é marginalizado (como esquisito, pouco social) e da falta de esforço em se socializar; como se o mundo exterior ao grupo não importasse, mas também uma atitude de recusa em reação a essa marginalização. Um dos efeitos reativos à marginalização é esse humor somente acessível aos pertencentes ao grupo. O trecho acima é povoado de piadas nesse estilo, em que boa parte da graça advém do reconhecimento das referências feitas. Emacs é um editor de texto voltado a programadores e a aqueles que escrevem documentos, cuja primeira versão foi desenvolvida por Stallman. Vi é um editor de textos igualmente livre e também bastante popular. Apontá-lo como concorrente do Emacs seria algo impreciso, se

imaginarmos nessa concorrência algum tipo de disputa de mercado. Mas os usuários de ambos jocosamente simulam grandes divergências. Ao dizer “Vi-Vi-Vi is the editor of the beast” Stallman imita um refrão da música “The number of the beast”, do grupo de heavy metal Iron Maiden.

Mas nesse trecho, há também um interessante brincadeira com o papel que o próprio Stallman desempenha. Colocando-se como “santo”, ele admite o caráter extraordinário de sua própria posição: aquele que nunca usa, em hipótese alguma, software proprietário. Outros podem caminhar no sentido de atingirem também tal posição e, ao se aproximarem dela, o “santo” dirá que algo bom está sendo feito; ao se afastarem, o “santo” tentará reconduzi-los ao caminho. É uma relativização de seu próprio papel como “radical” e um assumir de sua posição de liderança.

Stallman, hoje, dedica-se completamente ao movimento. Não tem emprego e não se envolve mais, pessoalmente, no desenvolvimento de software. Procura manter algo como uma distância segura de outros movimento sociais, principalmente quando fala em nome da FSF. Mas em seus escritos pessoais não hesita em assumir suas posições políticas de esquerda, embora acabe dedicando bastante tempo a responder as acusações de que é comunista. Embora seja alvo de zombaria e desconsideração em suas posições políticas por parte do *open*, Stallman tem o respeito daqueles que se declaram mais técnicos, dadas as demonstrações de seu talento como desenvolvedor e por sua história na computação. Não fosse esse passado, possivelmente não teria prestígio algum. O que lhe autoriza como líder do movimento – mesmo que seja apenas como alguém a se discordar, contra a quem se posicionar – é, além de seu trabalho demonstrado, o fato de que alguns podem se identificar com ele como símbolo de uma opção de vida, ainda que difícil. Stallman optou por permanecer alguém não integrado, sem emprego que lhe oferecesse grande renda, família ou filhos, dedicando seu tempo a defender uma opção de vida e um mundo particular: entreter-se com códigos e problemas que requerem soluções inteligentes, ser um *hacker*. Mesmo que os ideólogos do *open* não queiram esse estilo de vida em sua

totalidade para si, é algo que são capazes de entender e que pode até fasciná-los, mesmo que rejeitem.

No caminho, os nerds

Para se entender melhor a figura de Stallman e o público do Fisl de uma maneira geral é preciso discorrer um pouco mais sobre a categoria de público que tenho chamado de nerds. Estudantes, jovens na faixa dos 17 aos 25 anos, formam o grosso do público do evento, algo que se percebe claramente ao andar pelos corredores. Eles circulam, principalmente, pelas palestras ditas “técnicas”, onde buscam aprender sobre softwares ou linguagens de programação. Isso não significa, certamente, dizer que permanecem alheios a qualquer conteúdo que não seja esse, mas é bastante claro como formam o público principal das palestras que falam diretamente sobre tecnologia (como ela funciona, como operá-la). Esses estudantes muitas vezes também estão prestes a entrarem no mercado profissional, ou desejam nele progredir. Nesse sentido, vale citar também as sessões que não são técnicas, mas ao mesmo tempo também não se encaixam necessariamente no que é chamado de “filosofia”. São palestras que discutem “como viver de software livre”, que debatem desde a sustentabilidade de um modelo de economia com forte peso do software livre até a possibilidade de conseguir o sustento na vida cotidiana, de um ponto de vista individual, tendo-se optado por trabalhar com software livre.

Um exemplo extremo desse tipo de palestra, e que ajuda a fundamentar melhor a categoria nerd, é a mesa que foi denominada "Profissionalismo para nerds – Eu já sei o que vou ser quando crescer". Considero-a um exemplo extremo por espelhar de forma acentuada uma tendência que vem se acentuando há alguns anos: a discussão parece estar se ampliando do debate sobre o software livre como uma atividade econômica alternativa (como conseguir dinheiro com algo que pode ser trocado de

graça ou como este funciona na economia tradicional) para abranger também um modelo de apresentação voltado à adequação do profissional de software livre à rotina diária de uma grande empresa, com todas as suas demandas de comportamento esperado. Na mesma nona edição do Fisl, um dos temas em debate foi a relação entre software livres e cooperativas. No anterior, o economista Paul Singer apresentou proposta de palestra em que discutiu a conexão entre software livre e economia solidária, tema abordado por outro palestrante na sétima edição. O público desses dois tipos de fala (a corporativa e a adequação profissional a um novo modelo de negócios) não parece ser o mesmo, embora estejam bem encaixadas em um mesmo evento. São dois caminhos diferentes que os jovens profissionais da área têm buscado trilhar: o emprego – ainda que frágil, muitas vezes trabalhando como pessoa jurídica – em grandes empresas que lidam com software livre, ou a filiação ou montagem de cooperativas de prestadores de serviços.

Ao mesmo tempo, a palestra "Profissionalismo para nerds – Eu já sei o que vou ser quando crescer" é interessante por abordar e estimular a identificação geral com a ideia do nerd, um dos estereótipos ligados ao aficionado em computação. Optei pela palavra nerd para descrever um dos grupos de público do Fisl, mas este não deve ser subsumido à aceção do termo nerd de uma maneira geral. O nerd do Fisl identifica-se e assemelha-se à categoria maior mas não é equivalente. O termo nerd é de origem estadunidense e foi popularizado mundialmente⁷⁵ na década de 1970, por meio de séries de televisão e filmes. O nerd serve para apontar, em geral, aquele que tem boa performance nos estudos – em especial nas disciplinas que mais dependem de pensamento lógico – mas que tem grande dificuldade em estabelecer relações sociais. Costuma também ter boa memória e ser fã obsessivo dos

⁷⁵ Kinney (1993) entrevistou mais de 80 adolescentes do ensino fundamental e médio estadunidenses e afirma que o oferecimento de atividades extra-curriculares a partir do ensino médio permite o envolvimento dos jovens considerados nerd em novas atividades. Nelas, os nerds passariam a ser mais aceitos por outros estudantes mais velhos e ganhariam em auto-confiança. Deste modo, construiriam uma nova categoria, dando a ela um caráter positivo, mesmo que alguns optem por continuarem se distinguindo dos “normais”.

gêneros ficção científica e fantasia. De caráter negativo, o estereótipo foi sendo progressivamente mais aceito conforme alguns ditos nerds alcançaram sucesso no mundo dos negócios (Bill Gates é frequentemente referido como o maior desses exemplos). No Brasil, o termo parece ter caráter menos negativo que no exterior (em especial nos EUA, onde o termo nasceu), possivelmente por ter se popularizado mais tarde no país, quando a ideia do sucesso dos nerds já era mais presente. Um termo similar e menos negativo, usado tanto no exterior e quanto no Brasil, é *geek*, que serve para marcar, em especial, os aficionados em tecnologia, mas sem carregar o peso tão negativo com relação à falta de habilidade social. Takhteyev (2007) realizou pesquisa de campo entre desenvolvedores de software livre do Rio de Janeiro em que identificou o uso – ao lado do termo nerd com identificação positiva, como trato aqui – do termo geek. Um falante pronunciou o termo como se fosse uma palavra brasileira – Takhteyev a grafou como “jeek” –, o que indicaria que o contato desses indivíduos com o termo é eminentemente por meio de textos e não com falantes nativos estrangeiros. Segundo Takhteyev, esses desenvolvedores expressariam pertencimento ao “mundo do software”, por meio da ligação com uma “comunidade open source” e pela identificação com uma “cultura nerd” baseada nos Estados Unidos. Os desenvolvedores estudados se sentiriam objeto de “preconceito” por parte das companhias estadunidenses e, por isso, entre outros, reagiriam reforçando esses laços culturais.



Cheguei somente no final da palestra “Profissionalismo para nerds”, mas consegui identificar que a audiência presente era mais jovem do que a do restante do evento, com alta presença de estudantes. Uma matéria, publicada em site dedicado ao mercado de Tecnologia da Informação, dá conta do assunto abordado. Cabe entender a matéria não como descrição acurada do ocorrido, mas observando-se os pontos elogiados por um site dedicado ao mercado de Tecnologia da Informação com forte ligação com as empresas.

“Só para nerds

Essa eu confesso que me chamou pelo título: “Profissionalismo para nerds – Eu já sei o que vou ser quando crescer”. A palestra parecia engraçadinha, um pouco de humor entre tantos zeros e uns, e era mesmo. Em cerca de 40 minutos de papo com uma sala lotada de cabeludos, Sulamita Garcia, da Intel, falou sobre a carreira na área técnica de software.

O assunto é ótimo, muitas dicas, muitas verdades, muitas instruções sobre comportamentos que devem ser mantidos, alterados ou expurgados do convívio

social e profissional. Mas tudo isso vou só pincelar por aqui, por enquanto, já que o tema rendeu tanto que me empolguei e fiz uma Entrevista da Semana com ela. Em breve, no Baguete, a totalidade do conteúdo.

Por hora, basta saber que, em primeiro lugar, nerd que é nerd sabe que, além de ser genial, tem de ser corporativo, ou seja: entender como funciona o meio empresarial, e se sujeitar ou não a ele. Se não suportar andar de terno e gravata, ajeitar o cabelo todo dia, atender a prazos e horários, conviver com a rotina e – sem choro! – produzir os famosos relatórios, nem adianta tentar, é melhor mudar de mercado.

E então, a programação é para você ou não?”⁷⁶

Há uma descrição sobre a estranheza da audiência, identificada como “cabeludos”. Segundo minha observação, ninguém ali diferia do restante do público do evento nesse aspecto, ao contrário, a única característica que chamava a atenção era a baixa faixa etária. A palestra é descrita como “engraçadinha”, principalmente se comparada ao conteúdo técnico do resto do evento. Outro ponto positivo, segundo o texto, seria a clara recomendação de comportamentos ao nerd, qualificado como genial mas que, ao mesmo tempo, é tomado como alguém de aparência desleixada, com dificuldade no cumprimento de prazos e em dar satisfações a seus superiores hierárquicos. Na reprimenda “sem choro!”, há uma infantilização daqueles com dificuldades em se adequar, como se os que estão fora das empresas/mercado – em outros momentos descrito como “mundo real” – não fossem amadurecidos e estivessem nessa posição por problemas de comportamento, disciplina e higiene. O modo negativo como aqueles que não estão no mercado são descritos guarda forte semelhança com descrições usadas no debate entre software livre e software proprietário por aqueles que defendem este último. Do mesmo modo, há bastante semelhança com o retrato dos membros do *free* feito pelo grupo *open*.

Na entrevista concedida pela palestrante à autora do texto acima, há um reforço maior das imagens criadas (veja texto completo nos anexos):

⁷⁶ Disponível em <http://www.baguete.com.br/noticiasDetalhes.php?id=24186>. Acessado em 19/01/2010.

Sulamita Garcia - Cabelinho enebado não dá!

Quando a palestra se chama "Profissionalismo para nerds" você já sabe que vai dar pano pra manga, certo? E dá mesmo.

Assisti à tal palestra durante o Fisl 9.0, na PUC-RS, e no final conversei com a palestrante, Sulamita Garcia, a gerente de Estratégia Linux e Open Source da Intel para a América Latina. Muitas dicas, muitos conselhos e muita risada, este foi o resultado do papo.

E agora, na Entrevista da Semana, você confere este compilado de informações que ensinam literalmente a ser um bom profissional no ramo da programação. Traduzindo, um bom nerd – sem ofensa! (Gláucia Civa)

O profissional de software geralmente entra muito cedo na área. Como embasar este início para construir uma carreira promissora?

Sulamita Garcia: Primeiramente é preciso acabar com o deslumbramento e encarar as coisas como são. Ser um programador em uma empresa, por exemplo, não significa só entrar e programar, mas também se adequar às normas da companhia. Se você fica deprimido só em pensar na ideia de ter de usar terno e gravata todos os dias, é bom refletir, pois muitas organizações da TI exigem isso.

(...)

Também é preciso ver as opções da carreira, o que é um aspecto animador: antigamente, um profissional da programação ou virava programador pela vida inteira, ou se tornava, com muito custo, chefe de programação. Hoje, um profissional desta área tem mais possibilidades, como atuar na gerência de projetos, como um mentor técnico em algum cargo ou até mesmo evoluir para o posto de CIO [Chief Information Officer].

(...)

Então os departamentos pessoais não possuem ainda o entendimento e entrosamento necessário para ajudar no crescimento do pessoal da programação?

Sulamita Garcia: (...) O tempo livre dos profissionais da programação nas empresas, quando ocorre, também já é visto de forma diferente, quando bem utilizado. Por exemplo: se você utilizar este tempo para enviar colaborações pela web, por meio de comunidades de software livre, por exemplo, estará divulgando seu trabalho e o da empresa. Além disso, estará contribuindo diretamente para a expansão do próprio setor de software, o que poderá reverter em benéfico para a própria companhia onde está trabalhando, já que as colaborações geram sistemas melhores, aplicações facilitadas, etc.

Outra coisa: colaborar pelas comunidades é dar visão ao seu trabalho não só para o exterior, mas também para o interior da empresa. Se hoje você está em um cargo baixo, mostrar seu trabalho pode ser uma forma de chamar a atenção para seus esforços, rendendo, quem sabe, uma promoção.

(...)

E agora... Como é essa história de ter de mandar o pessoal do software pro banho?

Sulamita Garcia: Ah... A higiene é um problema sério em muitas empresas. Você pode não acreditar, mas o profissional de software, muitas vezes – não em todas, sejamos bem claros! (risos) – entra tanto em seu trabalho que se esquece de cuidar de si. Assim, usa a mesma camiseta do Google até que esteja completamente esburacada, desbotada, velha. E tem pior: tem conferências empresariais onde, por incrível que pareça, a gente tem de dizer “olha pessoal, tem que tomar banho todos os dias, escovar os dentes, se vestir direito. Cabelinho ensebado, camiseta e bermuda não dá!”.

Quero chamar a atenção para alguns pontos da entrevista e da matéria. Em primeiro lugar, para a caracterização do estereótipo do nerd, que mistura comportamento exótico com competência técnica e inteligência. A prescrição de comportamento, como vimos, não funciona apenas como sugestão para maior ascensão profissional, mas também para se marcar diferenças entre grupos dentro do próprio movimento software livre. A necessidade de adequação do vestuário e do comportamento de alguns membros do movimento é queixa frequente entre aqueles que querem uma maior aproximação das empresas. Ao mesmo tempo, fica clara a precarização do trabalho e a necessidade de os indivíduos conseguirem prestígio a ser usado no ambiente profissional a partir de atividades exercidas fora do horário de trabalho (como colaborar com projetos livres). Tenta-se dar valor positivo ao trabalho eventual (*freelancers*), assumindo-o não como condição daqueles que não entram no mercado formal, mas como opção profissional. Aqui, claramente, a colaboração em comunidades de desenvolvimento de software livre é retratada como possivelmente benéfica tanto para o profissional que a ela se dedica – pois se trataria de oportunidade para chamar a atenção e conseguir uma promoção –, como para a empresa que o contrata, que ganharia em termos de imagem. O profissional representa a empresa não somente no tempo em que é regularmente contratado, mas também em suas horas de folga.

Ao que parece, estamos diante de um jogo, em que os comportamentos excêntricos, como o de

Stallman, funcionam tanto como um elemento de prestígio – entendidos como sinal de rebeldia, inteligência e independência – quanto como característica negativa ao se adentrar o mundo corporativo, se expostos e usados de forma exagerada. A combinação correta implica em ser um nerd por dentro, mas comportar-se dentro da empresa como um executivo.

O verão do código

A menção feita ao Google na entrevista concedida por Sulamita Garcia não é gratuita. Do mesmo modo que, como observamos, a Microsoft foi significada como a maior inimiga do software livre, o Google acabou identificado como símbolo de um novo modelo de negócios, visto por alguns como tributário à ideia de que as novas empresas devem obter seus lucros da venda de serviços e não de código. Embora hoje seja uma empresa com poder de mercado equivalente ao da Microsoft, ela não ganhou uma imagem negativa equivalente. Ao contrário, suas vitórias no mercado são entendidas por muitos como vitórias do próprio software livre. Em parte, associa-se à ideia de que a vitória de um modelo de negócios baseado em serviços é uma vitória do software livre, já que este assim mostra-se viável economicamente. Enquanto o ato de usar uma camiseta da Microsoft durante o Fisl seria interpretado como uma contradição com o evento e mesmo como uma afronta – talvez equivalente a ir no setor da torcida mandante de um estádio de futebol com a camisa do time visitante –, usar uma camiseta do Google é algo normal ou até mesmo símbolo de status, em especial para o grupo *open*. Meus dados corroboram a percepção de Takhteyev (2007), que identifica dois locais nos Estados Unidos de grande poder simbólico no imaginário dos desenvolvedores: o Vale do Silício, onde fica o Google campus e mais um conjunto de outras empresas identificadas com o *open source*; e Redmond (estado de Washington), onde se localiza o escritório central da Microsoft. Enquanto Redmond

representaria um pólo negativo, “do mal”, o Vale do Silício seria o pólo positivo, “do bem”.

Uma das palestras do Fisl foi ocupada pelo programa Summer of Code, do Google. Trata-se de uma espécie de concurso voltado à produção de softwares com código livre. Colaboradores para projetos são selecionados e pagos pelo Google, que exige que todo código produzido seja licenciado com uma das licenças aprovadas pela Open Source Initiative (OSI). O website do Summer of Code aponta os objetivos do programa e enfatiza seu caráter profissionalizante ao oferecer “real-world scenarios”:

“Google Summer of Code has several goals:
Get more open source code created and released for the benefit of all;
Inspire young developers to begin participating in open source development;
Help open source projects identify and bring in new developers and committers;
Provide students in Computer Science and related fields the opportunity to do work related to their academic pursuits (think "flip bits, not burgers");
Give students more exposure to real-world software development scenarios (e.g., distributed development, software licensing questions, mailing-list etiquette).”⁷⁷

A sala está repleta de gente, e o que acontece não é bem uma palestra, assemelha-se a uma aula em que o professor ausentou-se por alguns instantes e alguns alunos tomaram conta. O GSoC, como é chamado pela própria empresa, funciona da seguinte forma: uma entidade (uma empresa, uma distribuição livre, um projeto de software) postula o posto de mentora de algum projeto; a organização do GSoC escolhe e autoriza esses mentores; estudantes apresentam propostas de trabalho junto a essas entidades; o Google financia os estudantes.

Na mesa estão quatro pessoas, sendo que uma delas é Fernanda Weiden, que participa do PSL-RS e da organização do Fisl há muito tempo e, no momento, é funcionária do Google. Ela diz, em tom elevado para uma plateia agitada: “Somos nerds, então para a gente isso [produzir código] é diversão”.

⁷⁷ Disponível em <http://code.google.com/opensource/gsoc/2008/faqs.html> Acessado em 21/06/2008.

Ela fala de seu trabalho para o Google e tenta convencer a plateia a “codar”. Segundo ela, “os brasileiros” produzem pouco código. “Está na hora de o Brasil deixar de ser apenas um usuário e contribuir com código”. Esse é um debate que nasce a partir de 2003, quando o movimento brasileiro ganha notoriedade. Enquanto o Brasil é anunciado para o mundo como pólo de software livre, parte da comunidade questiona a baixa quantidade de projetos brasileiros. As reclamações surgiram quando do anúncio, por parte do governo Lula, da “migração” (o termo é nativo) para software livre de parte dos sistemas da máquina pública. De alguma forma, tanto a fala de Weiden como as queixas surgidas na época são um recado aos “políticos” que “falam muito e fazem pouco”.

Weiden continua falando do GSoC: “Não é só bom para o Google, é bom para a comunidade, para quem participa, para a Internet”. Outro membro da mesa complementa, aparentemente um participante aprovado da edição passada do GSoC: “A participação tem sido crescente de brasileiros, aqui [refere-se aos presentes na sala e ao Brasil] com certeza tem cérebros”. O esforço individual, a ideia de que todos são capazes e concorrem em iguais condições, é enfatizado. “Mande que você passa, se você se esforçar você passa”. “Eles não olham etiqueta de universidade”, diz, para apontar que o importante é o mérito efetivo, sinalizando um sistema meritocrático, algo valorizado no ambiente do software livre. “Ele [ao falar de um malaico que participou do programa] se esforçou bastante e passou”, exemplifica, sinalizando que as oportunidades mundiais são concretas.

É exibido um vídeo de incentivo à participação produzido por membros do Umit Project, um dos projetos de software que oferece mentores ao programa. No fundo, uma bateria *drum n' bass*, um *riff* de guitarra e sons de sirene formam a trilha *tecno* para um conjunto de slides com as seguintes frases:

"Google Summer of Code: o que é e como participar

Quer desenvolver Software Livre⁷⁸ nas férias?

Quer ganhar *dinheiro* por isso?

O *Google* paga US\$ **4500,00** para você desenvolver *Software Livre* nas férias!

Isso *não é um concurso*! Você participa e ganha **US\$ 4500,00**!

Como???

No *Summer of Code*, o *Google* seleciona as maiores *Organizações de Software Livre* do mundo...

[segue-se a exibição dos símbolos do Firefox (navegador livre), Fedora (distribuição de sistema operacional livre), Python Software Foundation (ONG que promove a linguagem de programação Python), Debian (distribuição de sistema operacional livre), PHP (ONG que promove a linguagem de programação PHP), FreeBSD (distribuição de sistema operacional livre), PostgreSQL (ONG que promove a linguagem de programação PostgreSQL), Eclipse (ONG que promove o ambiente de programação Eclipse), Moodle (empresa), Umit (ONG que promove o software Umit)]

–(No total são mais de **170 Organizações**!)

–Estas Organizações selecionam algumas *ideias legais* que *você* pode trabalhar

–E divulgam estas *ideias* aqui: <http://code.google.com/soc>

[É exibida a página na internet do GSoC]

–*Você* entra lá...

–Escolhe a *ideia* que achar *mais legal*...

–E propõe para uma *Organização a sua maneira* de concretizar essa *ideia*

–A *Organização* vai avaliar a *sua proposta*...

–Se você for *selecionado*, a *Organização* irá *orientá-lo* durante a execução da sua proposta

–Você vai desenvolver um *Software Livre* com a ajuda de uma *Organização de peso*

–Esta *Organização* vai *distribuir o seu projeto* para o mundo todo

–O *Google* vai te pagar **US\$ 4500,00**

–vou repetir...

–O *Google* vai te pagar **US\$ 4500,00**

–Você vai ganhar uma *camiseta* ;-)

–E um *certificado de participação* emitido pelo *Google*!

–E vai poder colocar isso no seu *Currículo*!

–Sabe onde alguns *ex-participantes* estão *trabalhando* hoje?

[segue-se a exibição de logotipos de algumas empresas e projetos: Apple, Google, Nokia, Firefox, Drupal]

–*Está animado(a)???* Quer saber como fazer uma *boa proposta*???

–*Dica No 1*: Escolha apenas *uma*, no *máximo duas* propostas

–*Dica No 2*: Estude a *melhor* maneira de resolver o problema proposto. Procure a ajuda da comunidade.

⁷⁸ Aqui, o termo Open Source Software foi traduzido para Software Livre. Embora não seja uma tradução incorreta, ela aproxima duas ideias não equivalentes: software livre (free software) e código aberto (open source).

- Dica No 3:** Elabore uma proposta *criativa*, com *detalhes* sobre a sua forma de *resolver o problema*
 - Dica No 4:** Elabore um *cronograma*
 - Dica No 5:** Demonstre que você está se *dedicando* ao projeto e possui *real desejo* em participar
 - Dica No 6:** Estude e demonstre conhecimento nos seguintes assuntos...
 - Usabilidade; Portabilidade; Processo de instalação; Documentação; Internacionalização; Dependências
 - Dica No 7:** Não perca o prazo de inscrições: de **24 a 31 de Março!** ;-)
 - Ah! E fique **atento** ao seu **e-mail!**
 - A inscrição é feita no próprio site do *Summer of code*:
<http://code.google.com/soc>
 - Está **inseguro**?
 - Não precisa!**
 - Seguindo as **dicas**, suas **chances** de ser selecionado e concluir o projeto são **grandes**.
 - Quer outra **dica**?
- [a partir deste momento as frases sobrepõe-se ao logotipo do Projeto Umit]
- Participe com o **Umit Project!**
 - O **Umit** é uma interface para **varredura de rede**
 - Com o **Umit**, é possível descobrir **portas** e **serviços** nas máquinas de sua rede...
 - comparar resultados** de varreduras...
 - fazer varredura na sua rede **à distância**, usando uma **interface web**...
 - e **muito mais**... Logo, estaremos listando aqui a **sua grande ideia** ;-)
 - Envie-nos sua proposta!"**

Os termos em negrito e itálico acima estão grafados dessa forma no original. O vídeo se parece com uma apresentação motivacional utilizada em empresas. A maneira encontrada para estimular a participação no programa é bastante diferente do discurso típico de mobilização de apoiadores para projetos de software livre. Em nada se parece com o Manifesto GNU, aquele que Stallman publicou em 1984 e que foi o primeiro convite aos programadores para que participassem do movimento software livre. Stallman falava que o sistema livre beneficiaria a todos os usuários de computadores, que inclusive poderiam melhorar o programa eles mesmos, sem ficarem dependentes de empresas ou programadores. Apontava que criar software livre era a única maneira de se preservar “o ato

fundamental da amizade entre programadores”: o compartilhamento de software. Não se trata aqui de tomar como verdadeiro ou questionar o que diz, mas perceber quais são seus argumentos para a mobilização. Em lugar da preservação de laços entre entusiastas da mesma atividade, ganhou espaço a motivação individual, profissional. O candidato deve participar porque: haverá um bom pagamento; organizações importantes orientarão os participantes; o Google certificará a participação e isso é importante para o currículo; quem já foi selecionado por esse projeto hoje trabalha para grandes corporações transnacionais; o projeto será distribuído para o mundo. Takhteyev (2007) fala da importância para os desenvolvedores brasileiros de se sentirem parte de uma comunidade mundial. O que o vídeo faz perceber – distribuído mundialmente e originalmente produzido em inglês, sendo o português apenas uma das versões para outro idioma – é que esse esforço e essa vontade de participar dessa comunidade, simbólica e profissionalmente, não se restringe aos desenvolvedores do Brasil. É mexendo com ela e pelo desejo de ascensão profissional dos jovens que o GSoC recruta seus candidatos.

Após a exibição do vídeo, um terceiro elemento comenta à mesa: “Se esforce, estude à noite, dê um jeito. Não é só pelo dinheiro, é porque você aprende”. “Eu fiz isso pela camiseta”, responde outro membro da mesa. Seguem-se mais piadas sobre a camiseta, que só é dada a quem completa o programa. Ela acaba funcionando como um currículo móvel, um objeto de status em determinado grupo ou para se circular em ambientes como o Fisl. Mais tarde, alguém do público pergunta sobre possíveis inscrições em grupo e faz piada sobre a divisão do “grande prêmio”: a camiseta.

Na plateia, alguém levanta dúvida sobre a submissão ser obrigatoriamente em inglês. “Perdi dois dias com isso”, diz aquele que questiona. Weiden: “Tem que ser”. O jovem que já participou do GSoc pelo projeto Umit e que está na mesa complementa: “É em inglês. Também tive dificuldade com isso. Mas foi bom, aprendi inglês. O GSoc é uma ótima oportunidade de aprender inglês”. Segundo

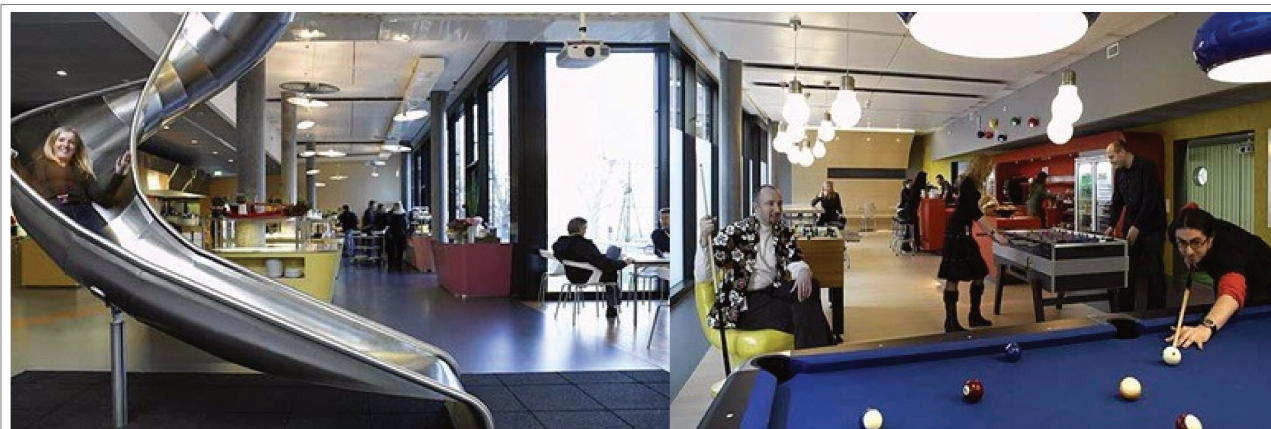
Takhteyev (2007), o domínio perfeito do inglês é uma das habilidades que os desenvolvedores consideram essencial, pois a informação de melhor qualidade e original estaria nesse idioma. Além disso, seria a língua em que a comunidade mundial se comunica.

Essa sessão dedicada ao Summer of Code parece de alguma forma complementar à que descrevi anteriormente, chamada “Profissionalismo para Nerds”. O crescimento do software livre no meio empresarial levou a uma transformação da ideia de “Negócios Livres”, como era o nome da seção de uma das primeiras versões do site do Fisl. Ela persiste, concretizada provavelmente nas cooperativas de software livre, que em sua maioria se pretendem como iniciativas empresariais diferenciadas, que não objetivam a construção de um grande patrimônio mas sim a obtenção de uma remuneração justa pelo trabalho⁷⁹. Porém, ao lado delas, crescem as iniciativas de código aberto nas grandes empresas, que recrutam funcionários em eventos como o Fisl e oferecem a eles carreiras bastante semelhantes à de seus profissionais tradicionais de informática. Assim, estudantes de computação passam a enxergar o software livre como um diferencial de qualificação para o mercado e não como a construção de “um outro mundo possível” ou de uma rede de economia alternativa com traços não-capitalistas.

A empresa que tem conseguido mais habilmente construir-se como pólo de atração para esses novos profissionais é o Google. Ao mesmo tempo em que se mostra como um concorrente forte no mercado de informática, que para muitos já desbancou a Microsoft, tem conseguido manter uma imagem alternativa, de estimuladora da criatividade e não inimiga dos ideais gerais do software livre. Reportagens que se tornaram frequentes sobre o cotidiano de trabalho no Google fascina os novos profissionais ao passarem uma imagem de trabalho-divertido, algo que se liga, como veremos, à ideia de trabalho para o hacker. O Google procura estimular essa imagem positiva franqueando suas instalações à imprensa e estimulando seus funcionários a relatarem uma experiência positiva de

⁷⁹ Uma dessas cooperativas, a já citada Colivre, já se tornou tema frequente em rodas de conversas sobre o assunto, por praticar uma divisão salarial igualitária: da faxineira ao desenvolvedor mais especializado todos ganham o mesmo.

trabalho. As imagens a seguir foram capturadas pelo editor de tecnologia do jornal espanhol *El Mundo*⁸⁰ em visita ao escritório de Zurique do Google e foram reproduzidas em sites brasileiros⁸¹. Elas exemplificam essa imagem que a empresa procura transmitir, de corporação que estimula a criatividade de seus funcionários⁸² e não os submete a uma disciplina rígida e repetitiva de trabalho, embora este seja intenso.



O texto que acompanha estas e outras fotos dá detalhes desse ambiente que mais se parece com um parque de diversões. A justificativa é que os funcionários precisam ter boas ideias, por isso haveria blocos de papel em todo o canto, para que nenhuma delas seja perdida. Enfatiza-se a liberdade, porém com responsabilidade: “Cada um administra seu tempo e seu trabalho **como quer**. Não há horário e nas horas de descanso é permitido jogar uma partida de *Guitar Hero* [videogame], sinuca ou um jogo de mesa. **Os prazos de entregas e desenvolvimento/ produção, isso sim, precisa ser cumprido.**”

O ethos dos executivos e o capital humano

Em tese de doutoramento intitulada *O "Ethos" dos Executivos das Transnacionais e o Espírito do Capitalismo*, Osvaldo López-Ruiz (2004), a partir de investigação sobre os executivos das transnacionais, examina o que seria um novo conjunto de valores, princípios e normas a permearem a

⁸⁰ Disponível em <http://navegante2.elmundo.es/navegante/gadgetoblog.html>

⁸¹ Disponível em <http://papodehomem.com.br/literalmente-dentro-do-google-com-fotos/>

⁸² Empresas como a Apple e até mesmo a Microsoft já tentaram vincular suas imagens a esse tipo de ambiente de trabalho. Nesse sentido, a Apple conseguiu melhores resultados. No ambiente do software livre, entretanto, o Google é quem carrega mais essa imagem positiva.

sociedade contemporânea. Segundo ele, os executivos e as corporações transnacionais são figuras emblemáticas de uma forma de se estar no mundo que começa a prevalecer, espalhando-se por diversas classes e profissões. No centro desse processo está a ideia de capital humano, que nasce nos anos 1960 para a explicação de questões econômicas, mas que se populariza trinta anos mais tarde, sendo aplicada na prescrição de condutas pessoais e profissionais que poderiam tornar não somente os indivíduos mais prósperos como mais felizes.

Além disso, López-Ruiz aponta para um novo tipo de ligação profissional entre empresa e executivos que nos interessa em particular. Segundo ele, o novo executivo não vê a si mesmo – e não é tratado discursivamente pelas empresas – como um empregado, mas como alguém que se associa ao empregador, de maneira temporária, pensando sua relação como a de uma empresa que se associa a outra. Essa ligação vai além da simples “prestação de serviços”, abarcando também a imagem de uma parceria simbólica, em que a associação leva a uma valorização das partes envolvidas em termos de mercado. Ter essa relação com uma empresa prestigiada, vista pelo mercado como inovadora, por exemplo, levaria a uma valorização futura desse parceiro associado, que pode passar a cobrar mais ao ligar-se a outra empresa num período posterior. O executivo não seria um trabalhador, mas um “empresário de si”. No patrimônio dessa “empresa de si” constariam suas ligações profissionais passadas, assim como todas as atividades cotidianas desse sujeito que possam ser revertidas, de forma direta ou indireta, em benefício a quem a ele se associar. Assim, atividades cotidianas realizadas em períodos distintos dos “de trabalho”, como cursos não-técnicos, relações pessoais e outros, seriam entendidos como investimento desse sujeito na sua “empresa de si”.

Quero trazer alguns dos elementos identificados por López-Ruiz e que percebo estarem presentes em especial nessas palestras dedicadas ao público nerd do Fisl, abordadas acima. Tanto a palestra “Profissionalismo para nerds” como na exposição do programa GsoC do Google vemos a

repetição de alguns elementos como: a ideia do trabalhador-investidor; a educação que não se encerra em um determinado período da vida; a construção de uma marca, como um produto, em lugar do nome do profissional, que garantiria a empregabilidade; a consonância entre valores pessoais e da empresa; entre outros. Poder-se-ia argumentar, com López-Ruiz, que se trata justamente da prevalência dessa determinada forma de estar no mundo em esferas que vão além do mundo dos executivos. Meu argumento, porém, é que se trata de um processo de conjunção mais intensa, um ponto de aproximação e de influência cruzada entre os “liberais-comunistas”, citados no capítulo anterior, e a ideologia do capital humano e do trabalhador não mais como funcionário, mas como capitalista de si mesmo que atua em relação de dependência com a empresa. Na área da Tecnologia da Informação, em particular, a formação permanente – pois os processos de transformação são acelerados –, o trabalho que se estende para além das horas e locais regulares – em que as facilidades da tecnologia têm peso grande –, e o trabalho coletivo – se não para a produção ao menos para os testes de produto – são a norma já há algum tempo. A ideia de transparência absoluta, tomada como um valor sempre positivo, que também está contida nos dez mandamentos dos “liberais-comunistas” citados por Žižek (2006), está no centro do software livre, em especial para o grupo *open*, que vê nela o caminho necessário para o software “evoluir”. É a abertura do código-fonte a todos que permitirá o melhoramento e a depuração dos erros. É a transparência irrestrita e imediata que, por si só, garantirá o filtro popular. Há uma analogia aí a ser feita com as empresas que se credenciam como “éticas” ao abrirem suas contas a auditores, ao se tornarem transparentes.

A conjunção desses elementos é um processo importante, em que as características do software livre como um movimento social em que militantes se envolvem na defesa de uma causa perde força para uma associação de cunho individual de sujeitos que se ligam a projetos livres como um “investimento em si mesmo” e cujo resultado de seu trabalho será sua própria valorização no mercado.

Trata-se de um processo de aproximação mútua, não apenas o software livre enquanto movimento se transforma com o reforço dessas características, mas o modelo do “capitalista de si” encontra um terreno fértil dentro da ideologia do software livre.

Vale então retomarmos alguns pontos. Na palestra “Profissionalismo para nerds”, pudemos acompanhar como o trabalho nas horas de folga nas redes de colaboração do software livre significa um incremento de valor não somente para o sujeito como para a empresa que ele representa. Da mesma forma, o GSoC é descrito como algo que é bom “para o Google, para quem participa, para a comunidade e para a Internet”. Ele é o veículo por essência para construção de uma marca em torno do nome de um profissional, que se valoriza ao se conjugar a projetos de prestígio, o que inclui as organizações com quem o inscrito vai trabalhar diretamente, mas principalmente o Google, a quem interessa ao profissional ter seu nome associado. Não cabe aqui questionar o valor de US\$ 4500,00 destinado ao profissional, que pode ser alto do ponto de vista do estudante, mas é relativamente baixo se comparado à remuneração dos profissionais da área. A questão é que mesmo que esse valor fosse ainda mais baixo, o profissional toma seu envolvimento como um investimento na própria carreira e não uma remuneração ao trabalho desenvolvido. Nesse sentido é que a graça feita com a camiseta oferecida ganha significado. A peça de roupa é símbolo da existência de um contrato de associação profissional entre duas empresas.

Outro aspecto a se salientar é o receituário oferecido para se tornar um trabalhador global do mercado corporativo. Pudemos acompanhar isso de forma mais direta na palestra proferida por Sulamita Garcia e é algo também presente na sessão dedicada ao GSoC. Trata-se de se explicitar um modelo de trabalhador de alta renda do setor de tecnologia, em que aparecem combinadas as características imaginadas de um executivo de empresa transnacional com características específicas de um trabalhador de tecnologia.

López-Ruiz descreve o final do dia de um desses executivos, criando uma mini-estória fictícia em que o personagem, após 16 horas de trabalho continuado, chega na casa em que reside sozinho e, exausto, por trinta minutos consegue apenas deleitar-se, inerte, apreciando os caros objetos de design que possui. Recobra-se somente após sentir uma pontada no estômago de fome, já que não come desde o café, quando então se levanta para pedir comida tailandesa pelo telefone, não sem antes se lamentar por não ter ido, naquele dia, à academia de ginástica. Deseja que a comida chegue logo, pois em cinco horas estará novamente no escritório.

Estão, na descrição, alguns dos elementos que compõem essa forma de estar no mundo que, mais do que terem relação com o cotidiano de um número significativo de pessoas globalmente, passaram a ser objeto de desejo profissional de um número crescente de trabalhadores em busca de ascensão. O trabalho por longas horas, o sofrimento físico que é sinal de intensa dedicação, a necessidade de recorrer a serviços especializados, como o de preparação de refeições, dada a falta de tempo, embora pareçam percalços incômodos, são antes penitências cuja recompensa é o consumo e, principalmente, a sensação de sucesso. No caso dos trabalhadores em tecnologia, todos esses elementos estão presentes, especialmente o pertencimento a um mercado global de trabalhadores, integrados em um processo produtivo para além do Estado, e a conexão incessante à rede que, por sua vez, funciona como um escritório virtual que nunca fecha. Além disso, há elementos que são próprios do que é tomado como uma cultura própria do software livre, como a mistura entre trabalho e diversão – se a diversão também é trabalho o trabalho é divertido, e ambos nunca param – e, principalmente, o elogio a uma certa independência relativa, em que o patrão é quase um mecenas, que financia uma atividade essencialmente criativa que precisa ser exercida com certa liberdade, não-conformidade a padrões e de modo lúdico. Isso justifica ambientes como o do Google, descrito como um parque de diversões, porém disciplinado pela necessidade de resultados.

Conclusão

Nos últimos anos, mudanças técnicas e de mercado levaram a um crescimento expressivo do uso de código livre por parte das grandes empresas. A força de mercado de corporações como a Microsoft diminuiu frente ao crescimento de outras como o Google, que baseia seu negócio em prestar serviços e vender anúncios, utilizando softwares livres em várias de suas operações. Com isso, mudou também o movimento software livre. Inicialmente, os desenvolvedores eram chamados a colaborar em projetos ou a partir de argumentos altruístas – visando a construção de uma alternativa ao sistema proprietário – ou por exercício de um entretenimento intelectual. Atualmente, ganha força a mobilização de voluntários interessados em melhorar seu currículo visando seu crescimento profissional. Isso altera o perfil dos militantes do movimento, que passam a fazer parte dele visando mais a construção de uma imagem pública profissional. Este trabalho não pretende afirmar a superioridade numérica de nenhum dos modelos, mas indicar a presença e postular os efeitos.

Soma-se a isso o fato de que arregimentação política e de trabalho quase nunca funcionarem isoladamente. No software livre, convencer um membro a aderir politicamente a um determinado grupo significa também contar que este indivíduo contribuirá no desenvolvimento de determinados softwares livres, mesmo que seja apenas ao utilizá-los.

Adiciona-se, então, um novo elemento na competição entre projetos de software por desenvolvedores voluntários. Surge como o interesse mais forte a projeção pessoal e a inserção profissional, a escolha passa a ser pautada por projetos que ofereçam projeção e/ou projetos que utilizem tecnologias e linguagem que estão sendo usadas pelas grandes empresas. Nessa escolha de

pertencimento a uma determinada corrente política, quase nunca feita de forma explícita e consciente, pesa o imaginário sobre o caminho profissional a se seguir, que pode ser o do mercado, em geral de mais prestígio e associado com a ideia, até certo ponto contraditória, de “mundo real do mercado”; ou o dos ganhos mais módicos – e de hipotético maior risco, dada a recusa em se seguir “o padrão do mercado”.

Neste capítulo, além de procurar demonstrar esses processos, objetivei também trazer alguns dos valores e normas que operam em um evento como o Fisl e no movimento software livre como um todo. Busquei descrever apenas algumas características principais, que julgo como mais importantes para se desenhar esse grupo e entender seu comportamento no evento. Nesse sentido, destaca-se, em especial entre os mais jovens, a ideia de que participar do movimento software livre é também ser nerd (ou geek), o que significa inteligência técnico-matemática e o consumo de certos produtos da cultura pop (principalmente filmes de ficção científica e história em quadrinhos), além de uma certa dificuldade de adequação social e inaptidão para atividades esportivas. O nerd é entendido como, de alguma forma, um marginal. Porém não no sentido do transgressor criminoso, mas como gênio incompreendido e excêntrico, por isso estando à margem. Vimos como Stallman é alguém que trabalha as excentricidades a seu favor ao usá-las para aumentar a mística em torno de sua capacidade técnica. Embora seja atacado por membros do grupo *open* que afirmam que esse tipo de comportamento afasta as empresas – acompanhamos como elas desejam nerds “limpos” e trajados como executivos –, Stallman é respeitado por esse grupo menos por sua história e mais por ser reconhecido como alguém tecnicamente muito capaz. Ao mesmo tempo, a trajetória de vida e profissional de Stallman acabam por funcionar como imagem típica do grupo *free*.

No capítulo seguinte, abordo as ideias em torno da palavra hacker, que de certa forma se mistura e se alimenta de parte dos elementos que formam a ideia de nerd. Hacker, contudo, é uma

qualificação mais nobre e menos vulgar, destinada aos membros de prestígio do movimento e utilizada com mais parcimônia.

Cap. 5. Hackers

Nos capítulos anteriores, aparecem diversas referências ao termo hacker, ora como adjetivo aplicado a certos membros ou coletivos, ora como designador de um “espírito”, “ética”, ou como elemento de diferenciação de grupos distintos, como na oposição entre “hackers” e “políticos” abordada no capítulo 3. Como dito anteriormente, o termo hacker encerra a qualificação de maior prestígio no movimento software livre. Muitas vezes, o software livre é definido como um “movimento de hackers”, ou então os programas utilizados são definidos como “escrito por hackers”. O “ser hacker” ou ter o “espírito hacker” ou agir e pensar de acordo com a “ética hacker” estão em relação de equivalência com o que viria a ser o próprio software livre, seja enquanto movimento, comunidade ou conjunto de softwares.

Contudo, essa relação tão próxima entre hackers e software livre não significa que, por exemplo, ao nível individual, a qualificação hacker possa ser aplicada a todos os participantes do Fisl. Um membro do movimento, se questionado a respeito disso, utilizaria o termo para designar apenas alguns poucos indivíduos presentes no evento, e estes seriam pessoas de prestígio – mesmo que, no caso, o informante tenha algum tipo de divergência com aquele que ele qualifica como hacker. Assim sendo, temos que a separação entre aqueles que são hackers e aqueles que não são hackers implica em uma posição de autoridade para com as coisas do software livre. As opiniões de alguém que é reconhecidamente tido como hacker sobre o software livre estão em melhor posição de serem aceitas como legítimas do que as de outro membro qualquer.

Neste capítulo, quero trazer elementos que permitam fazer uma discussão sobre quem são e o

que é ser hacker, mostrando como essa caracterização reflete o que seria o sujeito ideal (o que ele faz, como ele age, que tipo de lógica de pensamento adota) do software livre, como ser apontado como hacker implica em assumir uma posição de autoridade e como sutis alterações ou ênfases na ideia geral de hacker estão ligadas à definição sobre qual seria o “espírito” do software livre.

Para estabelecer essa discussão, faço um duplo movimento. Primeiro, trago três textos que, em momentos distintos, foram importantes na popularização ou estabelecimento do termo hacker: o livro *Hackers: Heroes of the Computer Revolution*, publicado em 1984 pelo jornalista Steven Levy; o documento “How to become a hacker”, escrito por Eric Raymond e publicado em sua primeira versão em 1996; e o livro *A ética dos hackers e o espírito da era da informação*, escrito pelo filósofo Pekka Himanen e publicado em 2001. Posteriormente, analiso alguns debates travados em listas de discussão e páginas na Internet e que referem-se ao que é ser hacker e quem pode ser assim chamado. Nesses debates, aborda-se a propriedade de o então ministro da Cultura, Gilberto Gil, ter se auto-intitulado “ministro hacker” e o que se constituiria “ser hacker”.

Minha abordagem aqui procura conjugar uma visão sobre o termo hacker que se constitui internacionalmente (publicada em livros e documentos na Internet), com a perspectiva brasileira para o termo, dada especificamente pelo meu contato com o movimento software livre. O propósito é entender e dimensionar o peso político da aplicação do termo hacker pelos integrantes do movimento brasileiro. O objetivo não é discutir exaustivamente os hackers ou a “cultura hacker”, mas construir os caminhos essenciais para o entendimento do que está sendo dito e do que está em jogo quando da menção ao termo ou quando da caracterização de grupos ou pessoas como hackers.

Hacker não é cracker, ser hacker é uma honra

O sentido mais popular para o termo hacker, ligado à ação frequentemente criminosa de invasão e destruição de sistemas, é rejeitado pelos adeptos do software livre. Essa visão negativa, divulgada pela mídia a partir dos anos 1980 e enraizada mais fortemente no imaginário geral, seria uma deturpação de um sentido original. Embora a pulsão pela descoberta de maneiras criativas de desbloqueios de sistemas protegidos (quebrar o código de proteção de um arquivo ou ter acesso completo para modificar um aparelho eletrônico, por exemplo) seja considerada uma característica dos hackers, o uso de modo a prejudicar alguém ou para a destruição de algo é atribuído a um outro conjunto de tipos que não hackers. Em geral, o nome utilizado para esses outros é cracker, que por sua vez podem ser diferenciados de maneira específica em outro sub-conjunto de tipos⁸³.

Em lugar de ser algo negativo, obter o reconhecimento como hacker é quase como receber um título honorífico. Sendo assim, embora a maioria ambicione ser hacker ou possuir o “espírito hacker” e julgue comungar com a “cultura hacker”, poucos adotarão o título para si. Fazê-lo pode significar colocar-se em posição de certo ridículo ou de questionamento público. Esse “ridículo” torna-se mais grave no caso de o postulante não ser alguém de habilidades técnicas reconhecidas. Embora fale-se constantemente em “espírito hacker” ou “cultura hacker”, características que iriam além do conhecimento técnico por estarem mais ligadas à atitude, a um determinado jeito de fazer as coisas e lidar com o mundo, praticamente na totalidade das vezes apenas sujeitos com alguma produção objetiva em termos de código ou hardware serão classificados como hackers. Mas mesmo sujeitos com alguma produção técnica (como manuais simplificados de uso de software, os chamados tutoriais) estão sujeitos a terem questionada a sua posição se, em fóruns na internet, blogs ou listas de discussão, forem

⁸³ Script kiddies, defacers, packet monkey, etc.

enfáticos demais ao afirmarem sua ligação com hackers de maior prestígio reconhecido.

O surgimento da ideia de hackers

Embora o termo tenha surgido em meados dos anos 1950, o termo hacker só vai se popularizar a partir do início dos anos 1980, mais especificamente em 1984, quando é publicado o livro do jornalista Steven Levy *Hackers: Heroes of the Computer Revolution*. Trata-se de um trabalho extenso, rico na construção de personagens, que consistiu na realização de entrevistas com o que o autor afirma serem três gerações diferentes de hackers: os hackers originais, chamados de “verdadeiros hackers” dos anos 50 e 60; uma segunda geração de hackers, os “hackers do hardware”; e os “hackers dos games”, surgidos entre os anos 70 e 80. Embora o autor repita o termo hacker com bastante insistência, percebe-se muito pouca menção ao termo no discurso direto das dezenas de entrevistados. O livro é um trabalho essencialmente jornalístico, o que significa o arranjo de muitas histórias pessoais, coloridas pela descrição física dos personagens. A obra teve um impacto tão grande que estimulou a realização da primeira conferência hacker, na Califórnia, um evento para convidados que reuniu alguns dos personagens entrevistados por Levy, além do próprio autor⁸⁴. O evento deu origem também a um vídeo documentário, *Hackers: Wizards of the Electronic Age*, produzido por uma emissora local e retransmitido nacionalmente pela rede pública de televisão dos EUA, PBS.

A época é também de grande efervescência, de popularização dos primeiros computadores pessoais. No cinema, no ano anterior registra-se o grande sucesso do filme *War Games*, que retrata um adolescente expert em computadores que inadvertidamente invade um computador do sistema de

⁸⁴ Ver <http://www.think.org/conference/faq.html> Acessado em 18/01/2010.

defesa nuclear dos EUA encarregado de identificar possíveis ataques militares externos. O jovem pensa tratar-se de um game, e assim inicia um processo de crise que quase culmina em guerra mundial. Poucos anos mais tarde, em 1988, notabiliza-se o caso de Kevin Mitnik, jovem que invadiu os sistemas da Digital Equipment Corporation, foi condenado e, na parte final de sua condicional, voltou a invadir sistemas, desta vez da empresa Pacific Bell. Mitnik recebeu uma punição exemplar, considerada por muitos exagerada, e o caso teve grande repercussão na mídia, associando-o ao termo hacker.

A inquietude com qualquer tipo de bloqueio ou trava é uma das atitudes, entre outras, que Levy destaca como ligada aos primeiros “verdadeiros hackers” e de alguma forma persistente nas gerações seguintes. Embora nenhum manifesto tenha sido escrito, Levy elenca alguns itens que formariam essa ética, itens que, a partir de sua expressão em livro de Levy, foram manipulados e transformados em novos guias gerais nos anos seguintes. Os itens postos por Levy são:

“Access to computers and anything which might teach you
something about the way the world works should be unlimited and total.
Always yield to the Hands-On Imperative!
All information should be free.
Mistrust Authority Promote Decentralization.
Hackers should be judged by their hacking, not bogus criteria
such as degrees, age, race, or position.
You can create art and beauty on a computer.
Computers can change your life for the better.
Like Aladdin's lamp, you could get it to do your bidding.”
(Levy, 1984)

Para cada um desses princípios, Levy expõe uma racionalização, derivada da história específica daquele grupo que vivia a emergência da computação em meados do século XX. A aversão ao bloqueio de acesso, por exemplo, se daria pela avidez por conhecimento dos hackers, que teriam a necessidade de manipularem e explorarem em profundidade objetos mecânicos, eletrônicos e, a partir daquele momento, digitais. Cadeados e portas fechadas em universidades seriam empecilhos para o

aproveitamento dos recursos materiais utilizados para a exploração intelectual. Pelo mesmo motivo, toda a informação deveria ser livre, para que não existam obstáculos ao conhecimento. Os computadores, máquinas que surgem naquele período, são importantes viabilizadores do progresso intelectual, capazes de dar aos indivíduos que os manipulam a sensação de poder criativo de soluções e de controle da própria máquina.

Esses “verdadeiros hackers” de que fala Levy formam um pequeno grupo que frequentava o campus da Massachusetts Institute of Technology (MIT), em sua maioria jovens estudantes e alguns professores, grande parte também pertencente ao Tech Model Railroad Club, um grupo de apaixonados por modelismo ferroviário e que gastava boa parte de seu tempo montando e projetando complexos modelos. As máquinas que centralizavam a atenção desses hackers foram, inicialmente, o *mainframe* IBM 704 e, mais tarde, máquinas mais simples mas mais poderosas como o PDP-1 e PDP-6. Os personagens descritos não se distanciam do estereótipo do nerd como entendido em seu contexto original – ou seja, o ambiente colegial estadunidense – e que mais tarde se espalhou pelo mundo pela ficção hollywoodiana: o jovem em geral do sexo masculino, avesso a esportes, com dificuldades de relacionamento social – por vezes assexuado –, inclinado a abrir e construir aparelhos eletrônicos e mecânicos, de bom desempenho escolar e fã de ficção científica. Levy descreve Peter Deutsch, por exemplo, então um jovem de pouco mais de 12 anos e que circulava pelo grupo, desta forma: “He was a shy kid, strong in math and unsure of most everything else. He was uncomfortably overweight, deficient in sports, but an intellectual star performer”.

De acordo com Levy, esses hackers formavam um grupo até certo ponto fechado, de hábitos não muito comuns mesmo para os colegas do MIT e por vezes com manias difíceis de se lidar para quem não estivesse disposto a reconhecer o que o grupo teria de especial. Os hábitos de trabalho eram pouco usuais para a época, com longas jornadas por vezes atravessando a madrugada em busca de uma

solução perfeita. Depois de desabar de cansaço, os hackers se retiravam igualmente por um período mais longo de tempo. Essa rotina estafante só seria possível pelo alto envolvimento intelectual de todos, que não viam a atividade como um fardo, mas como diversão. A comida era do tipo *fast food* ou chinesa, acompanhada por refrigerante. Embora entre os princípios da ética hacker elencados por Levy estejam a descentralização e a desconfiança de autoridades, o grupo não deixava de usar rótulos como *winners* e *losers*, provavelmente reproduzindo divisões existentes em suas escolas na juventude, quando então eles eram os excluídos. Segundo Levy, o rótulo era aplicado especialmente aos estudantes de graduação do MIT e os *winners* eram os hackers. Era um julgamento binário, aplicado a todos que circulavam em torno do laboratório de inteligência artificial do MIT: “The sole criterion was hacking ability. So intense was the quest to improve the world by understanding and building systems that almost all other human traits were disregarded. You could be fourteen years old and dyslexic, and be a winner. Or you could be bright, sensitive, and willing to learn, and still be considered a loser.”

Essa primeira geração, com o passar do tempo e em especial com a maior consolidação do mercado de computadores, teria se espalhado pelos EUA, alguns migrando para a Califórnia. É lá, mais especificamente na região que viria a ser conhecida como o Vale do Silício, que Levy identifica o nascimento de uma segunda geração hacker, que chama de “hackers do hardware”. Os “hackers do hardware” surgem também em maior sintonia com o ambiente de contestação política da época, entre as décadas de 1960 e 1970. Diferente dos “verdadeiros hackers”, estes já se mostram mais preocupados com a popularização da tecnologia e dos computadores, entendendo-os como ferramentas que podem ser usadas na própria luta política, ferramentas que poderiam libertar as pessoas e não aprisioná-las, como era a visão de boa parte dos movimentos de contestação da época. Esse viés mais politizado, contudo, não deve ser generalizado.

Um efeito mais palpável dessa geração foi a proliferação dos computadores pessoais. Muitos do

grupo estiveram envolvidos com a apropriação dos primeiros microchips e implementação em conjuntos montáveis de computadores, entregues sob encomenda. Destaca-se, entre esses indivíduos, a livre troca de informações sobre os seus produtos, dada em reuniões como as do Homebrew Computer Club, que tinha entre seus fundadores ativistas políticos. Essas reuniões foram frequentadas por muitos dos futuros milionários do mercado de informática que ascenderam no início dos anos 1970 e início da década seguinte, como Steve Jobs e Steve Wozniak, fundadores da Apple. O Apple I teve suas primeiras vendas feitas em uma reuniões do Homebrew Computer Club.

Já a terceira geração, que Levy chama de “hackers dos games”, seria tributária à popularização dos computadores proporcionada pela ação da geração anterior. Um dos principais atrativos para a compra de computadores pessoais, como o Apple II, por parte das famílias do início dos anos 1980 eram os games, jogos eletrônicos utilizando recursos de som e vídeo que fascinavam os jovens. O período é de explosão da indústria de software, calcada especialmente nos games. Como era programas ainda rudimentares, o esforço para sua criação podia ser desenvolvido às vezes por um único indivíduo. Levy conta histórias que envolvem a crescente profissionalização dessa indústria, cujos profissionais muitas vezes eram adolescentes que batiam na porta das empresas produtoras e distribuidoras com um game praticamente pronto. Em comum com os outros hackers, os “hackers dos games” teriam a paixão pelo trabalho criativo; a abordagem não tradicional em relação aos horários de trabalho; a cultura do compartilhamento e de liberdade de expressão; o sentimento de satisfação pela capacidade de controlar a máquina e por obter respostas lógicas aos comandos apresentados a ela; e uma atitude um tanto desinteressada com relação ao dinheiro. No campo pessoal, esses novos hackers seriam semelhantes aos antigos, nerds com paixão pela ficção científica, jogos de fantasia e inabilidade social. Com a explosão do negócio dos games, essa nova geração passa a ter que lidar mais fortemente com as contradições entre uma cultura de compartilhamento de códigos e troca livre de informações e o fato de

que a estrutura do negócio passa a depender de mecanismos de controle e proteção contra a cópia dos jogos.

Nesse sentido, a primeira manifestação de aversão a essa cultura da livre troca de software data ainda da geração anterior. Trata-se de um texto escrito por Bill Gates e intitulado “Open Letter to Hobbyists”, publicada em 1975 na revista *Popular Electronics*, um dos principais veículos dos aficcionados em computação dos anos 1970. Nela, Gates, então um jovem de 21 anos, queixa-se da violação de seus direitos que estaria acontecendo quando da cópia e troca de software, acusando os “hobbyists” de roubo. Gates e seus outros dois sócios, Paul Allen e Monte Davidoff, adaptaram a linguagem BASIC, uma linguagem de programação bastante simples, para o primeiro computador doméstico, o Altair, e recebiam direitos a cada cópia vendida. A carta contribuiu simbolicamente para que Gates fosse visto por alguns, nos anos seguintes, como um anti-hacker.

Como se tornar um hacker

Embora o termo hacker tenha sido associado, a partir de meados dos anos 80 com roubos de senhas e atividades ilícitas praticadas com o computador, os anos 90, além explosão da internet, assistem também ao crescimento em termos de reconhecimento público do que seria um produto exemplar e bóia de salvação da cultura hacker. O final do livro de Levy contém um epílogo, dedicado quase inteiramente a Richard Stallman, então intitulado “o último dos verdadeiros hackers”, em que é anunciada a intenção de Stallman de construir um sistema inteiramente livre, o GNU, um código aberto, em que todas as contribuições feitas a ele também seriam livres. Stallman é retratado como pesaroso com o que seria o fim da cultura hacker, do espírito do laboratório de inteligência artificial do

MIT, instituição que ele estava abandonando para dedicar-se ao GNU. Em meados dos anos 1990, o GNU se tornaria um grande sucesso ao receber a contribuição decisiva do kernel idealizado por Linus Torvalds, que por sua vez passaria a receber progressiva atenção da mídia.

Toda essa atenção significou também um resgate – e transformação – do termo hacker. Os novos adeptos do software livre passaram a ser os novos combatentes contra conotação negativa do termo, apontando o tipo por ele descrito como o guia comportamental e ético para todo o movimento. Nesse momento de fluxo de novos adeptos, então fascinados com o novo sistema operacional, Eric Raymond publica seu guia na internet “How to Become a Hacker” (veja anexo). A publicação original é de 1996, mas o texto sofre contínuas adaptações, traduções e acréscimos. Raymond se torna mais conhecido no mundo do software livre – que vive ainda uma fase bastante embrionária, especialmente no Brasil – a partir de 1998, quando da publicação de *A Catedral e o Bazar*. É desse mesmo ano a tradução de seu texto sobre os hackers para o português, que uso como referência.

O texto de Raymond é frequentemente citado quando há debates sobre o termo hacker, sendo usado para confirmar ou refutar certas posições. Em especial, o texto é especialmente interessante por enfatizar determinados pontos da imagem divulgada por Levy e dar menor relevo a outros. Como já dito, Raymond é uma figura proeminente e extremamente atuante, agindo de acordo com uma visão política bastante consolidada e em muitos pontos contraditória com a posição de Richard Stallman. Anarquista-libertário de viés conservador, Raymond parte da perspectiva de que a interação entre o software livre e a indústria é o melhor meio para propagar o estilo bazar de produção de software, que seria o meio mais eficiente para se produzir software de boa qualidade. “Como se Tornar um Hacker” vai além de outros textos por não só buscar uma definição para o termos, uma descrição do que seria a essência do hacker; isso é algo subjacente, mas o documento se configura mais como um manual, com orientações a respeito de comportamentos que permitiriam a um indivíduo ser considerado um hacker.

Pode-se afirmar que o documento se divide em três eixos de recomendações: qual seria (ou deve ser) a atitude hacker, quais seus fundamentos em termos de valores; que habilidades um hacker deve desenvolver ou possuir; e o que fazer para melhorar seu status entre os hackers. Na seção que antecede os três eixos, como introdução, Raymond afirma alguns dos principais feitos hackers (“Hackers construíram a Internet. Hackers fizeram do sistema operacional Unix⁸⁵ o que ele é hoje. Hackers mantém a Usenet. Hackers fazem a World Wide Web funcionar.”), afirma que a mentalidade hacker não é exclusiva à informática – embora esta seja seu objeto naquele texto –, e faz a distinção entre hackers e crackers (“hackers constroem coisas, crackers as destroem”).

O eixo das recomendações sobre as atitudes e valores contém cinco itens que apontam especialmente para a relação que os hackers deveriam estabelecer entre seu esforço de trabalho, motivação intelectual e diversão. Mas antes, Raymond fala sobre como seria necessário interiorizar esses valores, em um treinamento mental.

“Hackers resolvem problemas e constroem coisas, e acreditam na liberdade e na ajuda mútua voluntária. Para ser aceito como um hacker, você tem que se comportar de acordo com essa atitude. E para se comportar de acordo com essa atitude, você tem que realmente acreditar nessa atitude. (...) Assim como em todas as artes criativas, o modo mais efetivo para se tornar um mestre é imitar a mentalidade dos mestres – não só intelectualmente como emocionalmente também.”

Os cinco itens abaixo deveriam ser repetidos “até que que você acredite” neles:

- 1.O mundo está repleto de problemas fascinantes esperando para ser resolvidos.
- 2.Não se deve resolver o mesmo problema duas vezes.
- 3.Tédio e trabalho repetitivo são nocivos.
- 4.Liberdade é uma coisa boa.
- 5.Atitude não substitui competência.

Os três primeiros pontos giram em torno de uma relação específica entre o hacker e seu mundo

⁸⁵ O Unix foi a base para os sistemas operacionais livres. O GNU foi, inicialmente, um projeto de reescrita do código Unix.

de trabalho intelectual. A atividade é vista como uma brincadeira criativa, em que há diversas coisas a serem descobertas e desvendadas no mundo, e esse fazer intelectual não deve ser redundante, e sim compartilhado entre os diversos hackers. Repetir o mesmo trabalho seria um esforço não só tedioso para quem o desempenha como um desperdício de talento nocivo também para os não hackers. O hacker – assim como “as pessoas criativas em geral” – seria, de certa forma, um ser especial dada a sua habilidade na resolução de problemas. “Mentes criativas são um recurso valioso e limitado” e quando elas deixam de empregar seu tempo na resolução de problemas o mundo perde. “Tédio e trabalho repetitivo não são apenas desagradáveis, mas nocivos também”, diz Raymond.

O item 4 tem um funcionamento acessório nessa lógica de evitar o desperdício, dado que, para se evitar o trabalho redundante, as soluções precisariam ser intercambiadas e compartilhadas livremente. Mas antes Raymond fala sobre a valorização da liberdade do ponto de vista do combate ao autoritarismo, em que este funcionaria como algo que potencialmente leva o hacker a não orientar a sua atenção para um problema que genuinamente o fascina – assim, seria desperdiçado tempo e energia hacker, como referido anteriormente. O problema da liberdade é colocado de forma ampla, tanto do ponto de vista do indivíduo impedido de gerenciar a si mesmo e sua energia criativa, quanto do ponto de vista do bloqueio à comunicação, que levaria ao esforço repetitivo de se buscar uma solução já encontrada.

É interessante como Raymond dá uma perspectiva para o problema que é consoante com sua visão política e libertária sobre o mundo. Os indivíduos, em especial essa elite intelectual que seriam os hackers, são melhor administrados por si mesmos do que por qualquer autoridade externa. Ao mesmo tempo, agindo dessa forma e tendo liberdade para compartilhar, os hackers garantem uma distribuição não-organizada mas ótima de esforços não repetidos. Raymond formula a luta pela liberdade do ponto de vista do esforço individual de resistência: “...para se comportar como um hacker, você tem que

desenvolver uma hostilidade instintiva à censura, ao segredo, e ao uso da força ou mentira para compelir adultos responsáveis”.

Steven Levy, ao falar da “ética hacker”, também cita o que seriam o item 4 de Raymond, mas desdobrado em dois pontos: “All information should be free” e “Mistrust Authority Promote Decentralization”. No primeiro ponto, a racionalização vai na mesma linha dos itens 1, 2 e 3 de Raymond, ou seja, a informação deve ser livre para que não seja necessário reinventar a roda. Mas não porque esse seja um exercício pouco produtivo e sim porque é muito mais difícil se encontrar uma solução quando há carência de informações do que quando não há. Já “Mistrust Authority Promote Decentralization” não se refere a uma questão de liberdade do indivíduo frente a uma corporação ou a um Estado controlador – como parece ver Raymond – mas há um contexto específico do ambiente da informática até os anos 1980, dominado pela IBM e por padrões de relacionamento entre programadores e os então escassos computadores burocratizados. Trata-se da resistência a uma organização do tempo de uso das então grandes máquinas, muito pouco acessíveis. Essas regras eram dadas por critérios vistos como arbitrários e excessivamente práticos, tendo em vista atividades planejadas com antecipação, sem abertura para a interação criativa e improvisada dos “verdadeiros hackers” de então.

O item 5 de Raymond, “Atitude não substitui competência”, liga-se lateralmente ao “hands-on imperative” citado por Levy. Esse imperativo mão na massa, como descrito por Levy, refere-se ao acesso irrestrito aos sistemas para fim de conhecimento. Mas para Raymond a ideia de fundo é o valor a de um trabalho que é mais prático-demonstrativo do que teórico. Ele refuta o prestígio que não derive de uma ação prática e dá cor ao esforço disciplinado, diligente: “Para se tornar um hacker é necessário inteligência, prática, dedicação, e trabalho duro. (...) Se você reverenciar competência, gostará de desenvolvê-la em si mesmo – o trabalho duro e dedicação se tornará uma espécie de um intenso jogo,

ao invés de trabalho repetitivo.” É algo um tanto distante da descrição da relação com o trabalho dos “verdadeiros hackers” feita por Levy, em que esse trabalho duro, por mais que existisse, não era visto dessa forma, e sim como uma atividade que absorvia o hacker ao ponto da compulsão.

Em seguida, Raymond fala sobre as habilidades que um hacker deveria desenvolver. Elas são três:

1. Aprenda a programar.
2. Pegue um dos Unixes livres e aprenda a mexer.
3. Aprenda a usar a World Wide Web e escrever em HTML.

Raymond procura convencer os futuros hackers de que a capacidade de ser hacker é melhor adquirida utilizando-se linguagens e códigos abertos, que permitiriam um estudo mais fácil de como os programas são feitos e possibilitam a alteração de sua programação por aqueles que estão aprendendo. A referência a “Unixes livres” não é fortuita. Ele está procurando incluir no rol dos programas “de hacker” não apenas o kernel Linux que funciona em conjunto com o sistema GNU (este idealizado por Richard Stallman), mas todas as outras variações igualmente livres, como o FreeBSD. E polariza com a Microsoft: “Você terá ferramentas de programação (incluindo C, Lisp e Perl) melhores do qualquer sistema operacional da Microsoft pode sonhar em ter, você se divertirá, e irá absorver mais conhecimento do que perceber, até que você olhará para trás como um mestre hacker.”

Então Raymond passa a discutir o “status na cultura hacker”, igualmente fazendo recomendações pontuais. Mas, antes, ele faz considerações interessantes sobre como o status seria um dos motores da motivação. “Como a maioria das culturas sem economia monetária, a do hacker se baseia em reputação”, diz ele. “Você está tentando resolver problemas interessantes, mas quão interessantes eles são, e se suas soluções são realmente boas, é algo que somente seus iguais ou superiores tecnicamente são normalmente capazes de julgar.” Segundo Raymond, o status se daria pela

avaliação que os pares fazem das habilidades do hacker. “Quando você joga o jogo do hacker, você aprende a marcar pontos principalmente pelo que outros hackers pensam da sua habilidade (por isso você não é hacker até que outros hackers lhe chamem assim).” Aquele que é melhor julgado por seus feitos técnicos – a habilidade demonstrada na prática, enfatize-se – por membros que já tenham boa reputação, teria um maior status.

Ao mesmo tempo, haveria um altruísmo de cunho egoísta, interessado em aumentar a reputação: “Especificamente, a cultura hacker é o que os antropologistas chamam de cultura de doação. Você ganha status e reputação não por dominar outras pessoas, nem por ser bonito, nem por ter coisas que as pessoas querem, mas sim por doar coisas. Especificamente, por doar seu tempo, sua criatividade, e os resultados de sua habilidade.” O compartilhamento reconhecidamente não seria motivado por uma contribuição desinteressada mas teria um fundo de interesse no incremento do status individual. Os hackers acabariam por alimentar e incentivar o crescimento de códigos comuns, livres, tendo em vista a ascensão em termos de prestígio junto a outros hackers. Esse ponto é particularmente relevante e marca uma inflexão com a “ética hacker” como descrita por Levy. Refletindo ou não algo verificável, o ponto é a legitimação de uma ação egoísta e interessada em lugar de uma justificativa moral e ética como: contribuir para um sistema que seja de usufruto comum.

Os itens listados por Raymond para ser “respeitado por hackers” são:

1. Escrever open-source software.
2. Ajude a testar e depurar open-source software
3. Publique informação útil.
4. Ajude a manter a infra-estrutura funcionando.
5. Sirva a cultura hacker em si.

Embora Raymond não deixe isso claro, percebe-se que as recomendações estão em nível hierárquico decrescente. Escrever “open source” reflete uma contribuição entendida como mais

significativa do que publicar informação útil (manter manuais técnicos, tutoriais) ou servir “a cultura hacker em si”.

Ao final, ele faz um alerta sobre a busca por esse status que, embora possa ser consciente a ponto de existirem recomendações a serem seguidas como as que ele faz, não deve ser pública ou de algum modo ostensiva: “A cultura hacker não têm líderes, mas têm seus heróis culturais, "chefes tribais", historiadores e porta-vozes. Depois de ter passado tempo suficiente nas trincheiras, você pode ser tornar um desses. Cuidado: hackers desconfiam de egos espalhafatosos em seus "chefes tribais", então procurar visivelmente por esse tipo de fama é perigoso. Ao invés de se esforçar pela fama, você tem que de certo modo se posicionar de modo que ela "caia" em você, e então ser modesto e cortês sobre seu status.”

A ética hacker, o espírito da era da informação e o trabalho

Vimos até agora como o termo e a ideia de hacker ganham popularidade no início da década de 1980, quando da primeira onda de popularização dos microcomputadores, máquinas logo associadas às primeiras gerações de hackers. Logo, no entanto, o termo passa a ser associado com a atividade criminosa de invasão dos computadores, sentido que ganha o imaginário popular também porque os hackers desde um primeiro momento são vistos como indivíduos de raras habilidades. No início da década de 1990, quando surgem as primeiras versões do Linux, e o sistema operacional completo GNU/Linux ganha gradual aceitação, em especial nos círculos mais especializados, o sentido original de hacker passa a ser resgatado. O sistema é anunciado como “de hackers” ou “feito por hackers”, e nessa cultura estaria explicada a prática de se trocar códigos e programas livremente, com um

melhoramento contínuo dos programas, prática bastante diferente do modelo comercial em curso até então, em que os programas são vendidos em caixas, como objetos acabados e não-alteráveis. O crescente interesse pelo sistema ajuda a explicar guias como o escrito por Eric Raymond, em que tornar-se hacker transforma-se em um objetivo desejável. Além disso, no esforço de popularização dos sistemas livres era preciso retirar o sentido negativo da palavra hacker, de modo a não despertar ressalvas em futuros usuários. É bastante difícil convencer um indivíduo comum a usar um sistema cujo código não é secreto, o desenvolvimento é coletivo e não garantido por nenhuma empresa e ainda cujo desenvolvedor-modelo era associado com práticas criminosas.

Nesse contexto, ganha proeminência a figura de Linus Torvalds, o desenvolvedor responsável pelo Linux. Sua história, e a rápida adoção do sistema, fascina a mídia e permitem a identificação de um novo grupo de hackers. Torvalds é finlandês e cresceu distante da cultura hacker estadunidense, seja dos “verdadeiros hackers” do MIT, seja da nova geração de hackers do Vale do Silício. Ainda assim, sua prática de compartilhar o código e receber contribuições de voluntários para o kernel do Linux aparentemente guarda forte ligação com o que seria a cultura hacker. O Linux, e Linus, tornam-se símbolos principalmente após o estouro da chamada “bolha da Internet”, a decadência após a escalada especulativa no valor das ações de empresas de tecnologia, iniciada em 1998 e que atinge seu pico em 2000. Novas empresas, como a Red Hat, SUSE e outras passam a fazer do Linux um produto a ser inserido no mercado comum de informática, porém a partir de um novo modelo de negócio, já que o código é livre. Em tempos de crise, a alternativa atrai progressiva atenção.

É principalmente a partir da experiência de Torvalds, e num contexto pós-boom da Internet, que Pekka Himanen tenta entender os hackers. Em *A Ética dos Hackers e o Espírito da Era da Informação* (Himanem, 2001), o autor procura compatibilizar a ideia de Era da Informação, desenvolvida por Manuel Castells, e o que seria uma nova ética de um novo capitalismo, agora baseado na informação. A

comparação seria com a ética protestante, que ainda seria predominante nos tempos atuais, inclusive nas empresas envolvidas com o negócio da informação. Ele usa a literatura de “desenvolvimento pessoal”⁸⁶ para explorar a configuração da ética protestante hoje e a descreve como retratada de forma exagerada nesse tipo de literatura e dada a partir de sete valores: objetivos, otimização, flexibilização, estabilidade, diligência, economia e contabilização de resultados.

Mas esses não seriam valores hackers, os hackers fariam uma ruptura, principalmente na relação entre tempo, dinheiro, trabalho e diversão. Himanem apóia-se principalmente em Torvalds, que escreve o prefácio do livro, para afirmar o que seriam outros valores e que poderiam representar uma alternativa na Era da Informação. Em seu texto, Torvalds afirma três motivações humanas que seriam básicas – sobrevivência, vida social e diversão – e explica o trabalho do hacker a partir desta última, um tipo de diversão que ele afirma ser com D maiúsculo, aquela que alguém obtém jogando xadrez, pintando ou com "a ginástica mental que envolve tratar de explicar o universo" (p.14). Himanem interpreta essa Diversão como “paixão”, incorporando explicitamente a palavra usada por Eric Raymond para descrever as motivações hackers.

Desse jogo entre trabalho, diversão e paixão, Himanem elabora a ética hacker, muito mais herdeira de valores da academia, das universidades, do que da ética protestante, esta cuja base seria a acumulação de dinheiro. Ele postula sete valores hackers: paixão, liberdade, valor social, abertura, atividade, cuidar e criatividade. Paixão e liberdade estariam sob o eixo de uma ética do trabalho, e fariam referência à motivação principal do hacker, como afirmada por Torvalds, e à capacidade de organizar a si mesmo, o tempo dedicado ao trabalho. A frase “tempo é dinheiro” seria substituída por “tempo é minha vida” (p.47). Já valor social e abertura estariam dentro do eixo da ética do dinheiro, ou

⁸⁶ Utilizei a versão em português para o livro, que contém perceptíveis erros de tradução. No caso, o termo “desenvolvimento pessoal” recebeu uma tradução literal, é grafado no original como “personal development”. Porém, a literatura a que o autor parece se referir é a de auto-ajuda empresarial.

seja, o dinheiro não é desprezado, mas deixaria de ser um fim em si mesmo, sendo veículo para o compartilhamento da paixão com os outros. Esse é um ponto particularmente interessante da teoria de Himanem, pois é uma referência direta às então emergente empresas que lidam com software livre. O que ele está postulando é que a abertura não seria uma decisão de negócios tomadas por empresas que enxergam como uma vantagem competitiva no mundo da informática, mas algo que os “hackers capitalistas” são levados a fazer pelo desejo de compartilharem suas criações de maneira social. O desenvolvimento dado a essa questão pelo autor é reticente, preferindo apenas refutar que a tal abertura tenha algo de comunismo⁸⁷. Voltando para os valores hackers, haveria ainda a relação dos hackers com a rede, descritas pelas palavras atividade e cuidar, entendidas como atividade na rede e cuidar da rede. Atividade expressaria a preocupação dos hackers com a liberdade de expressão, a liberdade individual e uma valorização do buscar pela paixão individual. Já o cuidar relacionaria-se com uma preocupação com o próximo, mas voltada à inclusão das pessoas na rede e em como a rede poderia ajudar as pessoas a vencerem as necessidades para a sobrevivência. O último valor, que Himanem não coloca sob nenhum eixo e que ele classifica como o mais importantes para os hackers, seria a criatividade: “A utilização imaginativa das habilidades de cada um, a surpreendente superação contínua de si mesmo, e a doação ao mundo de uma nova contribuição genuinamente valiosa”(p.127).

É interessante como a obra de Himanem acaba por ter um duplo caráter de predição e de projeto. Segundo ele, estaríamos somente começando a entender essa nova ética hacker, que tenderia se tornar regra na Era da Informação em lugar da ética protestante. Ao mesmo tempo, a ética hacker é tratada muitas vezes de um modo acabado, o que tem por efeito se assemelhar a orientações para a saída de uma ética protestante já não mais funcional em um capitalismo que estaria deixando de ser industrial.

⁸⁷ Segundo ele, o comunismo também comungaria dos valores protestantes. (Himanem, 2001:65)

O ser hacker em discussão

A definição de hacker é algo que motiva o debate entre os adeptos do software livre, principalmente por se permitir, a partir dessa definição, marcar quem está dentro ou quem está fora do movimento ou, mais especificamente, definir quem é um legítimo porta-voz. Mesmo que se trate de alguém pouco ativo dentro do grupo, ser definido ou visto como hacker implica em certa autoridade sobre os assuntos do software livre.

É o que se permite perceber no debate que aconteceu em junho de 2008 pela lista de discussão PSL-Brasil em torno da propriedade de o então ministro da Cultura, Gilberto Gil, ter se intitulado “ministro hacker”. A PSL-Brasil é uma lista pensada originalmente para congrega membros dos diversos Projetos Software Livres brasileiros, e que com o passar do tempo foi se tornando o principal ambiente virtual de discussão política do tema⁸⁸, tornando-se uma lista aberta e de conteúdo público. O debate envolveu o envio de mais de quarenta mensagens de diferentes autores – alguns repetidos – se deu na *thread*⁸⁹ “OT: Gilberto Gil: 'Sou hacker. Um ministro hacker’”. O início se deu pelo envio de uma mensagem contendo a seguinte notícia, publicada originalmente no website Estadão:

Ele assume sem pudor: “Sou hacker. Sou um ministro hacker. Sou um cantor hacker.” Fonte: <http://www.softwarelivre.org/news/11554>

Aos 65 anos, o ministro da Cultura, cantor e compositor Gilberto Gil resolveu fazer de sua música um “manifesto político” pela democracia digital. Na faixa-título de seu novo CD, Banda Larga Cordel, a ser lançado amanhã, ele brada:

⁸⁸ A lista reúne centenas de membros do movimento, muitos deles proeminentes, mas apenas alguns poucos envolvem-se nas discussões. Esses participantes ativos não são necessariamente os membros de maior prestígio, embora alguns o sejam. Outros, apenas leem as discussões, que são comentadas em particular e servem como uma espécie de termômetro.

⁸⁹ *Thread* é o nome dado à discussão incremental de determinado tema em uma lista de discussão, com mensagens marcadas pelo mesmo subject/assunto. Ela é composta de respostas acumuladas que formam uma discussão, em que os participantes editam trechos da mensagem anterior para respondê-las.

“Banda larga mais democratizada ou então não adianta nada. Os problemas não terão solução”.

“Sempre fui politizado, mas meu novo disco está muito influenciado pela função ministerial”, disse ao Link. Criador do neologismo “bandalargar” (espalhar a banda larga), Gil se tornou uma espécie de garoto-propaganda e ativista do governo para assuntos tecnológicos. Em 2005, fez dueto com o famoso ativista de software livre Richard Stallman em evento da ONU. Gil ficou ao violão enquanto Stallman desafinava: “Junte-se a nós e compartilhe o software. Você será livre. Você será hacker”.

Na Campus Party deste ano, o ministro acompanhou o instrumento digital ReacTable com falsetes de “u-hu” e roubou a cena do robô que abriria a megafesta nerd ao falar as palavras “conteúdo livre”. Já em 2004, foi ovacionado no Fórum de Software Livre, em Porto Alegre, ao cantar a canção “Oslodum”, que disponibilizou sob a licença Creative Commons.

Fonte original: Link Estadão: http://www.link.estadao.com.br/index.cfm?id_conteudo=13952

Enviada às 21 horas, ainda naquela noite a mensagem recebe duas respostas, a primeira positiva e a segunda de questionamento⁹⁰.

Resposta 1:

Vi isto hj no Caderno LINK do Estadão.
Bom, né???

Resposta 2:

Do Jargon File:
"It is better to be described as a hacker by others than to describe oneself that way. Hackers consider themselves something of an elite (a meritocracy based on ability), though one to which new members are gladly welcome. There is thus a certain ego satisfaction to be had in identifying yourself as a hacker (but if you claim to be one and are not, you'll quickly be labeled bogus)."
Eu o chamaria de wannabe e olha lá.

A partir daí a discussão progride, eventualmente derivando para assuntos um pouco paralelos – a possível existência de um “advogado hacker”, por exemplo, que manterei fora desta análise – mas

⁹⁰ Para nenhuma das mensagens transcritas foram retiradas as saudações, apenas as assinaturas, quando presentes.

preservando como central o questionamento sobre propriedade de Gil declarar-se hacker.

A resposta número 2 busca a definição do termo no Jargon File, glossário organizado desde 1975, originalmente por Raphael Finkel, a partir de termos nascidos nos laboratórios de inteligência artificial do MIT e da Universidade de Stanford⁹¹. Nos anos seguintes, a ele são adicionadas contribuições sucessivas, incluindo de Richard Stallman. Em 1983, esse material é adaptado e é publicado no livro *The Hacker's Dictionary*. Depois de um hiato de sete anos, Eric Raymond volta a editar o material e o publica em livro em 1991, *The New Hacker's Dictionary*, tendo recebido sucessivas correções e atualizações em sua versão na internet. O trecho que o autor da resposta 2 escolhe faz parte apenas da versão atualizada. O interessante é que, tanto na versão original⁹² quanto na versão de Raymond, uma das definições, a de número 3, poderia se encaixar bem ao caso de Gil: “A person capable of appreciating hack value”. O autor prefere reforçar uma definição menos inclusiva (“It is better to be described as a hacker by others than to describe oneself that way (...) ... but if you claim to be one and are not, you'll quickly be labeled bogus) e que requer mais habilidade técnica (“a meritocracy based on ability”).

Em seguida, outro argumento contrário à ligação Gil-hacker – somado a uma alfinetada em Sérgio Amadeu, ativista e acadêmico que se notabilizou pela defesa ao software livre e diretamente envolvido na implantação de software livre no governo federal nos primeiros anos do governo Lula.

Resposta 3:

Pelo que tem lá na matéria é o Sérgio Amadeu quem o chama de hacker. Nada contra o Sérgio Amadeu, mas “Software Livre é o google”[1][2]⁹³ fala por si só. “Hacker é Gilberto Gil” está dentro dessa mesma linha.

⁹¹ Ver <http://www.catb.org/jargon/html/revision-history.html>

⁹² Ver <http://www.dourish.com/goodies/jargon.html>

⁹³ O autor da mensagem aponta dois links: [1] <http://www.youtube.com/watch?v=9s690u52SxM>; [2] <http://www.youtube.com/watch?v=GyM0tY3yHVQ>. Trata-se das duas partes da entrevista de Amadeu a Jô Soares

Amadeu foi entrevistado no *talk show Programa do Jô* em 2006. A repercussão da entrevista foi em geral negativa, com críticas à postura do entrevistador Jô Soares com relação ao assunto e ao desempenho de Amadeu em particular. “Software Livre é o google” é citado como exemplo de frase incorreta dita por Amadeu, que procurava afirmar a qualidade dos programas livres ao apontar seu uso pelo Google.

A resposta número 3 afirma que, por ter sido chamado de hacker por Amadeu, a auto-classificação de Gil seria algo impróprio. A frase “Software Livre é o google” seria um exemplo de desconhecimento ou imprecisão técnica por parte de Amadeu. O autor da resposta 2 concorda, dá a entender que Amadeu seria “pessoa não autorizada”, e jocosamente tenta encaixar Gil em uma categoria subalterna.

Resposta 4:

Tá. "hacker honorário", "café-com-leite" e "mascote da turma". O ministro pode escolher os três, se quiser, mas precisa devolver a carteirinha de hacker indevidamente emitida por pessoa não autorizada.

Surge então um protesto, que parte de um membro bastante proeminente do movimento, organizador histórico do Fórum Internacional de Software Livre, e frequentemente classificado – e criticado por isso – como um dos “políticos” do movimento brasileiro⁹⁴.

Resposta 5:

de forma nenhuma!!

Só o que falta agora é quererem emitir certificado LPI [o Linux Professional Institute emite certificados profissionais, reconhecidos por empresas, que servem como “diplomas” em Linux] para hacker ou “alguém” determinar quem pode ser chamado de hacker e quem não pode.

Eu aprendi com meus amigos hacker's nos anos 90 que um hacker é um

⁹⁴ Ver no capítulo 3 discussão sobre a divisão entre “políticos” e “hackers” no contexto interno da organização do Fisl.

"fussador", um cara que não se contenta apenas em usar as tecnologias mas em penetrar a fundo nelas. Em construí-las e reconstruí-las. Um termo antigo que vem dos radioamadores que montavam seus próprios transmissores...

Pra mim o Gil é um Hacker da cultura...

respeito que pense o contrário.

Mas não me venham com essa de emitir certificações para hackers ou "regulamentar a profissão"..he he ...

A resposta é uma defesa de um sentido mais amplo para a palavra, primeiro afirmando seu princípio descritor de uma atitude e em seguida retirando-a de um exclusivo contexto técnico. Ao mesmo tempo, é interessante por apontar, ainda que involuntariamente, uma contradição da “ética hacker” como normalmente é descrita. Hackers teriam aversão ao autoritarismo e à imposição de classificações, porém somente aqueles que já são reconhecidos como hackers poderiam reconhecidamente apontar outros hackers. Ao reforçar a comparação com uma certificação empresarial o autor tenta inviabilizar os julgamentos sobre quem pode e quem não pode chamar a si próprio de hacker.

O autor das respostas número 2 e 4 então volta à carga, agora ironizando a ideia de “hacker da cultura”.

Resposta 6:

Se formos nessa linha todo mundo é. Meus gatos desmontam um sofá para dormir debaixo das almofadas. Eles vão fundo na tecnologia do mobiliário e a usam de acordo com seus próprios critérios, de uma forma que nós, os construtores de sofás, normalmente não fazemos.

Ainda que, quando criança, eu montava grandes fortalezas na sala da minha avó usando almofadas de sofá como paredes, as mesas como estrutura e as almofadas pequenas como munição. Assim como meus gatos são hoje, eu era um "hacker de sofás".

O perigo aí é o termo "hacker" perder ainda mais do significado.

Mas eu concedo uma: Ele pode ficar com a carteirinha, mas só se ele souber explicar claramente porque ele merece ficar com ela.

A resposta vai na linha de defender alguma restrição ao sentido da palavra, pois senão o significado se perderia. O uso de analogias, às vezes com viés humorístico, é algo muito frequente na PSL-Brasil e em outras listas que discutem o software livre de um ponto de vista não técnico. A mensagem não agrega muito em termos de conteúdo, serve mais como reafirmação, ainda que em tom de brincadeira, do status do argumentador como definidor de quem é e quem não é hacker.

Então um membro da lista argumenta favoravelmente pela definição de Gil como “ hacker no domínio musical”, embora afirme o caráter técnico e relativo à informática do termo. Em seguida, outro vai na mesma linha, afirmando a existência de uma definição que seria a “correta” e tentando encaixar Gil em outra categoria.

Resposta 7:

Você quer dizer que o Gilberto Gil penetrou fundo nas tecnologias da cultura, as construiu e reconstruiu, não se contentando em apenas usá-las?

Eu acho isso meio forçado, já que o termo Hacker é associado à tecnologia, e um cara que navega na internet e joga guitar hero não se qualifica.

Isso que você vê no ministro é o espírito revolucionário, e não o espírito hacker. Voltaire[1] não era hacker, e foi bem mais fundo dentro das tecnologias que o ministro. Se até Voltaire ficou conhecido como revolucionário, e não hacker, porque não se contentam em chamar o ministro de revolucionário e deixem o termo hacker ser aplicado corretamente?

E não era pelo fato de no tempo de Voltaire o termo "hacker" não haver sido cunhado, pois existem trabalhos hackers reconhecidos na antiga Grécia[2]. Existem certos critérios para que uma obra seja considerada "coisa de hacker".

links:

[1] <http://pt.wikipedia.org/wiki/Voltaire>

[2] <http://www.tuxdeluxe.org/node/104>

Na resposta acima, é interessante a escolha de Voltaire como contraponto, em especial se

examinarmos o *link* oferecido como referência, da Wikipedia em português, em que Voltaire é descrito logo na primeira frase do verbete como: “...Voltaire, foi um escritor, ensaísta, deísta e filósofo iluminista francês conhecido pela sua perspicácia e esprituosidade na defesa das liberdades civis, inclusive liberdade religiosa e livre comércio”⁹⁵. O argumento é que mesmo Voltaire, um defensor das liberdades civis, religiosa e do comércio – algo que estaria consonância com o software livre – não deveria ser considerado hacker. O trabalho hacker feito pelos gregos, que é citado em sequência, seria uma primeira calculadora mecânica, utilizada para cálculos astronômicos⁹⁶.

Na mesma linha de busca de personagens históricos outro membro então argumenta.

Resposta 8:

Eu gosto de Da Vinci como Hacker e Santos Dumont como precursor brasileiro do Conhecimento Livre. :-P

Gilberto Gil... hacker... forçou a amizade, de boa.

Mas, hoje em dia é Web 2.0, Web 3.0, Die hard 4.0....

Que se exploda tudo

Retomando a linha do debate, então outro membro recorre ao livro de Himanem sobre a ética dos hackers.

Resposta 9:

No livro “*A Ética dos Hackers e o Espírito da Era da Informação”, *isto é colocado um pouco diferente, no sentido de hacker ser alguém realmente apaixonado pelo que faz e faz isso muito bem. De qualquer forma acho bom o ministro afirmar ser hacker, pois leva as pessoas a reverem seu conceito sobre hacker, elas vão pensar: “Se o Gil é hacker e não é do mal, então quem são aqueles caras que invadem sistemas e roubam cartão de crédito na internet?”

⁹⁵ Neste link, Wikipedia em junho de 2008 <http://pt.wikipedia.org/w/index.php?title=Voltaire&oldid=15740788> versão lida pelo respondente.

⁹⁶ http://en.wikipedia.org/wiki/Antikythera_mechanism

A Ética dos Hackers e o Espírito da Era da Informação funciona tanto como a citação de uma análise teórica de rigor – algo com que outros membros demonstram concordância, em mensagens posteriores na mesma *thread* – como para afirmar um sentido mais amplo para o termo, em especial a ideia de paixão envolvida na relação dos hackers com o trabalho. Mas além disso, a mensagem agrega uma razão pragmática para a ampliação do escopo de indivíduos classificados como hackers, a quebra da associação entre hackers e a atividade criminosa. Em uma mensagem posterior, outro membro vai concordar: “É como já disseram, acho bom que ele faça isso pra diminuir um pouco a ideia de que hacker == ladrão.”

O autor da resposta 5, o ativista político proeminente, então retorna ao debate, expressando concordância com alguns argumentos levantados e afirmando ser importante a popularização do termo hacker como algo amplo e positivo.

Resposta 10:

“...para mim a síntese principal está aqui. O Gil como figura pública, artista, músico, Ministro e o-que-ele-é, recuperando o sentido do termo hacker perante milhões de pessoas e assumindo as causas das liberdades na sociedade em rede. Fazendo retumbar na rede mundial, liderando desde nosso país, com nosso sotaque baiano-tche, os recados fundamentais da comunidade software livre e da cultura livre.

Eu me sinto maior com isso.”

A expressão que mais chama a atenção é “liderando desde nosso país, com nosso sotaque baiano-tche, os recados fundamentais da comunidade software livre e da cultura livre”. O sotaque baiano, do próprio Gil, aparece conjugado com um sotaque “tche”, alusão ao Rio Grande do Sul e a um papel de liderança assumido por alguns membros do movimento software livre do estado, como o próprio autor da mensagem. Além disso, o “hacker da cultura” Gil seria representante não apenas do software livre, movimento ligado ao software e suas licenças, mas também da “cultura livre”,

movimento mais amplo e de questionamento das formas atuais de direito autoral.

A ideia de representação perante o mundo é especialmente questionada.

Resposta 11:

Acho que sua interpretação está errada. Um hacker tem aversão a títulos. Você poderia justificá-lo como hacker se tivesse citado os hacks que ele teria feito, em prol da liberdade do conhecimento (...)

O hacker promove a descentralização. Isso significa que o hacker não precisa de "recados", nem de representantes. O hacker é autônomo, de certa forma anárquico.

Isto significa que as causas do Software Livre (ou Cultura Livre) não podem ser representadas ou ter porta-voz (de recados). Um senso de representação contradiz a própria definição de liberdade da EH [ética hacker]. Por outro lado, podemos dizer que o Gil é um apoiador ou ativista da causa.

(...)

Por outro lado, gostaria de analisar o perfil hacker do Min. Gil, em função desta frase do "Hacker Howto", escrito pelo ESR [Eric Raymond]: "Especificamente, a cultura hacker é o que os antropologistas chamam de cultura de doação. Você ganha status e reputação não por dominar outras pessoas, nem por ser bonito, nem por ter coisas que as pessoas querem, mas sim por doar coisas. Especificamente, por doar seu tempo, sua criatividade, e os resultados de sua habilidade."

Quais foram as doações do Min. Gil, sem estar exercendo suas obrigações de seu cargo ou sem ganhar nada em troca? Quais foram outras doações, sem ter ganho visibilidade na mídia? Mesmo a própria Oslodum [música lançada por Gil com licença Creative Commons] é licenciada sob uma licença restritiva. (...) [Diz a licença que] "Cópias não-comerciais e distribuição (como troca de arquivos) do trabalho inteiro são permitidas." – isto significa que um camelô não poderia ganhar por um CD que distribuísse esta música.

Ainda acho que como disseram, o Min. Gil é um wannabe :-|

A resposta 11 tem dois sentidos. O primeiro é questionar o comprometimento de Gil com a própria causa do software livre e da cultura livre em geral. Gil fez vários pronunciamentos em favor de ambas, porém não demonstraria comprometimento real, principalmente no que tange à sua obra.

“Oslo dum” utiliza uma licença do grupo Creative Commons, equivalentes às licenças de software abertas no mundo da arte, porém adota uma de tipo bastante restritivo, que não permite as práticas compreendidas pela expressão software livre (liberdade de uso, cópia, estudo e modificação). O texto “Como se Tornar um Hacker”, de Eric Raymond, é citado para reforçar a ideia de um altruísmo que Gil não teria demonstrado. É uma argumentação calcada no que seriam os princípios culturais de uma “ética hacker” – com os quais Gil não demonstraria comungar em sua ação – e não na exigência de habilidades técnicas.

O segundo sentido liga-se a questões políticas do movimento brasileiro. Não é o papel de Gil como representante que está efetivamente sendo questionado – ou não somente o dele – mas o do membro da lista que afirmou o “sotaque baiano-tche”, que postulou a existência de recados a serem levados ao mundo, de lideranças.

Conclusão

Coleman e Golub (2008), investigando o contexto estadunidense, escrevem um estudo antropológico da prática hacker e sua relação com o liberalismo e dividem-nos em três gêneros: um primeiro dedicado em especial à práticas de criptografia, tendo como preocupação as questões de privacidade; outro ligado ao software livre e que teria uma visão positiva e liberal sobre a liberdade, criando uma base tecnológica – as licenças livres de software e os programas livres em si – sobre a qual uma comunidade de hackers pudesse florescer; e um terceiro voltado à transgressão, à quebra de travas e proteções, notadamente aqueles que o movimento software livre tem chamado de crackers. Com isso, os autores pretendem ir além do usual dualismo entre uma visão vilanizada dos hackers como a

divulgada pela imprensa não especializada, e “visionários cujo utópico estilo de vida tecnológico teria o potencial de destruir as mazelas do capitalismo e da modernidade em geral”. Ao fazê-lo, eles advogam pelo pertencimento de outros grupos que não apenas o do software livre ao termo hacker, grupos que também apresentam-se e postulam serem hackers.

Não é o caso aqui de se afirmar ou refutar esse pertencimento, mas de refletir sobre como a articulação do termo, a escrita frequente de sua definição e o recontar de sua história estão ligados a questões de poder dentro do movimento software livre, servindo como norma para atitudes, posicionamentos, inclusões/exclusões e autoridade.

O argumento de Coleman e Golub (2008) é que os três gêneros de hackers por eles identificados tem como mesma base o liberalismo – entendido não como um corpo coerente de pensamento legal econômico e filosófico, mas como uma sensibilidade cultural que, em prática, está em constante negociação e reformulação –, porém em diálogo com correntes diversas. Mesmo entre os grupos *open* e *free* do software livre eles apontam o recurso a diferentes tradições: “Mas enquanto Stallman [referindo-se ao grupo *free*] desenha uma comunidade mantida por normas compartilhadas e valores, o OSS [open source software] busca pensadores como os iluministas escoceses, como Bernard Mandeville, que argumenta que o bem público vem dos vícios privados”. Enquanto Stallman entenderia a troca de códigos como uma liberdade equivalente à liberdade de expressão, o grupo *open* partiria da ideia de que a liberdade proporcionaria a forma mais eficiente de se fazer código, em que a busca egoística dos hackers por reputação e diversão seriam os motores – como indivíduos no mercado que, lutando por si, gerariam prosperidade coletiva.

Nesse sentido, pudemos acompanhar as inflexões específicas dadas por Raymond ao termo em seu manual de como se tornar hacker. Está lá não apenas uma descrição do que seria um hacker, mas uma orientação sobre como deveriam pensar e agir os novos membros, a que motivações deveriam

responder. Já no texto de Himanem (2001), temos a ética hacker como saída possível para um novo capitalismo que estaria em curso, baseado na informação.

No contexto brasileiro, percebe-se o recurso dos membros do movimento a essas fontes internacionais, que definiriam o termo e estabeleceriam as mesmas bases para um hacker global. Coleman e Golub falam da validade local – o escopo “americano” e anglo-europeu – dos gêneros do *hacking* que identificam, o que parece correto em especial se pensarmos nas diferenciações que postulam e na exposição específica dos estadunidenses à tradição liberal. Porém, percebemos no caso da discussão do hackerismo de Gilberto Gil, como essas são definições de peso no embate que se estabelece sobre o termo no contexto local.

Conclusões gerais

Os últimos dez anos foram de forte expansão do software livre, tanto em termos de práticas e discursos em favor das licenças livres de software e de uma “cultura do compartilhamento” como no sentido de ter se tornado uma realidade dominante no mercado de informática. O incremento da velocidade na Internet, a maior capacidade de processamento dos computadores e popularização dos dispositivos móveis de acesso à rede contribuíram decisivamente para uma mudança no modelo de negócios do mercado de informática, que progressivamente vêm sendo dominado pela comercialização de serviços agregados em lugar do licenciamento de programas instalados nos computadores pessoais. Na implantação dessa infra-estrutura, os software livres têm particular relevância, formando a base em que funcionam esses sistemas⁹⁷.

Como fenômeno econômico e de produção descentralizada, o software livre desde logo atraiu a atenção da comunidade científica. Mais recentemente, surgiram estudos sobre a dinâmica política e cultural do mesmo. Este trabalho foi construído nessa direção, investigar o software livre buscando ressaltar seus aspectos culturais, entendendo-os como força operativa nos embates e disputas políticas inerentes ao movimento.

Dessa forma, procurei inicialmente descrever a principal divisão política do software livre, distinguindo dois grupos majoritários em âmbito internacional, os quais representam algumas das ambiguidades do movimento. Sob o chapéu genérico da palavra liberdade constituíram-se os grupos *free* e *open*, que em certos momentos colocam-se como aliados, mas em outros são adversários

⁹⁷ A referência aqui é a o que tem sido chamado de “cloud computing”. Ver “The Importance of Free Software in the Cloud”, disponível em <http://www.workswithu.com/2009/10/13/the-importance-of-free-software-in-the-cloud/>, acessado em 17/11/2009.

políticos. Tentei apontar algumas das aproximações e diferenças dos grupos, atentando para o papel da ideologia enquanto força atrativa ou repulsora de adeptos *free* e *open*. Enquanto o grupo *free* reforça argumentos que levam em conta fundamentos morais de uma troca social de códigos de computador – o que é e o que não é justo, ético, etc – o grupo *open* fundamenta-se no que seria o método de produção de um software de melhor qualidade. A reboque dessa distinção colocam-se ainda outros fatores, como o maior distanciamento ou aproximação com as empresas e com o sistema econômico capitalista tradicional.

A partir dessa distinção, procurei entender como o software livre se insere no Brasil em um contexto político e social distinto do estadunidense, onde o movimento software livre se origina. Coleman (2004) aponta o software livre nos Estados Unidos – tanto o grupo *free* como o *open* – como eivado de um “agnosticismo político” e formado a partir de tradições liberais que o significam como um tipo de luta pela “liberdade de expressão”. No Brasil, porém, o software livre ganhou força inicial principalmente entre sindicalistas e militantes sociais, alguns ligados a partidos políticos, que o significaram com outros sentidos, sendo entendido como ferramenta de independência e autonomia nacional, justiça social, arma contra a globalização corporativa e o neoliberalismo, entre outras bandeiras da esquerda. Assim, o Brasil acabou constituindo um movimento software livre bastante ativo politicamente, ainda que pouco atuante na produção de softwares livres. No processo de constituição do software livre no Brasil o grupo *free* teve força particular, estabelecendo uma forte interação com movimentos sociais e com lideranças políticas à esquerda no espectro político e permitindo uma associação produtiva entre as ideias políticas desses grupos e a ideologia *free*.

Isso, porém, não significou o abandono da polarização entre *free* e *open* no Brasil. Ao contrário, ela parece ter se tornado ainda mais evidente. O Fórum Internacional de Software Livre, como procuro demonstrar, evidencia os conflitos a cada uma de suas edições anuais. Nessa disputa política que se dá

sobre o sentido do movimento, *free* e *open* afirmam seus posicionamentos também em relação ao cenário político tradicional, entre esquerda e direita. Acho importante ressaltar como os debates sobre o software livre não acontecem desconectados das ideias políticas que permeiam a sociedade em seu contexto histórico. Dessa forma, o *open*, ao posicionar-se diferencialmente ao grupo *free*, faz suas aproximações mais à direita, incorporando as visões neoliberais e traduzindo-as para as brigas internas do movimento. Assim, o grupo *open* tem defendido o caráter capitalista do software livre e se posicionado de maneira contrária a muitos dos esforços feitos pelo Estado na disseminação e promoção dos softwares livres. A popularização deveria ocorrer “naturalmente”, de acordo com a percepção do mercado, que os adotaria por serem programas de melhor qualidade.

O embate entre *free* e *open*, contudo, não se encontra parado no tempo e essa divisão política e ideológica é ressignificada e reinterpretada continuamente. É preciso ressaltar o fortalecimento recente das empresas de tecnologia *open*, que se tornaram um peso-pesado no ambiente do movimento software livre. Elas têm constituído seu negócio a partir de software livre e, da mesma forma, incorporam, formulam e transformam a ideologia do software livre, em especial em contato com o grupo *open*. Valores como a transparência, a descentralização da produção, uma postura aparentemente não gananciosa, as hierarquias flexíveis, a abertura, entre outros, passam a fazer parte da imagem dessas próprias empresas.

Barbrook (2005), ao comentar o que chama de *high-tech gift economy*, argumenta apontando o surgimento de uma nova economia, um misto de trocas capitalistas e não capitalistas no ciberespaço. O mercado, ao financiar o software livre, por exemplo, estaria patrocinando o anarco-comunismo na rede, a existência de trocas para além do Estado e das corporações. Ele afirma que o “*copyright* é protegido e quebrado. Os capitalistas se beneficiam de um lado e perdem de outro”. Naquele texto, escrito originalmente em 1998, ele posiciona o neoliberalismo e o que ele chama de “ideologia da Califórnia”

– a mistura, surgida no primeiro *boom* da internet, de tecno-utopias neoliberais dos anos 90 com uma postura anti-autoridade herdeira da contracultura dos anos 60 – como defensores das regras de propriedade intelectual vigentes.

Mais de uma década depois, acredito haver uma nova configuração dessas posições. O ponto não é mais a defesa das regras rígidas de propriedade intelectual, ao contrário, é o incentivo ao fluxo, à troca, à criação coletiva. As novas empresas buscam no movimento software livre códigos para suas operações comerciais e, eventualmente, funcionários para integrarem seus quadros. Trata-se de uma nova síntese de um conflito entre *free* e *open*, que persiste. Nesse sentido, os liberal-comunistas, como descritos por Zizek (2006), são uma pista importante para o novo momento, em que as empresas *open* colocam-se como “fazedoras do bem”, “abertas”, “transparentes”, “horizontais”.

Neste texto, em especial a partir de dados de campo coletados durante o nono Fisl, espero ter também demonstrado a relação essencial entre ideologia, trabalho e poder no software livre. A ideologia em torno de cada uma das correntes políticas serve, embora não de modo exclusivo, como fator de atração para a colaboração e uso de determinados softwares. O grupo *open*, por exemplo, ao procurar distanciar-se das ideias políticas associadas especialmente a Richard Stallman, trouxe para dentro do movimento outros setores não-identificados com a perspectiva política até então propagada pela Free Software Foundation – empresas e desenvolvedores de software com ideias mais à direita. Ao fazê-lo, mudou o balanceamento de poder dentro do movimento, poder esse manifesto não somente pelo apoio público a determinadas visões, mas também pela colaboração arregimentada em determinados projetos de software.

Esse é um aspecto importante e diferencial do software livre com relação a outros movimentos. O software livre não é um movimento que apenas demanda políticas e/ou busca por uma nova ordem de relações sociais, ao contrário, ele busca, a partir de um sistema jurídico já constituído, usar a lógica

desse mesmo sistema para construir alternativas. A força do movimento não está em suas demandas, mas em sua capacidade de arregimentar trabalho voltado à produção-popularização de softwares que usam as regras convencionais, de forma inteligente, para garantir certas práticas. É um movimento que produz softwares e esses softwares se tornam produtos distribuídos no mercado de informática e que ocupam posições antes ocupadas, ou que poderiam ser ocupadas, por softwares proprietários. Por isso é especialmente relevante a análise das correntes políticas e ideológicas do software livre, pois elas se fortalecem em suas ligações produtivas com certos projetos de software assim como servem a esses projetos como fatores de atração de trabalho voluntário.

A entrada mais forte das corporações *open* impacta o software livre enquanto movimento, pois traz a ele militantes/colaboradores de perfil diferenciado, mais interessados no software livre pelas perspectivas profissionais que lhes são oferecidas. O perfil da base do movimento está em transformação, agora abarcando jovens interessados em carreiras mais tradicionais nas grandes empresas, carreiras semelhantes às dos executivos tradicionais das empresas transnacionais.

Neste trabalho, minhas preocupações são similares à de Terranova⁹⁸ (2000) quando busca trazer como tema fluxos de trabalho, cultura e poder. Embora o trabalho no software livre seja muitas vezes descrito como uma paixão ou como diversão – como na caracterização dos hackers feita por Himanem (2001) e Raymond (ver anexo) – é preciso reconhecer sua integração com o mercado das tecnologias da informação e investigar como cultura e poder estão presentes em suas estruturas – como em todas as relações humanas, como diria Wolf (1999) –, que não estão desconectadas do resto do mundo.

Terranova cita um dos gurus da literatura gerencial e de negócios, Don Tapscott, para confirmar que a

⁹⁸ Em seu ensaio intitulado “Free Labor: Producing Culture for the Digital Economy” Terranova afirma: “This essay does not seek to offer a judgment on the “effects” of the Internet, but rather to map the way in which the Internet connects to the autonomist “social factory.” I am concerned with how the “outernet” - the network of social, cultural, and economic relationships that criss-crosses and exceeds the Internet - surrounds and connects the latter to larger flows of labor, culture, and power. It is fundamental to move beyond the notion that cyberspace is about escaping reality in order to understand how the reality of the Internet is deeply connected to the development of late postindustrial societies as a whole.”

estrutura aberta é necessária para a atração de “ativos” para a empresa – ou seja, trabalhadores – e que essa empresa precisa ser convidativa a esses “ativos”, que a ela se associam. Diz ele: “A new challenge to management is first to attract and retain these assets by marketing the organization to them, and second to provide the creative and open communications environment where such workers can effectively apply and enhance their knowledge.”. Essa atração de “ativos”, porém, não vale apenas para os projetos corporativos. Vimos como, para Richard Stallman, falar GNU/Linux significa reconhecer os esforços da FSF o que, por sua vez, significa dar melhor condições a ela para atrair colaboração em seus projetos.

Terranova faz um debate interessante com Barbrook (2005) quando posiciona-se contrariamente à ideia de que a “gift-economy” funcionaria como um processo de superação do capitalismo por dentro. Os participantes dessa gift-economy usariam recursos públicos e privados para perseguirem uma “potlach economy” de trocas livres. Porém, segundo Terranova:

“Barbrook overemphasizes the autonomy of the high-tech gift economy from capitalism. The processes of exchange that characterize the Internet are not simply the reemergence of communism within the cutting edge of the economy, a repressed other that resurfaces just at the moment when communism seems defeated. It is important to remember that the gift economy, as part of a larger digital economy, is itself an important force within the reproduction of the labor force in late capitalism as a whole.”

Esse debate, contudo, não deve se encaminhar para uma demonização ou santificação da livre troca de bens simbólicos na Internet. É preciso reconhecer o quanto esse processo contém de contraditório, o quanto ao mesmo tempo funciona como viés de democratização e massificação do acesso e quanto é canalizado de forma a realimentar práticas capitalistas.

Rather than capital "incorporating" from the outside the authentic fruits of the collective imagination, it seems more reasonable to think of cultural flows as originating within a field that is always and already capitalism. Incorporation is not about capital descending on authentic culture but a more immanent process

of channeling collective labor (even as cultural labor) into monetary flows and its structuration within capitalist business practices. (Terranova, 2000)

Apontar para a ocorrência desses processos no software livre, cujo método de produção compartilhada progressivamente caminha para o centro da criação de software, não deve funcionar no sentido de “condená-lo” a um processo de degradação ou cooptação, mas sim no sentido de aprofundar o entendimento de sua dinâmica. Nesse esforço, procurei aqui dar ênfase aos aspectos ideológicos e culturais, aos modelos profissionais e de comportamento, às transformações operadas no nível do sentido que levam à atração de diferentes perfis de militantes e apoiadores.

Bibliografia

Apgaua, Renata. *A Dádiva Universal – reflexões em um debate ficcional*. Belo Horizonte, Dissertação de Mestrado em Sociologia, Faculdade de Filosofia e Ciências Humanas da Universidade Federal de Minas Gerais (UFMG), 1999.

_____. O Linux e a perspectiva da dádiva. In: *Horiz. Antropol.*, Porto Alegre, v. 10, n. 21, Junho 2004. Disponível em http://www.scielo.br/scielo.php?script=sci_arttext&pid=S0104-71832004000100010&lng=en&nrm=iso. Acesso em 20/01/2010.

Barbrook, Richard. The High-Tech Gift Economy. In: *First Monday*, volume 3, número 12. Disponível em http://firstmonday.org/issues/issue3_12/barbrook/, 1998; acesso em 28/10/2006.

_____. *Futuros Imaginários: das máquinas pensantes à aldeia global*. Ed. Peirópolis, São Paulo, 2009.

_____. e Cameron, Andy. *Californian Ideology*. 1995. Disponível em: http://www.alamut.com/subj/ideologies/pessimism/califIdeo_I.html; acesso em 28/10/2006.

Barlow, John P. The Economy of Ideas. In: *Wired*. Disponível em http://www.wired.com/wired/archive/2.03/economy.ideas_pr.html, 1994; acesso em 28/10/2006.

Benjamin, Walter. *A Obra de Arte na Era de sua Reprodutibilidade Técnica*. 1936. Disponível em <http://www.marxists.org/reference/subject/philosophy/works/ge/benjamin.htm> Acessado em 21/06/2008 .

Benkler, Yochai. *The Wealth of Networks: how social production transforms markets and freedom*. Yale University Press, 2006.

Antunes, Ricardo e Braga, Ruy. *Infoproletários: degradação real do trabalho virtual*. São Paulo: Boitempo, 2009.

Castells, Manoel. *A Sociedade em Rede. A Era da Informação v.1*. Tradução de Roneide Venâncio Majer. São Paulo: Paz e Terra, v.1, 2002.

Coleman, E. Gabriella. *The Politics of Survival and Prestige: Hacker Identity and the Global Production of an Operating System*. Dissertação de mestrado, University of Chicago 1999.

_____. The Political Agnositism of Free Software and the Politics of Contrast. In *Anthropology Quarterly*. Summer, 2004. Disponível em <http://www.healthhacker.org/biella/colemanaq.pdf>; acesso em 28/10/2004.

Dias, Eurípedes da Cunha. "Arqueologia dos Movimentos sociais". In: Gohn, Maria da Glória (org.) *Movimentos Sociais no Início do Século XXI: antigos e novos atores sociais*. Petrópolis, RJ, Editora Vozes, 2003.

- Evangelista, Rafael de Almeida. *Política e linguagem nos debates sobre o software livre*. Dissertação de Mestrado. IEL/Unicamp, 2005.
- Foucault, Michel. *Nascimento da Biopolítica*. São Paulo, Ed. Martins Fontes, 2008
- Feldman-Bianco, Bela (org. e introdução). *Antropologia das Sociedades Contemporâneas*. São Paulo, Global, 1987
- Feldman-Bianco, Bela e Ribeiro, Gustavo Lins. *Antropologia e Poder: Contribuições de Eric Wolf*. *Etnográfica*, Vol VII (2), 2003.
- Georgopoulou, Panayiota. The free/open source software movement: Resistance or change?. *Civitas*, Porto Alegre, v. 9, n. 1, p. 65-76, jan.-abr. 2009
- Gohn, Maria da Glória. *Teorias dos movimentos sociais. Paradigmas clássicos e contemporâneos*. São Paulo: Loyola, 1997.
- Goss, Karine Pereira e Prudencio, Kelly. O conceito de movimentos sociais revisitado. In *Tese: Revista Eletrônica dos Pós-Graduandos em Sociologia Política da UFSC*. Vol. 2, nº 1 (2), janeiro-julho 2004, p. 75-91
- Gorz, André. *O Imaterial: conhecimento, valor e capital*. São Paulo: Annablume, 2005.

Harvey, David. *Condição Pós-Moderna*. Edições Loyola, 4a edição. 1994.

_____. *O Neoliberalismo: história e implicações*. São Paulo: Loyola, 2008

Himanem, Pekka. *A Ética dos Hackers e o Espírito da Era da Informação*. Rio de Janeiro. Ed. Campus, 2001.

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

Kelty, Christopher. *Two Bits: The Cultural Significance of Free Software and the Internet* Durham: Duke University Press, 2008. Disponível em <http://twobits.net/>

Laclau, Ernesto. Os novos movimentos sociais e a pluralidade do social. *Revista Brasileira de Ciências Sociais*, n. 2, vol. 1, out. ,1986.

Lessig, Lawrence. *Code and Other Laws of Cyberspace*. New York: Basic Books. 1999.

Lojkine, Jean. O Novo Salariado Informacional. Nas Fronteiras do Salariado. *Crítica Marxista* n. 25. 2007.

López-Ruiz, Osvaldo. *O "Ethos" dos Executivos das Transnacionais e o Espírito do Capitalismo*. Tese de doutorado, Campinas, Universidade Estadual de Campinas, Instituto de Filosofia e Ciências Humanas, 2004.

- Mazières, Antoine Bernard Marie. *Uma análise do movimento do software livre e de código aberto*. Campinas, SP : [s. n.], 2009.
- Mauss, Marcel. *Ensaio Sobre a Dádiva: Forma e Razão da Troca nas Sociedades Arcaicas*. In: *Sociologia e antropologia*. São Paulo, EPU, 1974. v. 2, p. 37-184.
- Mendes, Cassia Isabel Costa. *Software Livre e Inovação Tecnológica: Uma Análise Sob a Perspectiva da Propriedade Intelectual*. Dissertação de mestrado. Universidade Estadual de Campinas . Instituto de Economia, 2006.
- Murillo, Luis Felipe Rosado. *Tecnologia, política e cultura na comunidade brasileira de software livre e de código aberto*. Dissertação de mestrado. Universidade Federal do Rio Grande do Sul. Instituto de Filosofia e Ciências Humanas. Programa de Pós-Graduação em Antropologia Social, 2009.
- Orlandi, Eni. *Interpretação. Autoria, leitura e efeitos do trabalho simbólico*. Ed. Vozes, 1996.
- Pêcheux, Michel. *Semântica e Discurso: uma crítica à afirmação do óbvio*. Campinas, 1997, E. da Unicamp.
- Ribeiro, Gustavo Lins. A Internet e a emergência da comunidade imaginada transnacional. *Revista Sociedade e Estado*, v. 10, n. 1, p. 181-191, jan./jun. 1995.

Raymond, Eric S. Homesteading the Noosphere. In: *First Monday*, volume 3, número 10
http://firstmonday.org/issues/issue3_10/raymond/ . 1998; acessado em 28/10/2006.

Raymond, Eric S. "Homesteading the Noosphere" *First Monday* [Online], Volume 3 Number 10 (5
October 1998).

_____. *A Catedral e o Bazar*. Versão traduzida para o português por Erik Kohler, 1997.
<http://www.geocities.com/CollegePark/Union/3590/pt-cathedral-bazaar.html>; acessado em
28/10/2006.

Rifkin, Jeremy. *A Era do Acesso*. Makrom Books, São Paulo.

Sahlins, Marshall D. Poor Man, Rich Man, Big-Man, Chief: Political Types in Melanesia and
Polynesia. *Comparative Studies in Society and History*, Vol 5, n 3. Abril, 1963.

Sanchez, Wilken. *O Movimento de Software Livre e a Produção Colaborativa do Conhecimento*.
Dissertação de Mestrado, PUC-SP, São Paulo, 2007.

Shaw, Aaron. *Mobilizing Insurgent Expertise: The Politics of Free and Open Source Software in
Brazil*. Dissertação de Mestrado. UC Berkeley, EUA, 2008.

Singer, Paul. Desenvolvimento capitalista e desenvolvimento solidário. In: *Estud. av.* , São Paulo, v.
18, n. 51, 2004. Disponível em: http://www.scielo.br/scielo.php?script=sci_arttext&pid=S0103-40142004000200001&lng=en&nrm=iso. Acesso em: 15/07/2008.

Söderberg, Johan. Copyleft vs. Copyright: A Marxist Critique. in *First Monday*. Fevereiro, 2002.

Disponível em http://www.firstmonday.org/issues/issue7_3/soderberg/; acesso em 28/10/2006

Stallman, Richard. *Free Software, Free Society*. Disponível em: <http://www.notabug.com/2002/rms-essays.pdf>; acesso em 28/10/2006.

_____. *The GNU Manifesto*. 1985. Disponível em <http://www.gnu.org/gnu/manifesto.pt-br.html>. Acessado em 19/01/2010.

Terranova, Tiziana. Free Labor: Producing Culture for the Digital Economy. *Social Text* - 63 (Volume 18, Number 2), Summer 2000, pp. 33-58

Touraine, Alain. *Can We Live Together? Equality and Difference*. Stanford, CA: Stanford Univ. Press, 2000

Touraine, Alain. From understanding society to discovering the subject. In *Anthropological Theory*, 2, 4: 387-398. 2002.

Virilio, Paul. *Velocidade e Política*. Estação Liberdade. São Paulo, 1996.

Weber, Max. *Ética Protestante e o Espírito do Capitalismo*. São Paulo, Ed. Pioneira, 1967.

Weber, Steve. *The Success of Open Source*. Cambridge, MA: Harvard University Press, 2004.

Williams, Sam. *Free as in Freedom: Richard Stallman's Crusade for Free Software*. EUA, março, 2002. Disponível em <http://www.oreilly.com/openbook/freedom/>; acessado em 28/10/2006.

Wolf, Eric. Culture: Panacea or Problem? In: *Pathways of Power: building an anthropology of modern world*. University of California Press, 2001.

_____. Facing Power – Old Insights, New Questions. In: *American Anthropologist*, New Series, Vol 92, n3. Setembro, 1990.

_____. *Envisioning Power*. University of California Press, 1999.

Žižek, S. "Nobody Needs to Be Vile". *London Review of Books* Vol. 28, No7 London, 2006.

Anexos

Como se Tornar um Hacker⁹⁹

Por que esse documento?

Como editor do [Jargon File](#), frequentemente recebo pedidos por email de entusiasmados iniciantes, perguntando (de fato) "como eu posso aprender a ser um grande hacker?". Estranhamente, parece que não existem FAQs ou documentos na Web que se refiram a essa importante questão, então aqui está o meu.

Caso você esteja lendo um trecho deste documento off-line, a versão atual fica em <http://www.ccil.org/~esr/faqs/hacker-howto.html>.

O que é um hacker?

O [Jargon File](#) contém um monte de definições do termo "hacker", a maioria deles tendo a ver com aptidão técnica e um prazer em resolver problemas e superar limites. Se você quer saber como se *tornar* um hacker, entretanto, apenas duas são realmente relevantes.

Existe uma comunidade, uma cultura compartilhada, de programadores experts e gurus de rede cuja história remonta a décadas atrás, desde os primeiros minicomputadores de tempo compartilhado e os primeiros experimentos na ARPAnet. Os membros dessa cultura deram origem ao termo "hacker". Hackers construíram a Internet. Hackers fizeram do sistema operacional Unix o que ele é hoje. Hackers mantêm a Usenet. Hackers fazem a World Wide Web funcionar. Se você é parte desta cultura, se você contribuiu a ela e outras pessoas o chamam de hacker, você é um hacker.

A mentalidade hacker não é confinada a esta cultura do hacker-de-software. Há pessoas que aplicam a atitude hacker em outras coisas, como eletrônica ou música – na verdade, você pode encontrá-la nos níveis mais altos de qualquer ciência ou arte. Hackers de software reconhecem esses espíritos aparentados de outros lugares e podem chamá-los de "hackers" também – e alguns alegam que a natureza hacker é realmente independente da mídia particular em que o hacker trabalha. Mas no restante deste documento, nos concentraremos nas habilidades e dos hackers de software, e nas tradições da cultura compartilhada que deu origem ao termo 'hacker'.

Existe outro grupo de pessoas que se dizem hackers, mas não o são. São pessoas (adolescentes do sexo

⁹⁹ Texto escrito originalmente por Eric Raymond e atualizado por diversas vezes. Esta é uma versão de 1998, traduzida e publicada em português por Rafael Caetano dos Santos. Disponível em <http://www.linux.ime.usp.br/~rcaetano/docs/hacker-howto-pt.html>, acessada em 15/12/2009.

masculino, na maioria) que se divertem invadindo computadores e fraudando o sistema telefônico. Hackers de verdade chamam essas pessoas de "crackers", e não tem nada a ver com eles. Hackers de verdade consideram os crackers preguiçosos, irresponsáveis, e não muito espertos, e alegam que ser capaz de quebrar sistemas de segurança torna alguém hacker tanto quanto fazer ligação direta em carros torna alguém um engenheiro automobilístico. Infelizmente, muitos jornalistas e escritores foram levados a usar, erroneamente, a palavra "hacker" para descrever crackers; isso é muito irritante para os hackers de verdade.

A diferença básica é esta: hackers constróem coisas, crackes as destróem.

Se você quer ser um hacker, continue lendo. Se você quer ser um cracker, vá ler o newsgroup [alt.2600](#) e se prepare para se dar mal depois de descobrir que você não é tão esperto quanto pensa. E isso é tudo que eu digo sobre crackers.

A Atitude Hacker

Hackers resolvem problemas e constróem coisas, e acreditam na liberdade e na ajuda mútua voluntária. Para ser aceito como um hacker, você tem que se comportar de acordo com essa atitude. E para se comportar de acordo com essa atitude, você tem que realmente acreditar nessa atitude.

Mas se você acha que cultivar a atitude hacker é somente um meio para ganhar aceitação na cultura, está enganado. Tornar-se o tipo de pessoa que acredita nessas coisas é importante para *você* -- para ajudá-lo a aprender e manter-se motivado. Assim como em todas as artes criativas, o modo mais efetivo para se tornar um mestre é imitar a mentalidade dos mestres -- não só intelectualmente como emocionalmente também.

Então, se você quer ser um hacker, repita as seguinte coisas até que você acredite nelas:

1. O mundo está repleto de problemas fascinantes esperando para serem resolvidos.

Ser hacker é muito divertido, mas é um tipo de diversão que necessita de muito esforço. Para haver esforço é necessário motivação. Atletas de sucesso retiram sua motivação de uma espécie de prazer físico em trabalhar seus corpos, em tentar ultrapassar seus próprios limites físicos. Analogamente, para ser um hacker você precisa ter uma emoção básica em resolver problemas, afiar suas habilidades e exercitar sua inteligência. Se você não é o tipo de pessoa que se sente assim naturalmente, você precisará se tornar uma para ser um hacker. Senão, você verá sua energia para "hackear" sendo esvaída por distrações como sexo, dinheiro e aprovação social.

(Você também tem que desenvolver uma espécie de fé na sua própria capacidade de aprendizado -- crer

que, mesmo que você não saiba tudo o que precisa para resolver um problema, se souber uma parte e aprender a partir disso, conseguirá aprender o suficiente para resolver a próxima parte -- e assim por diante, até que você termine.)

2. Não se deve resolver o mesmo problema duas vezes.

Mentes criativas são um recurso valioso e limitado. Não devem ser desperdiçadas reinventando a roda quando há tantos problemas novos e fascinantes por aí.

Para se comportar como um hacker, você tem que acreditar que o tempo de pensamento dos outros hackers é precioso -- tanto que é quase um dever moral compartilhar informação, resolver problemas e depois dar as soluções, para que outros hackers possam resolver *novos* problemas ao invés de ter que se preocupar com os antigos indefinidamente. (Você não tem que acreditar que é obrigado a dar *toda* a sua produção criativa, ainda que hackers que o fazem sejam os mais respeitados pelos outros hackers. Não é inconsistente com os valores do hacker vender o suficiente da sua produção para mantê-lo alimentado e pagar o aluguel e computadores. Não é inconsistente usar suas habilidades de hacker para sustentar a família ou mesmo ficar rico, contanto que você não esqueça que é um hacker.)

3. Tédio e trabalho repetitivo são nocivos.

Hackers (e pessoas criativas em geral) não podem ficar entediadas ou ter que fazer trabalho repetitivo, porque quando isso acontece significa que eles não estão fazendo o que apenas eles podem fazer -- resolver novos problemas. Esse desperdício prejudica a todos. Portanto, tédio e trabalho repetitivo não são apenas desagradáveis, mas nocivos também.

Para se comportar como um hacker, você tem que acreditar nisso de modo a automatizar as partes chatas tanto quanto possível, não apenas para você como para as outras pessoas (principalmente outros hackers).

(Há uma exceção aparente a isso. Às vezes, hackers fazem coisas que podem parecer repetitivas ou tediosas para um observador, como um exercício de "limpeza mental", ou para adquirir uma habilidade ou ter uma espécie particular de experiência que não seria possível de outro modo. Mas isso é por opção -- ninguém que consiga pensar deve ser forçado ao tédio.

4. Liberdade é uma coisa boa.

Hackers são naturalmente anti-autoritários. Qualquer pessoa que lhe dê ordens pode impedi-lo de resolver qualquer que seja o problema pelo qual você está fascinado -- e, dado o modo em que a mente autoritária funciona, geralmente arranjará alguma desculpa espantosamente idiota para isso. Então, a

atitude autoritária deve ser combatida onde quer que você a encontre, para que não sufoque a você e a outros hackers.

(Isso não é a mesma coisa que combater toda e qualquer autoridade. Crianças precisam ser orientadas, e criminosos, detidos. Um hacker pode aceitar alguns tipos de autoridade a fim de obter algo que ele quer mais que o tempo que ele gasta seguindo ordens. Mas isso é uma barganha restrita e consciente; não é o tipo de sujeição pessoal que os autoritários querem.)

Pessoas autoritárias prosperam na censura e no segredo. E desconfiam de cooperação voluntária e compartilhamento de informação -- só gostam de "cooperação" que eles possam controlar. Então, para se comportar como um hacker, você tem que desenvolver uma hostilidade instintiva à censura, ao segredo, e ao uso da força ou mentira para compelir adultos responsáveis. E você tem que estar disposto a agir de acordo com esta crença.

5. Atitude não substitui competência.

Para ser um hacker, você tem que desenvolver algumas dessas atitudes. Mas apenas ter uma atitude não fará de você um hacker, assim como não o fará um atleta campeão ou uma estrela de rock. Para se tornar um hacker é necessário inteligência, prática, dedicação, e trabalho duro.

Portanto, você tem que aprender a desconfiar de atitude e respeitar todo tipo de competência. Hackers não deixam posers gastar seu tempo, mas eles idolatram competência – especialmente competência em "hackear", mas competência em qualquer coisa é boa. A competência em habilidades que poucos conseguem dominar é especialmente boa, e competência em habilidades que envolvem agudeza mental, perícia e concentração é a melhor.

Se você reverenciar competência, gostará de desenvolvê-la em si mesmo – o trabalho duro e dedicação se tornará uma espécie de um intenso jogo, ao invés de trabalho repetitivo. E isso é vital para se tornar um hacker.

Habilidades básicas do hacker

A atitude hacker é vital, mas habilidades são ainda mais vitais. Atitude não substitui competência, e há uma certo conjunto de habilidades que você precisa ter antes que um hacker sonhe em lhe chamar de um.

Esse conjunto muda lentamente com o tempo, de acordo com a criação de novas habilidades. Por exemplo, costumava incluir programação em linguagem de máquina, e até recentemente não incluía HTML. Mas agora é certo que inclui o seguinte:

1. Aprenda a programar.

Essa é, claro, a habilidade básica do hacker. Em 1997, a linguagem que você absolutamente precisa aprender é C (apesar de não ser a que você deve aprender primeiro). Mas você não é um hacker e nem mesmo um programador se você souber apenas uma linguagem – você tem que aprender a pensar sobre problemas de programação de um modo geral, independentemente de qualquer linguagem. Para ser um hacker de verdade, você precisa ter chegado ao ponto de conseguir aprender uma nova linguagem em questão de dias, relacionando o que está no manual ao que você já sabe. Isso significa que você deve aprender várias linguagens bem diferentes.

Além de C, você também deve aprender pelo menos LISP e Perl (e Java está tentando pegar um lugar nessa lista). Além de serem as linguagens mais importantes para hackear, cada uma delas representa abordagens à programação bem diferentes, e todas o educarão em pontos importantes.

Eu não posso lhe dar instruções completas sobre como aprender a programar aqui – é uma habilidade complexa. Mas eu posso lhe dizer que livros e cursos também não servirão (muitos, talvez *a maioria* dos melhores hackers são auto-didatas). O que servirá é (a) *ler código* e (b) *escrever código*.

Aprender a programar é como aprender a escrever bem em linguagem natural. A melhor maneira é ler um pouco dos mestres da forma, escrever algumas coisas, ler mais um monte, escrever mais um monte, ler mais um monte, escrever... e repetir até que seu estilo comece a desenvolver o tipo de força e economia que você vê em seus modelos.

Achar bom código para ler costumava ser difícil, porque havia poucos programas grandes disponíveis em código-fonte para que hackers novatos pudessem ler e mexer. Essa situação mudou dramaticamente; open-source software (software com código-fonte aberto), ferramentas de programação, e sistemas operacionais (todos feitos por hackers) estão amplamente disponíveis atualmente.

2. Pegue um dos Unixes livres e aprenda a mexer.

Estou assumindo que você tem um computador pessoal ou tem acesso a um (essas crianças de hoje em dia tem tão facilmente :-)). O passo mais importante que um novato deve dar para adquirir habilidades de hacker é pegar uma cópia do Linux ou de um dos BSD-Unixes, o instalar em um PC, e rodá-lo.

Sim, há outros sistemas operacionais no mundo além do Unix. Porém, eles são distribuídos em forma binária – você não consegue ler o código, e você não consegue modificá-lo. Tentar aprender a "hackear" em DOS, Windows ou MacOS é como tentar aprender a dançar com o corpo engessado.

Além disso, Unix é o sistema operacional da Internet. Embora você possa aprender a usar a Internet sem conhecer Unix, você não pode ser um hacker sem entendê-lo. Por isso, a cultura hacker, atualmente, é fortemente centralizada no Unix. (Não foi sempre assim, e alguns hackers da velha guarda não gostam da situação atual, mas a simbiose entre o Unix e a Internet se tornou tão forte que até mesmo o músculo da Microsoft não parece ser capaz de ameaçá-la seriamente.)

Então, pegue um Unix – eu gosto do Linux, mas existem outros caminhos. Aprenda. Rode. Mexa. Acesse a Internet através dele. Leia o código. Modifique o código. Você terá ferramentas de programação (incluindo C, Lisp e Perl) melhores do qualquer sistema operacional da Microsoft pode sonhar em ter, você se divertirá, e irá absorver mais conhecimento do que perceber, até que você olhará para trás como um mestre hacker.

Para aprender mais sobre Unix, veja [The Loginataka](#).

Para pegar o Linux, veja [Where To Get Linux](#).

3. Aprenda a usar a World Wide Web e escrever em HTML.

A maioria das coisas que a cultura hacker tem construído funciona "invisivelmente", ajudando no funcionamento de fábricas, escritórios e universidades sem nenhum óbvio na vida dos não-hackers. A Web é a grande exceção, o enorme e brilhante brinquedo dos hackers que até mesmo *políticos* admitem que está mudando o mundo. Por esse motivo (e vários outros também) você precisa aprender como trabalhar na Web.

Isso não significa apenas aprender a mexer em um browser (qualquer um faz isso), mas aprender a programar em HTML, a linguagem de markup da Web. Se você não sabe programar, escrever em HTML lhe ensinará alguns hábitos mentais que o ajudarão. Então faça uma home page.

Mas apenas ter uma home page não chega nem perto de torná-lo um hacker. A Web está repleta de home pages. A maioria delas é inútil, porcária sem conteúdo – porcária muito bem apresentada, note bem, mas porcária mesmo assim (mais sobre esse assunto em [The HTML Hell Page](#)).

Para valer a pena, sua página deve ter *conteúdo* – deve ser interessante e/ou útil para outros hackers. E isso nos leva ao próximo assunto...

Status na Cultura Hacker

Como a maioria das culturas sem economia monetária, a do hacker se baseia em reputação. Você está tentando resolver problemas interessantes, mas quão interessantes eles são, e se suas soluções são

realmente boas, é algo que somente seus iguais ou superiores tecnicamente são normalmente capazes de julgar.

consequentemente, quando você joga o jogo do hacker, você aprende a marcar pontos principalmente pelo que outros hackers pensam da sua habilidade (por isso você não é hacker até que outros hackers lhe chamem assim). Esse fato é obscurecido pela imagem solitária que se faz do trabalho do hacker; e também por um tabu hacker-cultural que é contra admitir que o ego ou a aprovação externa estão envolvidas na motivação de alguém.

Especificamente, a cultura hacker é o que os antropologistas chamam de *cultura de doação*. Você ganha status e reputação não por dominar outras pessoas, nem por ser bonito, nem por ter coisas que as pessoas querem, mas sim por doar coisas. Especificamente, por doar seu tempo, sua criatividade, e os resultados de sua habilidade.

Há basicamente cinco tipos de coisas que você pode fazer para ser respeitado por hackers:

1. Escrever open-source software.

O primeiro (o mais central e mais tradicional) é escrever programas que outros hackers achem divertidos ou úteis, e dar o código-fonte para que toda a cultura hacker use.

(Nós costumávamos chamar isto de "free software", mas isso confundia muitas pessoas que não sabiam ao certo o significado de "free". Agora, muitos de nós preferem o termo "[open-source](#)" software).

[nota do tradutor: "free" significa tanto "livre" como "gratuito", daí a confusão. O significado que se pretende é "livre".] Os "semi-deuses" mais venerados da cultura hacker são pessoas que escreveram programas grandes, competentes, que encontraram uma grande demanda e os distribuíram para que todos pudessem usar.

2. Ajude a testar e depurar open-source software

Também estão servindo os que depuram open-source software. Neste mundo imperfeito, inevitavelmente passamos a maior parte do tempo de desenvolvimento na fase de depuração. Por isso, qualquer autor de open-source software que pense lhe dirá que bons beta-testers (que saibam descrever sintomas claramente, localizar problemas, tolerar bugs em um lançamento apressado, e estejam dispostos a aplicar algumas rotinas de diagnóstico) valem seu peso em ouro. Até mesmo um desses beta-testers pode fazer a diferença entre uma fase de depuração virar um longo e cansativo pesadelo, ou ser apenas um aborrecimento saudável. Se você é um novato, tente achar um programa sob desenvolvimento em que você esteja interessado e seja um bom beta-tester. Há um progresso natural

de ajudar a testar programas para ajudar a depurar e depois ajudar a modificá-los. Você aprenderá muito assim, e criará um bom karma com pessoas que lhe ajudarão depois.

3. Publique informação útil.

Outra boa coisa a se fazer é coletar e filtrar informações úteis e interessantes em páginas da Web ou documentos como FAQs ("Frequently Asked Questions lists", ou listas de perguntas frequentes), e torne-os disponíveis ao público.

Mantenedores de grandes FAQs técnicos são quase tão respeitados quanto autores de open-source software.

4. Ajude a manter a infra-estrutura funcionando.

A cultura hacker (e o desenvolvimento da Internet, quanto a isso) é mantida por voluntários. Existe muito trabalho sem glamour que precisa ser feito para mantê-la viva – administrar listas de email, moderar grupos de discussão, manter grandes sites que armazenam software, desenvolver RFCs e outros padrões técnicos.

Pessoas que fazem bem esse tipo de coisa são muito respeitadas, porque todo mundo sabe que esses serviços tomam muito tempo e não são tão divertidos como mexer em código. Fazê-los mostra dedicação.

5. Sirva a cultura hacker em si.

Finalmente, você pode servir e propagar a cultura em si (por exemplo, escrevendo um apurado manual sobre como se tornar um hacker :-)). Você só terá condição de fazer isso depois de ter estado por aí por um certo tempo, e ter se tornado conhecido por uma das primeiras quatro coisas.

A cultura hacker não têm líderes, mas têm seus heróis culturais, "chefes tribais", historiadores e porta-vozes. Depois de ter passado tempo suficiente nas trincheiras, você pode se tornar um desses.

Cuidado: hackers desconfiam de egos espalhafatosos em seus "chefes tribais", então procurar visivelmente por esse tipo de fama é perigoso. Ao invés de se esforçar pela fama, você tem que de certo modo se posicionar de modo que ela "caia" em você, e então ser modesto e cortês sobre seu status.

A Conexão Hacker/Nerd

Contrariamente ao mito popular, você não tem que ser um nerd para ser um hacker. Ajuda, entretanto, e muitos hackers são de fato nerds. Ser um proscrito social o ajuda a se manter concentrado nas coisas realmente importantes, como pensar e "hackear".

Por isso, muitos hackers adotaram o rótulo "nerd", e até mesmo usam o termo (mais duro) "geek" como um símbolo de orgulho – é um modo de declarar sua independência de expectativas sociais normais.

Veja [The Geek Page](#) para discussão extensiva.

Se você consegue se concentrar o suficiente em hackear para ser bom nisso, e ainda ter uma vida, está ótimo. Isso é bem mais fácil hoje do que quando era um novato nos anos 70; atualmente a cultura mainstream é muito mais receptiva a tecno-nerds. Há até mesmo um número crescente de pessoas que percebem que hackers são, frequentemente, amantes e cônjuges de alta qualidade. [Girl's Guide to Geek Guys](#).

Se hackear o atrai porque você não vive, tudo bem – pelo menos você não terá problemas para se concentrar. Talvez você consiga uma vida normal depois.

Pontos Sobre Estilo

Para ser um hacker, você tem que entrar na mentalidade hacker. Há algumas coisas que você pode fazer quando não estiver na frente de um computador e que podem ajudar. Não substituem o ato de hackear (nada substitui isso), mas muitos hackers as fazem, e sentem que elas estão ligadas de uma maneira básica com a essência do hacking.

- Leia ficção científica. frequente convenções de ficção científica (uma boa maneira de encontrar hackers e proto-hackers).
- Stude o Zen, e/ou faça artes marciais. (A disciplina mental parece similar em pontos importantes).
- Desenvolva um ouvido analítico para música. Aprenda a apreciar tipos peculiares de música. Aprenda a tocar bem algum instrumento musical, ou a cantar.
- Desenvolva sua apreciação de trocadilhos e jogo de palavras.
- Aprenda a escrever bem em sua língua nativa. (Um número surpreendente de hackers, incluindo todos os melhores que eu conheço, são bons escritores.)

Quanto mais dessas coisas você já fizer, mais provável que você tenha naturalmente um material hacker. Por que essas coisas em particular não é completamente claro, mas elas são ligadas com uma mistura de habilidades dos lados esquerdo e direito do cérebro que parece ser muito importante (hackers precisam ser capazes de tanto raciocinar logicamente quanto pôr de lado, de uma hora para outra, a lógica aparente do problema).

Finalmente, algumas coisas a *não* serem feitas.

- Não use um nome de usuário ou pseudônimo bobo e grandioso.
- Não entre em flame wars ("guerrinhas") na Usenet (ou em qualquer outro lugar).
- Não se auto-intitule um "cyberpunk", e não perca seu tempo com alguém que o faça.
- Não poste ou escreva email cheio de erros de ortografia e gramática.

A única reputação que você conseguirá fazendo alguma dessas coisas é a de um twit [um chato, geralmente filtrado nos grupos de discussão]. Hackers tem boa memória – pode levar anos antes que você se reabilite o suficiente para ser aceito.

Outros Recursos

O [Loginataka](#) tem algumas coisas a dizer sobre o treinamento e a atitude adequados a um hacker de Unix.

Eu também escrevi [A Brief History Of Hackerdom](#).

Peter Seebach mantém um excelente [Hacker FAQ](#) para gerentes que não sabem como lidar com hackers.

Eu escrevi um documento, [The Cathedral and the Bazaar](#) ("A Catedral e o Bazar"), que explica muito sobre como o Linux e as culturas de open-source software funcionam.

Perguntas frequentes

Q: Você me ensina como "hackear"?

Desde que publiquei essa página, recebi vários pedidos por semana de pessoas querendo que eu "ensinasse tudo sobre hacking". Infelizmente, eu não tenho tempo nem energia para isso; meus próprios projetos hackers tomam 110% do meu tempo.

Mesmo se eu fizesse, hacking é uma atitude e uma habilidade na qual você tem que basicamente ser auto-didata. Você verá que, embora hackers de verdade queiram lhe ajudar, eles não o respeitarão se você pedir "mastigado" tudo que eles sabem.

Aprenda algumas coisas primeiro. Mostre que você está tentando, que você é capaz de aprender sozinho. Depois faça perguntas aos hackers que encontrar.

Q: Onde eu posso encontrar hackers de verdade para conversar?

Bem, não no IRC, com certeza -- lá só existem flammers e crackers. A melhor maneira é encontrar um grupo de usuários local de Unix ou Linux, e frequentar as reuniões (você pode encontrar links para várias listas de grupos de usuários na página da [LDP](#) em Sunsite).

Q: Que linguagem devo aprender primeiro?

HTML, se você ainda não souber. Existe um monte de livros sobre HTML lustrosos, modistas e *ruins* por aí e, infelizmente, pouquíssimos bons. O livro de que mais gosto é [HTML: The Definitive Guide](#).

Quando você estiver pronto pra começar a programar, eu recomendaria começar com [Perl](#) ou [Python](#). C é realmente importante, mas muito mais difícil.

Q: Mas o open-source software não deixará os programadores incapazes de ganhar a vida?

Parece improvável – até agora, a indústria de open-source software parece estar criando empregos ao invés de tirá-los. Se ter escrito um programa é ganho econômico em relação a não tê-lo escrito, um programador será pago independentemente de o programa ser livre depois de feito. E, independentemente de quanto open-source software é feito, sempre parece haver mais demanda por aplicações novas e personalizadas.

Q: Como eu começo? Onde posso pegar um Unix livre?

Em outro lugar da página eu incluí ponteiros onde pegar o Linux. Para ser um hacker você precisa de motivação, iniciativa e capacidade de se educar. Comece agora...

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, precisamos 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 distribui-las como obras separadas. Todavia, quando você distribui-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. consequentemente, 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 consequência, 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 consequência 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 ideia 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.

Sulamita Garcia - Cabelinho ensebado não dá!

Quando a palestra se chama "Profissionalismo para nerds" você já sabe que vai dar pano pra manga, certo? E dá mesmo.

Assisti à tal palestra durante o Fisl 9.0, na PUC-RS, e no final conversei com a palestrante, Sulamita Garcia, a gerente de Estratégia Linux e Open Source da Intel para a América Latina. Muitas dicas, muitos conselhos e muita risada, este foi o resultado do papo.

E agora, na Entrevista da Semana, você confere este compilado de informações que ensinam literalmente a ser um bom profissional no ramo da programação. Traduzindo, um bom nerd – sem ofensa!

Gláucia Civa

O profissional de software geralmente entra muito cedo na área. Como embasar este início para construir uma carreira promissora?

Sulamita Garcia: Primeiramente é preciso acabar com o deslumbramento e encarar as coisas como são. Ser um programador em uma empresa, por exemplo, não significa só entrar e programar, mas também se adequar às normas da companhia. Se você fica deprimido só em pensar na ideia de ter de usar terno e gravata todos os dias, é bom refletir, pois muitas organizações da TI exigem isso.

Outra coisa é saber trabalhar na atual Gestão por Resultados, um dos modelos mais adotados pelas empresas. Ou seja: não adianta só produzir, trabalhar, fazer – o seu trabalho tem que render frutos, ou então está fora.

Também é preciso ver as opções da carreira, o que é um aspecto animador: antigamente, um profissional da programação ou virava programador pela vida inteira, ou se tornava, com muito custo, chefe de programação. Hoje, um profissional desta área tem mais possibilidades, como atuar na gerência de projetos, como um mentor técnico em algum cargo ou até mesmo evoluir para o posto de CIO.

E como chegar lá?

Sulamita Garcia: Estude! E não estou falando só de formação. A graduação é fundamental, mas não é suficiente. Leia muito, muitos livros técnicos, não fique o dia todo no ICR falando bobagem com a desculpa de que está trocando conhecimentos! Invista também em pós-graduação, e isso pode ser feito no exterior, garantindo um upgrade no currículo. Para isso, ofertas de bolsas de estudo não faltam, basta ficar atento e se preparar para as oportunidades. Agora, MBA, só se o seu interesse não for crescer na área técnica, mas sim na gerencial.

Também se prepare via freelancers, se estiver sem um emprego fixo. Você pode até mesmo optar por desenvolver sua profissão nesta linha. Existem muitos sites na web voltados a este nicho: as empresas/pessoas enviam trabalhos, o programador recebe, desenvolve, se relaciona com o cliente e entrega tudo sem precisar de contato presencial. Tudo muito prático e construtivo!

E quanto às certificações?

Sulamita Garcia: São muito boas de se ter, mas cuidado: cursos para certificações não se baseiam no mercado, mas na prova para os títulos, e a prova, além de ser somente sobre o curso, é muitas vezes comprável pela web, todo mundo sabe disso. Fique atento, portanto.

Outros problemas: os departamentos de Recursos Humanos das empresas de TI não são a TI, e portanto nem sempre entendem ou reconhecem as rotinas técnicas como esperam os profissionais desta área. Para os RHs, certificações valem menos do que um diploma, é bom saber.

Além do mais, foi-se o tempo em que as empresas bancavam com gosto os cursos e certificações de seus profissionais. É claro que isso ainda existe, e são vários os casos, mas muitas companhias reduziram custos e passaram a bancar não mais o curso, mas sim a prova, somente, e mais ainda: apenas se você passar.

Então os departamentos pessoais não possuem ainda o entendimento e entrosamento necessário para ajudar no crescimento do pessoal da programação?

Sulamita Garcia: Não necessariamente. Há pontos a serem resolvidos, como os que acabei de comentar, mas também há mudanças para o bem. Hoje em dia, por exemplo, você já pode faltar ao trabalho por um dia inteiro porque foi a uma conferência. Antes, isso era extremamente estranho, era preciso dizer que foi ao médico ou coisa assim. Ou seja: você podia faltar por causa de uma gripe, mas não para se instruir. Isso mudou.

O tempo livre dos profissionais da programação nas empresas, quando ocorre, também já é visto de forma diferente, quando bem utilizado. Por exemplo: se você utilizar este tempo para enviar colaborações pela web, por meio de comunidades de software livre, por exemplo, estará divulgando seu trabalho e o da empresa. Além disso, estará contribuindo diretamente para a expansão do próprio setor de software, o que poderá reverter em benefício para a própria companhia onde está trabalhando, já que as colaborações geram sistemas melhores, aplicações facilitadas, etc.

Outra coisa: colaborar pelas comunidades é dar visão ao seu trabalho não só para o exterior, mas também para o interior da empresa. Se hoje você está em um cargo baixo, mostrar seu trabalho pode ser uma forma de chamar a atenção para seus esforços, rendendo, quem sabe, uma promoção.

Que outros artifícios podem ser usados para “chamar a atenção” da empresa?

Sulamita Garcia: Tem um ótimo, mas odiado: o relatório! Todo programador odeia ter que fazer relatórios, mas eles podem ser usados a seu favor. Veja bem: se a cada dois, três dias você fizer um relatório de tudo o que fez, seus trabalhos, procedimentos utilizados, ações concluídas, e enviar ao seu superior, estará demonstrando claramente seu serviço. Um serviço que poderia não ser visto, passar despercebido, pense nisso!

E agora... Como é essa história de ter de mandar o pessoal do software pro banho?

Sulamita Garcia: Ah... A higiene é um problema sério em muitas empresas. Você pode não acreditar, mas o profissional de software, muitas vezes – não em todas, sejamos bem claros! (risos) – entra tanto em seu trabalho que se esquece de cuidar de si. Assim, usa a mesma camiseta do Google até que esteja completamente esburacada, desbotada, velha. E tem pior: tem conferências empresariais onde, por incrível que pareça, a gente tem de dizer “olha pessoal, tem que tomar banho todos os dias, escovar os dentes, se vestir direito. Cabelinho ensebado, camiseta e bermuda não dá!”.