

Software Livre e o Modelo Colaborativo de Produção entre Pares: uma análise organizacional sobre o Projeto GNOME.

Autoria: Vicente Macedo de Aguiar

Resumo

Este artigo discute as especificidades do modelo colaborativo de produção entre pares presente nas comunidades *on-line* de softwares livres - em particular, na comunidade relativa ao Projeto *GNU Network Object Model Environment*, mais conhecido pela sigla GNOME. Com base em uma pesquisa netnográfica de dois anos, o artigo analisa a organização do trabalho que dá vida ao processo de desenvolvimento colaborativo entre pares desse projeto de software livre, empreendido por mais de 300 hackers e colaboradores de todos os cinco continentes do globo por meio de uma comunidade *on-line*. Para tanto, leva-se em consideração algumas dimensões de uma realidade organizacional complexa como, por exemplo, a estrutura social, o fluxo de atividades, além dos impactos do ambiente tecnológico que compõem o processo de produção de um sistema computacional livre num projeto de magnitude internacional. A partir da análise desenvolvida neste artigo, conclui-se que este fenômeno organizacional, que emerge juntamente com as novas tecnologias da informação, parece desafiar boa parte dos conceitos e práticas que haviam sido ditados pela ciência da administração e pelo paradigma dominante da produção de softwares.

Introdução

A sociedade contemporânea é caracterizada por muitos cientistas sociais – a exemplo de Castells (2003), Lévy (1999) e Benkler (2006) - como a sociedade em rede, onde as tecnologias da informação, como a Internet, estão se tornando numa espécie de “tecido digital” do cotidiano da vida humana. Assim, cada vez mais, a rede mundial de computadores se afirma como a base tecnológica para a constituição de agrupamentos e organizações sociais que acabam por impactar diretamente na dinâmica econômica, política e cultural do mundo contemporâneo. Este contexto atual de relações entre tecnologia e produção social acaba então se tornando um campo fértil para estudos e pesquisas na área das ciências humanas. Muitos são as pesquisas e produções acadêmicas que se debruçam para tentar compreender essa realidade de intensas transformações que anuncia até a superação da era industrial.

No entanto, do ponto de vista da análise organizacional, poucos ainda são os estudos que procuram analisar as características, a cultura e os impactos de organizações situadas nesse novo contexto digital de relações mediadas por computador. Grande parte dos fenômenos organizacionais, por exemplo, que tem como base a Internet se encontram, assim, fora de discussões e pesquisas acadêmicas. Pouco se sabe sobre a dinâmica organizacional de comunidades situadas no ciberespaço que muitas vezes são responsáveis por definir os rumos e padrões tecnológicos da própria sociedade em rede. Além disso, a ética, o trabalho, a cultura e a lógica econômica dos sujeitos que convivem nesse ambiente digital acabam da mesma forma sendo marginalizados e pouco compreendidos.

Sendo assim, de forma contrária ao ato de fechar os olhos para esta realidade (social) singular que se apresenta, “que tentemos entendê-la, pois a verdadeira questão não é ser contra ou a favor, mas sim reconhecer as mudanças qualitativas na ecologia dos signos, no ambiente inédito que resulta da extensão das novas redes de comunicação para a vida social e cultural.” (Lévy, 1999, p.12). Partindo desta perspectiva, este artigo se sedimenta sobre um estudoⁱ científico que buscou analisar as especificidades da organização do processo de produção colaborativo, presente nas comunidades virtuaisⁱⁱ de desenvolvimento de softwares livres - em particular, na comunidade relativa ao Projeto *GNU Network Object Model Environment*, mais conhecido e difundido pela sigla GNOME.

Envolvendo mais de 300 hackers de todos os cinco continentes do globo, o Projeto GNOME gira em torno de dois objetivos comuns. De um lado, tendo como foco o chamado “usuário final”, o projeto visa o desenvolvimento de um ambiente de *desktop*ⁱⁱⁱ com uma interface gráfica (*Graphical User Interface* – GUI) e um conjunto de aplicativos (softwares) que facilitem ao máximo a interação do usuário com o computador (hardware). Além do *desktop*, o projeto acaba tendo ainda como segundo objetivo a estruturação de uma plataforma^{iv} de desenvolvimento, contendo uma coleção de Interfaces de Programação de Aplicativos (APIs), além de diversas ferramentas de programação, que são utilizadas por hackers tanto do projeto, como também de outras comunidades de softwares livres espalhadas pelo ciberespaço.

Assim, como ponto de partida, este artigo analisa como se manifesta a organização e dinâmica do trabalho que impulsiona o processo de desenvolvimento de todo o conjunto de sistemas computacionais livres que compõem o Projeto GNOME e o impacto desse processo tecnológico no cotidiano da comunidade *on-line* que dá vida para este projeto internacional.

Todavia, antes de adentrar nesses pontos, faz-se necessário realizar neste mesmo artigo um esclarecimento sobre a ambigüidade e o mal-entendimento que se faz sobre o termo e a práxis social dos *hackers*. Afinal, na sua essência, o termo *hacker* não está associado à indivíduos irresponsáveis que visam penetrar em sistemas computacionais de forma ilícita – como é normalmente propagado pela mídia de massa tradicional. Esse tipo de indivíduos que violam sistemas de segurança e quebram códigos computacionais são, especificamente, denominados de *crackers* e em geral são repudiados pelos membros das comunidades internacionais de software livre, como a do GNOME.

De forma contrária à uma visão pré-concebida, este estudo considera que a práxis dos hackers fundamenta uma cultura que “diz respeito ao conjunto de valores e crenças que emergiu das redes de programadores de computador que interagem *on-line* em torno de sua colaboração em projetos autonomamente definidos de programação criativa” (Castells, 2003, p.38). Assim, essa cultura desempenha um papel central ao longo da história de desenvolvimento dos principais símbolos tecnológicos da atual sociedade em rede (como o PC - *Personal Computer*, a Internet e softwares como o sistema operacional GNU/Linux) até os dias atuais. Isto porque é esta cultura hacker que dá sustentação ao ambiente fomentador de inovações tecnológicas significativas mediante a cooperação e comunicação livre, como também acaba permitindo a conexão entre o conhecimento originado em universidades e centros de pesquisas com os sub-produtos empresariais que difundem as tecnologias da informação no “mundo dos átomos” - isto é, na materialidade da sociedade capitalista.

1. O GNOME e a gênese do movimento pela liberdade tecnológica.

Para melhor compreender a práxis e a organização social relativos à essa cultura hacker que se fazem presentes em comunidades virtuais como a GNOME, Manuel Castells (2003) afirma que é importante considerar a gênese do movimento de software livre^v que surge a partir da década de 1980. Até essa época, as aspirações de grande parte dos tecnólogos computacionais se direcionavam para a expansão da fronteira do “mundo dos bits” por meio do desenvolvimento colaborativo de um sistema operacional capaz de rodar em todos os tipos de computadores e, ao mesmo tempo, conectar os servidores à Internet. O sistema operacional UNIX proporcionava essas condições dentro de um ambiente de alta colaboração tecnológica. Elaborado por Ken Thompson do Laboratório Bells em 1969, o UNIX teve seu código fonte disponibilizado no seu processo de distribuição inicial.

Esta condição de abertura possibilitou que outros centros de pesquisas como, por exemplo, o *Computer System Research Group* da Universidade Berkeley na Califórnia

aperfeiçoassem esse sistema. Assim, no final da década de 1970, além de expandir a Internet por meio do projeto de suporte UNIX para os protocolos da Arpanet e da Usenet ao mesmo tempo, estudantes de Berkley (liderados pelos tecnólogos Bill Joy e Chuk Halley) criaram o mais avançado sistema operacional da época: o *Berkeley Software Distribution (BSD)*. “O UNIX tornou-se um ambiente de software para todo tipo de sistema, libertando assim os programadores da necessidade de inventar linguagens específicas para cada máquina: o software tornou-se portátil, o que permitiu a comunicação entre computadores e programação de computador cumulativa” (Castells, 2003, p.39).

Dentro desse contexto de colaboração tecnológica, “a ação preponderante dos hackers (tecnólogos), (des) organizada em torno da Internet, e até esse momento amplamente identificado com a cultura técnica do UNIX, não se preocupava com serviços comerciais. Queriam melhores ferramentas e mais Internet, e PCs baratos de 32 bits, prometendo pôr uma coisa e outra ao alcance de todos” (Raymond *apud* Castells, p.40). Assim, o movimento do software livre - como um movimento político de base tecnológica - emerge em 1984, quando um programador do Laboratório de Inteligência Artificial do MIT, o Richard Stallman, reagiu à decisão comercial da AT&T de reivindicar direitos de propriedade^{vi} sobre o sistema operacional UNIX, fechando então o seu código-fonte. Stallman, junto com um grupo de programadores, lançou-se então no árduo processo de desenvolvimento de um outro sistema operacional, inspirado no UNIX, que foi batizado com a sigla GNU (um acrônimo recursivo para “*GNU is Not Unix*”). Concomitante ao desenvolvimento desse sistema, Stallman criou a *Free Software Foundation (FSF)* e foi o protagonista da elaboração de um arcabouço jurídico que garantiu, além da abertura do código-fonte, a plena liberdade de uso, aperfeiçoamento e distribuição dessa tecnologia - propondo, assim, a substituição da lógica do *copyright* pelo “*copyleft*”^{vii}. O sistema operacional GNU foi então disponibilizado na Rede segundo os termos da GPL (*General Public License*), tendo o apoio da FSF para dar sustentabilidade a esse projeto.

No entanto, apesar de ter criado as condições políticas favoráveis à manutenção e ao desenvolvimento de softwares livres, um dos sistemas centrais do Projeto GNU, o Kernel^{viii} *HURD*, não funcionou efetivamente até o ano de 1996. Segundo Castells (2003), isso não ocorreu por incapacidade técnica de programação – já que Stallman, por exemplo, é reconhecido como um dos maiores programadores que passaram pelo MIT. Essa demora em desenvolver um projeto de tamanha complexidade aconteceu, na visão de Castells, em decorrência dos hackers do projeto GNU não terem percebido plenamente o poder de colaboração da Internet. “Só uma rede de centenas de milhares de cérebros trabalhando cooperativamente, com divisão do trabalho espontânea, e coordenação maleável, mas eficiente, poderia levar a cabo a tarefa extraordinária de criar um sistema operacional capaz de lidar com a complexidade de computadores cada vez mais potentes interagindo por meio da Internet” (Castells, 2003, p.41). Enquanto isso, na mesma década de 1990, a multinacional Microsoft comercializou e difundiu, com a força de um mercado sem concorrência, um sistema operacional proprietário (isto é, com o código-fonte fechado) mesmo sendo considerado tecnologicamente inferior. No entanto, pelo fato de ter uma interface (aparência) amigável e de fácil manuseio, o sistema fechado da Microsoft atendeu a uma demanda cada vez mais crescente de usuários comuns de computadores. Como alternativa a esse contexto, surge então o sistema operacional GNU/Linux, que se desenvolve a partir do trabalho de Richard Stallman (2002), mas com um método de desenvolvimento mais subversivo que o próprio Projeto GNU, descoberto meio que por caso pelo então estudante da Universidade de Helsinki da Finlândia, Linus Torvalds.

Em 1991, o então universitário Linus Torvalds, que havia adquirido o seu primeiro

computador pessoal (um PC 386 da Intel), quis utilizá-lo com um sistema operacional UNIX. No entanto, como não tinha recursos financeiros para ter acesso a essa tecnologia, dedicou seu trabalho de conclusão de curso enquanto estudante de ciência da computação para desenvolver seu próprio Kernel. Para empreender tal façanha, ele tomou como referência o próprio Kernel do UNIX, para máquinas 386 e utilizou o compilador GCC para linguagem C. Ao terminar sua primeira versão, ele a denominou de “Freix” - mas, segundo Castells, o administrador do servidor que ele utilizava, em homenagem ao próprio criador, o denominou de Linux. Devido as dificuldade encontradas ao longo do seu “projeto final de curso”, Linus divulgou o código-fonte do Kernel num *Newsgroup*^{ix} na Internet, em agosto de 1991, pedindo então ajuda e cooperação de outros programadores no desenvolvimento desse software livre.

Como resultado dessa iniciativa, Linus acabou obtendo uma extraordinária colaboração vinda de todos os cantos do planeta que, de alguma forma, tinham acesso à rede. Segundo Raymond (2004), no início do projeto Linux - por volta de 1991, não era estranho uma nova versão do Kernel ser lançada mais de uma vez por dia. Assim, “divulgações rápidas, ampla cooperação e total abertura da informação permitiram a testagem extensiva e a depuração do código de tal forma que, em 1993, o GNU/Linux era um sistema operacional melhor que os sistemas UNIX patenteados” (Castells, 2003, p.41). Em 2001, além dos sistemas UNIX comerciais passarem a adotar o Kernel Linux, segundo Castells (2003), já havia pelo menos 30 milhões de usuários do GNU/Linux em todo mundo.

1.2. Surge o projeto GNOME

Mesmo sendo difundido como um software de excelente qualidade tecnológica, no início do seu processo de desenvolvimento, o sistema operacional GNU/Linux também era considerado um software que exigia, por parte dos usuários, um nível de conhecimento técnico mais elevado para o seu bom funcionamento. Isto porque, por terem sido desenvolvidos “por tecnólogos” e “para tecnólogos”, o GNU/Linux utilizava um padrão de interface gráfica (ícones, janelas, bordas, etc) menos intuitivo e com padrões simbólicos menos acessíveis às pessoas comuns - ao contrário de algumas versões proprietárias (de código fechado) como o sistema operacional *Windows* e o *Mac OS X* que estavam sendo desenvolvidos por grandes corporações de Tecnologia da Informação (T.I.) que priorizavam mais estas questões de interface gráfica. Assim, para que o GNU/Linux se difundisse de forma mais ampla, fazia-se necessário o desenvolvimento de um ambiente de *desktop* que possibilitasse a existência de um sistema de interface com ícones e janelas entre outros aplicativos que seguissem os princípios de liberdade tecnológica e, ao mesmo tempo, que permitisse que um usuário comum pudesse facilmente interagir e acessar suas funcionalidades.

Por volta de 1996, o primeiro projeto que surge para tentar suprir essa lacuna foi denominado de *K – desktop Environment* (KDE). No entanto, o fundador desse projeto, Matthias Ettrich, optou por utilizar uma tecnologia até então de código fechado - o software *Qt toolkit* da empresa *TrollTech* - em seus pacotes oficiais. Assim, mesmo não sendo um software comercial, o *desktop* KDE causou muitas discordâncias no interior do movimento pelo Software Livre devido ao fato desse sistema não seguir, na sua integralidade, os princípios de abertura e liberdade da filosofia GNU. A partir desse contexto, em 1997, dois hackers mexicanos - Federico Mena e Miguel de Icaza - se articularam para então desenvolver um projeto de *desktop* que não apenas fornecesse um sistema aberto de interface gráfica mais inteligível para o uso, mas que também seguisse de forma integral os princípios de liberdade tecnológica e colaboração associados ao sistema operacional GNU/Linux. Este projeto foi denominado de *GNU - Network Object Model Environment* - mais conhecido e difundido pela

sigla GNOME.

Além de contar com o apoio e colaboração crescente de hackers de vários países, no ano seguinte após o lançamento da sua versão primeira versão, o projeto GNOME passa também a ter o apoio de desenvolvedores de softwares que eram pagos por empresas de T.I., à exemplo da *Red Hat* e a *Eazel*. Segundo Icaza (2002), com o apoio direto e a participação de vários hackers espalhados pela Rede juntamente com esses desenvolvedores contratados, em 1998, o projeto GNOME lança a sua versão (1.0). Posteriormente, no ano de 2000, já com intuito de fortalecer ainda mais a missão do projeto de desenvolver uma plataforma computacional livre, é criada a Fundação GNOME. Essa Fundação, com sede na Califórnia nos EUA, tornou-se então numa espécie de organização formal que dá apoio institucional à comunidade virtual. É ela que representa formalmente o Projeto no “mundo dos átomos”, sendo gerida de forma virtual por um Conselho Diretor composto por sete pessoas de diferentes partes do mundo que são eleitas pela comunidade.

2. O Projeto GNOME e o modelo colaborativo de produção entre pares

O Projeto GNOME, que dá o sentido de existência para toda esta comunidade virtual de hackers, gira em torno de duas metas. De um lado, tendo como foco o chamado “usuário final”, o projeto visa o desenvolvimento de um ambiente de *desktop* para distribuições do GNU/Linux, com uma interface gráfica (*Graphical User Interface – GUI*) e um conjunto de aplicativos (softwares) que facilitem ao máximo a interação desse usuário com o computador (*hardware*). Por isso, de forma imbricada ao desenvolvimento desse ambiente gráfico GNOME, se faz presente também o processo de tradução dos softwares que compõem o desktop para mais de 60 idiomas no mundo, abrangendo, assim, desde idiomas ocidentais (como Espanhol, Francês e Português) que usam o mesmo alfabeto, como também línguas e dialetos que usam alfabetos completamente distintos - como, por exemplo, Chinês, Grego, Árabe, Hindi e Hebraico. Por outro lado, o projeto acaba tendo também como foco a estruturação de uma plataforma de desenvolvimento, contendo uma coleção de Interfaces de Programação de Aplicativos (APIs), além de diversas ferramentas de programação, que são utilizadas por hackers tanto do projeto, como também de outras comunidades de softwares livres espalhadas pelo ciberespaço.

Partindo desta realidade, é importante frisar que o processo de desenvolvimento de softwares livres em comunidades dessa natureza é singular em relação a um aspecto pragmático do modo de vida hacker. Enquanto “a última revolução industrial marcou a transição para uma sociedade que contava com os resultados científicos; os hackers chamam a atenção para o fato de que, na Era da Informação, mais importante que os resultados científicos é o modelo de produção entre pares^x, que permite obtenção desses resultados.” (Himanem, 2001, p.73) Ou ainda, “a emergência do software livre e o sucesso fenomenal dos seus pioneiros – como o sistema operacional GNU/Linux, o servidor de rede Apache, Perl, BIND – e muitos outros projetos têm nos forçado a realizar uma revisão no paradigma dominante sobre produção.” (Benkler 2002, p. 371-372) [Tradução nossa]. Assim, apesar da gênese do desenvolvimento de tecnologias como a própria Internet ser um bom exemplo de inovação, liberdade e abertura, os projetos tecnológicos produzidos de forma livre e colaborativa como o *Linux* ou o GNOME, podem ser considerados um avanço ainda maior. Isto porque, durante o processo (mediado por computador) de desenvolvimento desses softwares, os hackers vêm então elaborando um modo subversivo de auto-organização para a produção e distribuição de softwares livres na Internet.

“Quem pensaria mesmo há cinco anos atrás que um sistema operacional de classe mundial poderia surgir como que por mágica pelo tempo livre de milhares de colaboradores espalhados por todo o planeta, conectados somente pelos tênues cordões da Internet?”

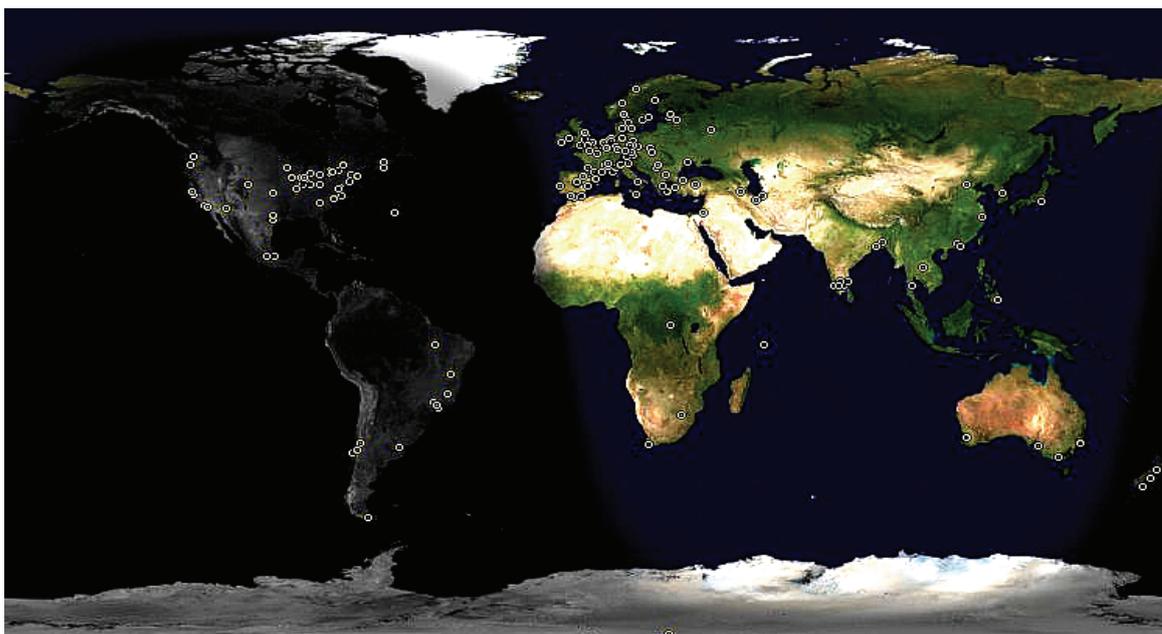
(Raymond, 2004, p.01) Segundo o hacker Eric Raymond (2004), esta especulação certamente não partiria dele, pois - assim como quase a totalidade dos programadores do início da década de 1990 - ele defendia a tese de que softwares de alta complexidade como um kernel ou um ambiente *desktop*, necessitavam ser desenvolvidos dentro de um “modelo catedral”. Dito de outra maneira, estes sistemas computacionais, segundo o padrão de desenvolvimento anterior ao surgimento do Linux, deveriam ser elaborados de forma centralizada e isolada por um pequeno grupo de desenvolvedores especialistas, que só disponibilizariam o resultado de seu trabalho quando percebessem que uma primeira versão de determinado software estivesse significativamente pronta - assim, como acontece na maioria das empresas (firmas) de produção de softwares que têm o código-fonte fechado. Esse modelo Raymond (2004) denominou de *Catedral*.

Dentro desse contexto, o chavão “libere cedo (uma versão do software) e freqüentemente, delegue tudo que você possa, esteja aberto ao ponto da promiscuidade,” chegou como um grande choque para o universo da engenharia de softwares complexos. Isto porque a maioria dos desenvolvedores, incluindo o próprio Eric Raymond, acreditava que esta era uma má política para projetos dessa natureza, tendo em vista que os possíveis defeitos (denominado de *bugs*) poderiam ser passados para os usuários.

Por outro lado, no início do projeto Linux - por volta de 1991, a relação entre usuários e desenvolvedores de um projeto de software livre foi completamente alterada: com a disponibilização do código fonte, muitos usuários passam a ser co-desenvolvedores eficazes. Linus Torvalds percebeu que muitos usuários são hackers também e, portanto, eles podem diagnosticar problemas, sugerir correções e ajudar a melhorar o código muito mais rapidamente do que um programador ou até mesmo uma equipe técnica de forma isolada, fechada. Em outras palavras, “dados olhos suficientes, todos os erros são triviais.” (Raymond, 2004, p.01) Assim, neste modelo aberto, não era estranho uma nova versão do Kernel Linux ser lançada mais de uma vez por dia - o que para muitos programadores e engenheiros computacionais, seria humanamente impossível a partir de um modelo tradicional. “Nenhuma catedral calma e respeitosa aqui -- ao invés, a comunidade Linux pareceu assemelhar-se a um grande e barulhento bazar de diferentes agendas e aproximações (adequadamente simbolizada pelos repositórios do Linux, que aceitaria submissões *de qualquer pessoa*) de onde um sistema coerente e estável poderia aparentemente emergir somente por uma sucessão de milagres” (Raymond, 2004, p.01).

No caso do Projeto GNOME, este “grande e barulhento bazar” acaba envolvendo mais de 300 pessoas entre tradutores, desenvolvedores e colaboradores em geral que juntos fazem parte de 59 países dos cinco continentes do planeta - como demonstra a figura 3. Mais especificamente, por meio da pesquisa quantitativa que fundamentou este trabalho, pôde-se observar que dos 297 Hackers catalogados (de janeiro a julho de 2006), quase a totalidade dos membros (99%) da comunidade são do sexo masculino e a maioria (33%) deles são de países da Europa Ocidental (como França, Inglaterra, Espanha e Alemanha), sendo outra boa parte (19%) dos países da América do Norte (México, Canadá e EUA). A América do Sul (Chile, Brasil e Argentina) chega a representar 5% dos membros. No entanto, existem também pessoas de países africanos e asiáticos como Índia, Irã, China, Vietnã, Tunísia, Togo, Madagascar e África do Sul; além da Oceania que se faz presente por meio de países como, por exemplo, Austrália e Nova Zelândia.

Figura 1 “Pontos” da rede mundial do Projeto GNOME



Fonte: <http://planet.gnome.org>

Entender, portanto, a dinâmica de trabalho dos hackers do GNOME é tentar compreender o “milagre” do desenvolvimento semestral de uma nova versão deste sistema computacional que une e mobiliza essa comunidade internacional, interligada apenas pelos liames digitais da rede mundial de computadores. Para alcançar então o desafio analítico de adentrar no enredo produtivo de uma nova versão da Plataforma e do *desktop* GNOME, faz-se antes necessário uma análise prévia da estrutura de times e setores que delineiam a base organizacional que mantém vivo este processo não-contratual de produção em rede.

2.1. GNOME: um bazar organizado?

Segundo German (2005), atualmente, o Projeto GNOME está associado a um sistema computacional composto por mais de 60 softwares entre bibliotecas e aplicativos do *desktop* e da plataforma. Em outras palavras, isto corresponde a mais de dois milhões de linhas de código - que podem se ampliar ainda mais a cada nova versão.

O fato do Projeto ser subdividido em diversos aplicativos e bibliotecas acaba então estimulando uma forma de divisão da produção em *módulos^{xi} de desenvolvimento* - de forma proporcional e correspondente ao número de softwares, que atualmente gira em torno de 66. Estes módulos de desenvolvimento são agrupados e sistematizados pelo Sistema de Controle de Versões (*Concurrent Versions System CVS*) do GNOME e contam com a coordenação de, no mínimo, um programador efetivo (para cada módulo) que recebe o nome de *mantenedor*. É este mantenedor o hacker responsável por planejar os rumos de um determinado software que compõem o sistema, além de gerir o envio de doações de códigos de programação - apelidadas de *patches*.

Estas doações podem estar relacionadas tanto à correção de um erro (*bug*) no programa, ou ao envio de códigos com novas funcionalidades e especificidades que algum usuário queira agregar àquele determinado software. E é neste ponto que reside, segundo Raymond (2004), uma das principais virtudes deste modelo aberto de produção colaborativa entre pares - que ele denominou de “bazar”: enquanto no modelo Catedral, presente em boa parte das empresas de software no mundo, os erros nos programas são tratados de forma fechada, por meio de um exame minucioso e demorado feito por um número pequeno de

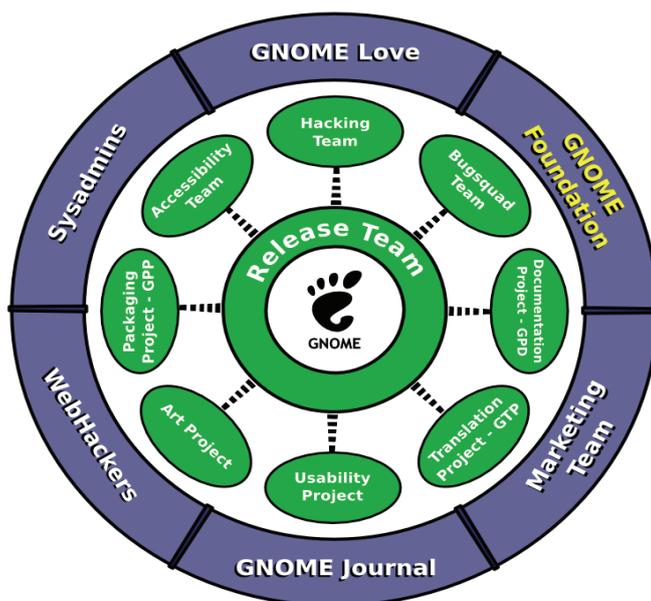
técnicos; “na visão bazar, por outro lado, você (mantenedor) assume que erros são geralmente triviais, muito rapidamente, quando expostos para centenas de ávidos co-desenvolvedores triturando cada nova liberação.” (Raymond, 2004, p.01)

Seguindo esta filosofia, por exemplo, o mantenedor de qualquer módulo do Projeto GNOME disponibiliza de forma freqüente na web da comunidade uma nova versão com o intuito de ter sempre mais correções (doações) de qualquer outro hacker ao redor do mundo que queira usar e contribuir. Dessa forma o mantenedor do pacote consegue reduzir o trabalho e o custo que teria para manter aquele determinado software de forma isolada. Além disso, quando um usuário/desenvolvedor passa a fazer doações constantes de *patches* ele passa também a ter grandes possibilidades de dividir o posto de mantenedor ou, até mesmo, assumir exclusivamente tal cargo - caso o atual mantenedor considere que não tem mais tempo livre e o indique para assumir tal função.

Dentro do contexto da estrutura organizacional da comunidade, todos os mantenedores e colaboradores dos módulos de desenvolvimento de software irão compor apenas um (nesse caso o de *Hacking*) dos 15 sub-projetos ou Times onde está dividido e alocado o trabalho da comunidade como um todo – como pode-se perceber no organograma do Projeto GNOME representado na Figura 2.

No universo destes quinze sub-projetos (times), todos têm a coordenação de pelo menos um líder ou mantenedor que trabalha de forma mais efetiva e periódica, além da participação de outros colaboradores mais eventuais. Enquanto na maioria dos times é comum a existência de um a dois coordenadores, em alguns módulos de desenvolvimento é possível se encontrar até cinco mantenedores, dependendo da complexidade do software e do interesse de se manter o software. Para facilitar o entendimento sobre essa complexa estrutura organizacional, estes times podem ser qualificados de forma geral entre aqueles que estão relacionados às “atividades fim” - isto é, que estão diretamente ligados ao processo de desenvolvimento dos softwares; e aqueles que são responsáveis por “atividades meio”, ou seja, que realizam atividades que oferecem uma espécie de suporte ao processo e desenvolvimento tanto do desktop como da plataforma.

Figura 2 - Organograma do Projeto GNOME



Fonte: www.gnome.org

A partir desta complexa estrutura organizacional e, ao mesmo tempo, tendo como base esse subversivo modelo colaborativo de produção tecnológica entre pares, a GNOME, desde de 2005, começa a adotar uma postura até então inexistente no universo dos projetos de software livre: lançar uma nova versão do seu *desktop* e da sua plataforma livre de desenvolvimento dentro de um prazo pré-estabelecido de seis meses.

Assim, cada versão torna-se fruto de todo um enredo geral de desenvolvimento tecnológico, espontaneamente colaborativo, que a cada seis meses termina e se inicia novamente com o lançamento de uma versão mais moderna, anunciada pelo *Release Manager* (Gestor de Versões) nas listas de emails da comunidade. No entanto, para efeito da análise adotada nesse trabalho, adotar-se-á um corte temporal no processo de desenvolvimento do GNOME. Como ponto de partida, escolheu-se o mês de setembro de 2005, quando foi lançada a versão 2.12^{xii} do *desktop*, por meio do email “*Celebrating the release of GNOME 2.12!*”, enviado por Elijah Newren, atual *Release Manager* da GNOME para a lista de anúncios da comunidade.

2. Planejando o enredo de uma nova versão

Como a inovação tecnológica é o espírito balizador dessa organização, após o lançamento de toda nova versão, a comunidade GNOME começa a se planejar para o lançamento de uma outra nova versão do seu *desktop* (de forma semelhante como acontece numa comunidade de escola de samba após seu desfile na Marquês de Sapucaí) - que neste caso estudado é a 2.14.0. É então nesse momento que a equipe responsável pelo lançamento de versões – *Release-Team* - se reuni no seu canal IRC (*Internet Relay Chat*) para elaborar uma proposta de calendário. Assim, logo após a definição de uma proposta, o *Release Manager* lança um outro email para lista de discussão dos desenvolvedores, com uma sugestão de calendário para o desenvolvimento geral da futura versão. Esta proposta passa então a ser discutida entre todos os desenvolvedores (mantenedores) e co-desenvolvedores de cada software por meio da lista de emails e do Canal, buscando-se, assim, a construção de um consenso que tem como mediador o *Release Manager*.

A partir de um calendário de desenvolvimento de seis meses, que fica disponível no site da comunidade, todos os mantenedores de softwares começam a se organizar para definir quais as futuras novidades e inovações que cada pacote terá na próxima versão. Quando pronta, todas as metas de inovação relativas aos 66 softwares do sistema são postadas numa página web de planejamento das versões - denominado de *RoadMap* (<http://live.gnome.org/RoadMap>). Além disso, após a definição do planejamento, abre-se uma nova etapa de discussão sobre a possibilidade de entrada de novos softwares para serem inseridos na plataforma ou no *desktop*.

Nesse momento, qualquer hacker pode propor um novo pacote de solução para ser agregado ao sistema GNOME. Da mesma forma como aconteceu na aprovação do calendário, essa proposta é discutida pelos membros desenvolvedores da comunidade, numa lista de email específica, tendo como mediador da discussão o *release-manager*. A aprovação ou não da proposta só é definida meses depois, quando *realease-manager* faz um levantamento do resultado da discussão.

2.2.1. A colaboração dos usuários e a “temporada de caça aos bugs”!

Para por em prática o processo de abertura e liberação imediata do código-fonte, presente nesse modelo de desenvolvimento aberto, os mantenedores dos pacotes de software da GNOME sempre buscam tratar os usuários do *desktop* como co-desenvolvedores - assim como acontece de forma semelhante no desenvolvimento do Linux e da distribuição Debian.

De forma paralela ao processo de desenvolvimento da nova versão, os erros de funcionamento - mais conhecidos como “bugs” - da versão atual são reportados por usuários de todos os continentes do globo para o *Bugzilla* da comunidade. Esses erros são avaliados pela equipe de triagem de bugs - a *Bugsquad* - para certificar-se de que os principais problemas não passarão despercebidos pelos mantenedores dos pacotes.

Assim, dada a contribuição significativa de usuários que se comportam como *beta-testers* e co-desenvolvedores, praticamente todo o problema relativo a algum aplicativo do *desktop* passa a ser detectado rapidamente e a sua respectiva solução pode também acabar tornando-se óbvia para alguns, que acabam até contribuindo com sugestões de novos códigos para a resolução de alguns *bugs*. Essas contribuições acabam então servindo tanto para a implementação de uma versão estável do *desktop* atual (isto é, uma versão com os *bugs* corrigidos) como também para a correção de possíveis erros da versão em desenvolvimento.

Na visão de Raymond (2004), seria então exatamente nesse ponto que se caracterizaria a diferença central entre os modelos *Catedral* (proprietário) e *Bazar* (acadêmico aberto). No primeiro os problemas de desenvolvimento levam meses pelo fato de passar pelas mãos de poucas pessoas especialistas, necessitando então de longos intervalos para o desenvolvimento de uma nova versão - que também não será perfeita e trará novos problemas. Já no modelo bazar, por outro lado, os erros são assumidos como fenômenos mais triviais pelo fato de serem logo expostos para centenas de ávidos co-desenvolvedores dispostos a elaborar e contribuir para o avanço tecnológico de uma nova versão. Dessa forma, numa comunidade de desenvolvimento tecnológico como a GNOME, uma nova versão é liberada a cada seis meses para se ter mais correções e, como um “efeito colateral”, passa-se a ter a possibilidade de ter também mais contribuições.

2.2.2. Períodos de congelamento: entram em cena os membros não-programadores.

Na medida em que o período de desenvolvimento do calendário de planejamento avança, começam a acontecer os “congelamentos” (*freezes*) agendados. Este método de “parada”, em termos de programação de códigos, cria condições para que outras equipes de colaboração transversal da comunidade virtual (ligados à atividades de tradução, documentação, usabilidade e acessibilidade) possam entrar em cena. Como demonstra o calendário de planejamento, esses períodos de parada acontecem após o anúncio e implementação de novas funcionalidades no sistema, podendo ser esquematizadas da seguinte forma:

- a) **Congelamento de Funcionalidades** – Este congelamento está relacionado aos softwares da plataforma (*API/ABI Freeze*) como também aos softwares do *desktop* (*Feature and Module Freeze*). Esta parada acontece com o intuito de permitir que a equipe do Projeto de Documentação (*GNOME Documentation Project*) comece a trabalhar na elaboração de manuais de uso e definição de padrões (tecnológicos e de escrita) para elaboração dos mesmos. Na elaboração da versão 2.1.4, esses congelamentos ocorreram entre os dias 16 e 18 de janeiro de 2005.
- b) **Congelamento de Interface de Usuário** – neste período, nenhuma das interfaces gráficas (ícones, botões e janelas) dos programas que compõem o *desktop* pode ser alterada. Neste momento, além da equipe do Projeto de Documentação continuar a elaborar os manuais e as definições dos padrões, entra em cena também a equipe do Projeto de Usabilidade (*GNOME Usability Project*). Esta última equipe fica então responsável nesse momento por revisar as interfaces a partir de especificações e padrões técnicos que são estabelecidos pelo setor de engenharia de software no mundo. Dessa forma, eles acabam auxiliando os hackers da comunidade na criação

de aplicações intuitivas, além de orientá-los por meio da elaboração de projetos e diretrizes gráficas para próprio *desktop* GNOME.

- c) **Congelamento das mensagens de comando dos softwares** – Este período visa inibir que qualquer mantenedor de algum software possa acrescentar qualquer mensagem “extra” nas barras de menu dos aplicativos, como “Salvar” ou “Importar Arquivo”, por exemplo. É partir então desse congelamento que as várias equipes do Projeto de Tradução (*GNOME Translation Project*)^{xiii} começam a trabalhar dentro de um ambiente mais estável. Isto porque, como esse projeto é composto por mais de 130 equipes voluntárias de tradução espalhadas pelo mundo, a modificação de uma mensagem após esse congelamento, pode trazer um impacto bem negativo em termos de re-trabalho.
- d) **Congelamento de Código** – esse último congelamento é o mais rígido. A partir desse momento, qualquer modificação fica impedida de ser implementada, salvo algum erro drástico que – por motivo de força maior tenha-se que ser modificada - terá que ter o aval do release-team antes. Entre em cena dessa vez, a equipe de marketing (*GNOME Marketing Team*)^{xiv} que irá preparar as notas que serão lançadas para imprensa e meios de comunicação em geral sobre a nova versão do GNOME.

Após a conclusão desse **ciclo semestral de produção entre pares**, o *desktop* passa a ser disponibilizado e, posteriormente, utilizado por diversas distribuições do sistema operacional GNU/Linux, que são desenvolvidas tanto por empresas como por outras comunidades on-line espalhas pela rede mundial de computadores. De acordo com as informações que constam no site do Projeto GNOME, mais de 140 distribuições diferentes do GNU/Linux no mundo adotam esse *desktop* livre.

Como resultado deste processo mundial de distribuição, segundo Foster (2005), sem contar com a maioria dos usuários domésticos em todos os países que não são contabilizados, mais de 200 mil pessoas em Extremadura e Andaluza na Espanha, um milhão na China e 400 mil pessoas nos telecentros de São Paulo, no Brasil, usam o GNOME como o ambiente de *desktop* dos seus computadores. Na Bahia, de acordo com Vaz (2005), mais de noventa mil pessoas estão cadastradas como usuários nos 101 Infocentros do Governo do Estado que adotam esse *desktop* livre em todas as estações. Assim, em outras palavras, para além da complexidade do processo de produção, o resultado do trabalho colaborativo desses hackers consegue ser distribuído livremente para os cinco continentes do planeta.

2.2.3. Organização social do Trabalho e relações de poder

Apesar de toda essa organização social do trabalho em módulos e Times de desenvolvimento, não é possível perceber uma divisão rígida entre as funções de cada membro no GNOME - como, por exemplo, uma divisão entre aqueles que são mantenedores de pacotes, desenvolvedores esporádicos e usuários. Ao contrário. O projeto GNOME é completamente dinâmico e flexível. Qualquer usuário que se beneficia desse sistema tem condições tanto para contribuir e fazer parte de forma mais efetiva, como também para sair a qualquer momento da organização ou limitar-se a fazer algumas contribuições esporádicas.

Todavia, esta sinergia na divisão dos papéis sociais no interior do grupo está subordinada ao nível de contribuição que cada um compartilha e do vínculo que se estabelece com a comunidade. Quando um determinado usuário passa a fazer contribuições significativas que são aceitas periodicamente pelos membros da comunidade, por mérito destas doações, o usuário pode até acabar recebendo o respeitado cargo de “mantenedor” do módulo que ele

está ajudando a desenvolver. Por causa dessa condição de flexibilidade e abertura, os próprios membros da comunidade podem afirmar de forma sincera e não utilitária que, mesmo tendo algumas características diferenciadas entre si, “os usuários são a base de toda a comunidade, somente por uma simples razão: todos os contribuidores eram usuários antes de começarem a contribuir.” (UNTZ, 2006, p.01) [TRADUÇÃO NOSSA]

Com a função de mantenedor de algum módulo do GNOME, o vínculo desse colaborador em relação à comunidade se amplia a ponto de tornar mais difícil um possível desligamento em relação ao Projeto – mesmo que não exista nenhum tipo de contrato além da obrigação que ele dá livremente a si mesmo. Por outro lado, como mantenedor de algum pacote, o nível de respeito e reconhecimento entre seus pares se amplia de forma mais significativa. Qualquer hacker que assuma essa função pelo mérito de suas contribuições adquire então mais poder de decisão sobre as questões que envolvem o projeto. “No final das contas, a direção do projeto é dada por qualquer um que escreva códigos. (...) Então se você desenvolve código ou se você tem influência direta sobre pessoas que desenvolvem códigos... no final este é o processo de tomada de decisão.”^{xv} Em outras palavras, isto significa dizer, portanto, que a dinâmica de poder na organização social do trabalho nesse Projeto Internacional irá depender do mérito associado às relações de trocas e compartilhamento de códigos que são empreendidas no interior do grupo.

5. Considerações finais sobre essa “organização atual”

O Conjunto das relações humanas mediadas por computadores, mais do que representar o “tecido digital” que compõe a sociedade contemporânea, parece desafiar boa parte dos conceitos e práticas que haviam sido ditados pelas ciências humanas, em particular, pela ciência da administração. “Hoje, um bom número de restrições desapareceu devido à disponibilidade de novas ferramentas de comunicação e de coordenação, e podemos pensar modos de organização dos grupos humanos, estilos de relações entre os indivíduos e os coletivos radicalmente novos, sem modelos na história e nas sociedades animais.” (Lévy, 1999, p. 132)

Dentro desse contexto, como um fenômeno organizacional da atual era do informacionalismo, as comunidades de produção e distribuição de softwares livres parecem ser então melhor compreendidas sob a expressão “organização atual” ou “comunidade atual” - citada inicialmente por Lévy (1999) - do que os termos “comunidades *on-line*” ou “comunidade virtual”, comumente utilizados. Afinal, de acordo com os dados de (apenas) um dos maiores repositórios de código aberto do mundo - o site *SourceForge.net* - é possível ser encontrados por lá mais de 140.000 projetos de software livre registrados. Estes projetos são desenvolvidos por diversas organizações “comunitárias” que somam juntas mais de 1 milhão e 500 mil hackers de todos os continentes do planeta que interagem por meio de um processo econômico comum: produzir e compartilhar códigos de forma aberta, colaborativa.

No universo dessas “comunidades atuais” de software livre, existem tanto aquelas menos expressivas do ponto de vista quantitativo - formadas apenas por dois ou três hackers, assim como existem grandes grupos formados por mais de 300 hackers como, por exemplo, caso do Projeto GNOME. Com um processo de organização aberto e quase que espontâneo (isto é, sem nenhum planejamento estratégico estrutural), a comunidade de hackers que compõem este projeto consegue se autogerir para o desenvolvimento e a distribuição semestral de um ambiente de *desktop* e uma plataforma de desenvolvimento compostos por mais de 60 softwares, entre aplicativos e ferramentas de todos os níveis técnicos. Para tanto, a comunidade GNOME divide-se de forma espontânea em quinze sub-projetos ou Times onde ficam alocados todo o trabalho da comunidade, desde ações diretamente ligadas ao processo

de desenvolvimento, até aquelas atividades que dão suporte para essa finalidade principal.

5.1. Novos horizontes organizacionais na Sociedade em Rede?

Todavia, na visão de autores como Kollock (1998), Benkler (2006) e Silveira (2006), as comunidades de software livre são apenas um exemplo proeminente de um fenômeno muito mais amplo na Sociedade em Rede. Para além do desenvolvimento de códigos abertos, o modelo de organização que surge ao longo do processo de compartilhamento dos hackers, influência e parece reconfigurar a forma de produção e distribuição da informação na economia contemporânea. Assim, de acordo com Benkler (2006), a questão que mobilizou este presente artigo, pode ser reformulada e utilizada em novas pesquisas. Afinal, como cinquenta mil voluntários podem, com sucesso, ser co-autores da Wikipedia, a mais séria alternativa *on-line* da Enciclopédia Britânica? Como 4.5 milhões de voluntários se organizam para contribuir com o que sobra do ciclo computacional de seus computadores para criar o mais poderoso computador da Terra, [SETI@Home](#)?

Referências Bibliográficas

AGUIAR, Vicente M. de. **Os Argonautas da Internet: uma Análise netnográfica sobre a comunidade on-line de software livre do projeto GNOME à luz da teoria da dívida**. 2007. 110 f. Dissertação (Mestrado) - Curso de Administração, Escola de Administração da UFBA, Salvador, 2007. Disponível em:

<<http://www.adm.ufba.br/pub/publicacao/5/MAA/2007/685/DissertacaoGnomeVersaoFinal.pdf>>. Acesso em: 30 set. 2007.

BELL, David. **An Introduction to Cybercultures**. London: Routledge, 2001.

BENKLER, Yochai. Coase's Penguin, or, Linux and The Nature of the Firm. **The Yale Law Journal**, Nova York, p. 371-446. 01 dez. 2002.

_____. **The Wealth of Networks: How Social Production Transforms Markets and Freedom**. New Haven And London: Yale University Press, 2006. Disponível em: <<http://habitat.igc.org/wealth-of-networks/#ch-8>>. Acesso em: 30 ago. 2006.

CASTELLS, Manuel. **A Galáxia da Internet: reflexões sobre a Internet, os negócios e a sociedade**. Rio De Janeiro: Jorge Zahar Editor, 2003.

_____. **A sociedade em rede**. 8. ed. São Paulo: Paz E Terra, 2005. (A era da Informação: economia, sociedade e cultura; volume I).

_____. O Informacionalismo e a Sociedade em Rede. In: HIMANEM, Pekka. **A ética dos hackers e o espírito da era da informação: a importância dos exploradores da era digital**. 1. ed. Rio De Janeiro: Contexto, 2001. p. 137-154.

FOSTER, Glynn. **101 Things to Know about GNOME**. Disponível em: <<http://www.gnome.org/~gman/blog/02062005-1>>. Acesso em: 14 dez. 2005.

GERMAN, Daniel M.. The GNOME project: a case study of open source, global software development. **Software Process: Improvement and Practice**, Victoria, p. 201-215. 22 set. 2004. Disponível em: <<http://www3.interscience.wiley.com/cgi-bin/abstract/109630255/ABSTRACT>>. Acesso em: 02 mar. 2006.

_____. Software Engineering Practices in the GNOME Project. In: FELLER, Joseph; FITZGERALD, Brian; HISSAM, Scott A.; LAKHANI, Karim R.; Perspectives on Free and Open Source Software. 1. ed. Cambridge: The MIT Press, 2005, p. 211-225.

HIMANEM, Pekka. **A ética dos hackers e o espírito da era da informação: a importância dos exploradores da era digital**. 1. ed. Rio De Janeiro: Campus, 2001.

ICAZA, Miguel de. **The Story of the GNOME project**. Disponível em: <<http://primates.ximian.com/~miguel/gnome-history.html>>. Acesso em: 06 set. 2002.

LAKHANI, Karim R.; WOLF, Robert G.. Why Hackers Do What They Do: Understanding Motivation and Effort in Free/Open Source Software Projects. In: FELLER, Joseph; LÉVY, Pierre. **Cibercultura**. São Paulo: Ed. 34, 1999.

KOZINETS, Robert V. On Netnography: Initial Reflections on Consumer Research Investigations of Cyberculture. In: **Advances in Consumer Research**, 1998, 366-371.

_____. The field behind the screen: using netnography for marketing research in on-line communities. In: **Jornal of Marketing Research**. v.39, p.61-72, Feb. 2002.

KOLLOCK, Peter. **The Economies of Online Cooperation::** Gifts and Public Goods in Cyberspace. Disponível em:

<<http://www.sscnet.ucla.edu/soc/faculty/kollock/papers/economies.htm>>.

Acesso em: 10 ago. 1999.

NEWREN, E. Behind the Scenes: Elijah Newren. **GNOME Journal**, On-line, 21 abr. 2006.

Disponível em: <<http://gnomejournal.org/article/42/behind-the-scenes-elijah-newren>>.

Acesso em: 21 abr. 2006.

RAYMOND, Eric Steven. **A Catedral e o Bazar**. Disponível em:

<<http://www.geocities.com/CollegePark/Union/3590/pt-cathedral-bazaar.html>>. Acesso em:

24 dez. 2004.

_____. **Homesteading the Noosphere**. Disponível em:

<http://www.firstmonday.org/issues/issue3_10/raymond/index.html>. Acesso em: 16 dez.

1999.

SILVEIRA, Sérgio Amadeu. **A mobilização colaborativa e a teoria da propriedade do bem intangível**. 2005. 165 f. Tese (Doutorado) - Ciência Política, Departamento de Departamento de Ciência Política, USP, São Paulo, 2005. Disponível em:

<<http://twiki.softwarelivre.org/bin/view/TeseSA/WebHome>>. Acesso em: 06 jan. 2006.

STALLMAN, Richard M. **Free Software, Free Society: Selected Essays of Richard M. Stallman**. Boston: Joshua Gay, 2002.

TORVALDS, Linus. **Só por prazer: Linux, os bastidores da criação**. Rio de Janeiro: Campus, 2001.

UNTZ, Vincent. **The GNOME Community: end users**. Disponível em:

<<http://www.gnomejournal.org/article/46/the-gnome-community-end-users>>. Acesso em: 15

jun. 2006.

VAZ, Tiago Bortoletto. **Berimbau Livre: construção de uma solução livre para projetos de infocentros comunitários**. 2005. 85 f. Monografia (Bacharelado) - Departamento de Ciência da Computação, Universidade Federal da Bahia, Salvador, 2005.

i Este estudo científico foi realizado por meio de uma imersão netnográfica (pesquisa participante) dentro do ambiente computacional que compõe a comunidade on-line do Projeto GNOME, no período de dezembro de 2005 a fevereiro de 2007. Além dessa importante imersão, foram realizadas 12 entrevistas semi-estruturadas com membros de diferentes países que participam de forma atuante no Projeto Internacional. Para mais informações sobre esta pesquisa, ver a dissertação intitulada “*Os Argonautas da Internet: uma Análise netnográfica sobre a comunidade on-line de software livre do projeto GNOME à luz da teoria da Dádiva*” que se encontra disponível no linK abaixo: <http://www.adm.ufba.br/pub/publicacao/5/MAA/2007/685/DissertacaoGnomeVersaoFinal.pdf>

ii Neste artigo, são consideradas como “comunidades virtuais” ou “comunidades on-line” os agrupamentos sociais que se manifestam na Internet, onde o processo de comunicação é essencialmente mediado por computadores. Segundo Kozinets (1998), o uso do termo “comunidade” para classificar tais agrupamentos de pessoas, justifica-se pelo fato desse tipo de organização social ter em geral uma existência perene, dedicar-se

de forma organizada a objetivos e ações comuns, além de possuir uma existência “real” para seus membros, independentemente das proximidades geográficas. Por isto, é sempre importante ressaltar que “uma comunidade virtual não é irreal, imaginária ou ilusória, trata-se simplesmente de um coletivo mais ou menos permanente que se organiza por meio do novo correio eletrônico mundial” (Lévy, 1999, p.130).

iii O *desktop* é um sistema computacional composto por um conjunto de aplicativos (softwares) que seguem e definem um padrão de interface gráfica (*Graphical User Interface* - GUI) e, ao mesmo tempo, permitem a interação do usuário comum com as funcionalidades de um determinado sistema operacional. Assim, pode-se dizer que toda a arquitetura simbólica e o padrão gráfico (visual) de menus, ícones, áreas de trabalho e “janelas” são definidos pelo sistema de *desktop*.

iv A plataforma é a infraestrutura tecnológica que serve de base para o desenvolvimento das aplicações do GNOME. Tecnicamente, ela é um conjunto (repositório) de bibliotecas de objetos, funções e estruturas de dados que podem ser reutilizadas no desenvolvimento de novas aplicações. Ou seja, ela é uma espécie de “armazém” aberto, de caixa aberta cheia de códigos pré-definidos, que serve de referência para qualquer desenvolvedor interessado em utilizá-la.

v Vale sublinhar que, de maneira contrária ao sentido trazido pela tradução realizada na 1ª edição do livro *A galáxia da Internet* de Castells, o termo *free software* não significa “software gratuito”. Isto porque, segundo o próprio fundador da *Free Software Foundation*, “software livre é uma questão de liberdade, não de preço. Para entender o conceito, você deve pensar em 'liberdade de expressão', não em 'cerveja grátis'. Software livre se refere à liberdade dos usuários executarem, copiarem, distribuírem, estudarem, modificarem e aperfeiçoarem o software” (Stallman 2002, p.41) [TRADUÇÃO NOSSA].

vi Reivindicar direitos de propriedade sobre um software significa transformá-lo em uma mercadoria. Neste processo, um usuário ao pagar para adquirir o direito de uso de um determinado software, acaba não tendo acesso ao código-fonte desse sistema, ficando assim impossibilitado de modificá-lo e compartilhá-lo posteriormente. Assim, apesar desta atitude mercadológica da ATT de fechar o código do UNIX ter sido um marco na comunidade hacker no ano de 1979, ele não foi um caso isolado. O avanço desta lógica de mercado no setor de desenvolvimento de softwares se deu, principalmente, com a criação da Micro-Soft (futura *Microsoft Corporation*) por Bill Gates e Paul Allen, em 1975, com o objetivo de desenvolver e comercializar interpretadores da linguagem de programação BASIC.

vii *Copyleft* é uma característica atribuída à determinadas obras publicadas sob licenças livres - como a GPL (General Public License) ou algumas licenças da Creative Commons - que obrigam que outros distribuam obras derivadas somente sob uma licença (livre) idêntica a que rege a obra originária. Na prática, isto significa que, caso uma empresa desenvolva um software usando pedaços de códigos de um outro sistema livre que adota a GPL, esse software derivado (obra derivada), obrigatoriamente, tem que usar também a GPL como licença.

viii Se um sistema operacional é descrito como um conjunto de programas e “pacotes” que fazem um computador funcionar, o *kernel* é o mais importante “pacote” desse sistema. Isto porque o *kernel* realiza todas as operações mais básicas, permitindo que um usuário de computador execute outros programas. Ele é muitas vezes descrito como o “sistema nervoso” de um sistema operacional para computadores, dada a sua complexidade e funcionalidade.

ix O *Newsgroup* é uma ferramenta de discussões na Internet, onde usuários postam mensagens de texto (denominadas de “artigos”) em fóruns que são agrupados por assunto. Ao contrário das mensagens de correio eletrônico (*e-mail*), que são transmitidas quase que diretamente do remetente para o destinatário, os artigos postados nos *newsgroups* são retransmitidos através de uma extensa rede de servidores conectados entre si.

x Este modelo de desenvolvimento foi denominado inicialmente de *modelo Bazar* por Raymond (2004), apesar dele também ser chamado de *modelo acadêmico aberto* por Himanem (2001) e de *modelo de produção entre pares baseado no bem-comum* (*common-based peer production*) por Benkler (2002). Vale frisar que Himanem (2001) pensa ser ainda melhor comparar o modelo de código-fonte aberto ao modelo acadêmico de produção do conhecimento, pois segundo a sua visão, os cientistas também divulgam seu trabalho abertamente para que ele seja utilizado, testado e ampliado por outros. Em outras palavras, as pesquisas científicas teriam como princípio basilar a abertura de informações, dentro de um processo que estimula, continuamente, a auto-correção. Além do direito de uso, crítica e desenvolvimento - a ética acadêmica também obrigaria que as suas fontes devam ser mencionadas e que a nova “descoberta” não deve ser mantida em segredo, e sim publicada para a comunidade científica. Desta forma, “o motivo pelo qual o modelo original de fonte aberta dos hackers funciona tão efetivamente parece dever-se ao fato de que, além dos hackers o utilizarem para concretizar suas paixões e estarem motivados pelo mero reconhecimento, tal modelo se ajusta ao modelo aberto acadêmico ideal, que, historicamente, é o mais bem adaptado para criação de informações” (Himanen, 2001, p.70).

xi “Módulo” é o termo usado no Sistema de Controle de Versão (*Concurrent Versions System* - CVS) do GNOME para se referir a um diretório ligado ao desenvolvimento de cada software que compõem o sistema

GNOME, no repositório de versões. Todos os softwares que compõem ou estão ligados de alguma forma ao Projeto GNOME podem ser encontrados no link: <http://bugzilla.gnome.org/describecomponents.cgi>

xii Para controlar o desenvolvimento contínuo de novas versões, o projeto GNOME adota o padrão de catalogação de versões elaborado pelo Projeto do Kernel Linux. Segundo esse padrão, nas versões estáveis do software (de funcionamento seguro para qualquer usuário) o *Y* do número de uma versão *X.Y.Z* é par e o *Z* é zero (por exemplo 2.14.0), ao passo que, nas versões em desenvolvimento, destinada aos programadores, o valor de *Y* é ímpar e *Z* > 0 (por exemplo, a versão em desenvolvimento 2.13.1 é a versão 2.14.0 em processo de desenvolvimento). Por fim, o valor *X* só é aumentado em uma unidade quando uma alteração substancial é implementada na versão do software.

xiii As equipes do Projeto de Tradução do GNOME são ambientes onde se mais agregam a maioria dos colaboradores voluntários que não sabem ou não exercem a atividade de *hacking* – isto é, o desenvolvimento de códigos de programação. Cada uma das mais de 130 equipes de tradução tem um ou mais coordenador, e fica responsável pela tradução dos 66 softwares que compõem o *desktop* (que estão escritos em inglês) para uma determinada língua ou dialeto dos cinco continentes do mundo.

xiv Essa grupo, além de ser formado por profissionais voluntários da área de Marketing, conta também com a colaboração de jornalistas. Além das notas da nova versão que são mandadas para imprensa, essa equipe é responsável por projetos como, por exemplo, o jornal on-line da comunidade (o *Gnome Jornal*) e pesquisas de opiniões sobre as preferências tecnológicas dos usuários.

xv Depoimento do Bugmaster do GNOME, Luis Villa, colhido em entrevista realizada no dia 30/06/2006.