



Internet como media de produção de software livre e código aberto

Comunidades, Interacção e Ferramentas

Ana Raquel de Ponte Figueiras
Nº16400

Lisboa
Fevereiro de 2009

Universidade Nova de Lisboa
Faculdade de Ciências Sociais e Humanas
Mestrado em Novos Media e Práticas Web

**Internet como media de produção de software livre e código
aberto**
Comunidades, Interação e Ferramentas

Trabalho realizado no âmbito do seminário de
Ciberespaço, Media e Interação, leccionado pela
Professora Graça Rocha Simões

Ana Raquel de Ponte Figueiras
Nº16400

Lisboa
Fevereiro de 2009

Índice

Introdução	1
1. Free Software e Open Source	2
2. Comunidades no Ciberespaço	5
2.1 Identidade, géneros e conteúdo	6
2.2 Interacção	7
3. A especificidade das comunidades free software e open source	10
3.1 Estrutura da comunidade	13
3.2 Motivações	14
3.3 Ferramentas	17
3.2.1 Mailing Lists	18
3.2.2 IRC/Real-time chat systems	19
3.3.3 Sites e Wikis	20
3.3.4 Launchpad	21
3.4 O sucesso das comunidades open source e free software	23
Conclusão	25
Bibliografia	26
Anexos	28
Anexo A	29
Anexo B	41
Anexo C	43
Anexo D	44
Anexo E	45
Anexo F	46
Anexo G	47
Anexo H	48
Anexo I	49
Anexo J	50
Anexo K	51
Anexo L	52

Índice de Figuras

Figura 1 – Crescimento do <i>kernel</i> do <i>Linux</i>	11
Figura 2 – Estrutura de uma comunidade <i>open source</i>	13
Figura 3 – Motivações para contribuir em projectos Free Software e Open Source	15
Figura 4 – Representação do <i>Launchpad</i>	21

Introdução

«Sempre que trabalhávamos em algo na universidade ou em Stanford, na maior parte do tempo trabalhávamos em máquinas da Digital Equipment ou da Sun, principalmente da Sun. Sempre que chegava uma máquina da Sun, a primeira coisa que fazíamos era passar literalmente dias a fazer o download do software livre do GNU , a partir da Internet, a compilar e a instala-lo naquela máquina da Sun.»

Larry Augustin
(Revolution OS)

No início dos anos 90, uma *Sun Spark Station*, a máquina que Larry Augustin (co-Fundador e CEO da *SourceForge*, anteriormente chamada *VA Linux Systems*) e muitos outros programadores usavam, custava à volta de 7000 dólares e incluía, para além do hardware, o software proprietário da *Sun Microsystems*. Pelo facto destas máquinas serem demasiado caras a maior parte dos programadores não se podiam dar ao luxo de ter uma em sua casa e ficavam limitados aos computadores da universidade. *«Eu queria ter uma máquina Unix em casa, e descobri que poderia usar Linux num PC. Por cerca de 2000 dólares montei um sistema, que era duas vezes mais rápido que a Sun Spark Station»* conta Augustin no documentário *Revolution OS*.

Embora o modelo de desenvolvimento de software livre e de código aberto já fosse utilizado há muito tempo, principalmente no meio universitário, houve um verdadeiro “boom” com a popularização da Internet. Esta demonstrou ser um bom meio para a produção de software livre e de código aberto, visto que potencializa a ligação entre pessoas que têm esse interesse em comum.

Este trabalho tem o intuito de, em primeiro lugar, esclarecer o que é o software livre e o código aberto, demonstrar a importância que a Internet têm na formação de comunidades que se dedicam a este modelo de produção de software e descrever as ferramentas essenciais para a sua produção. Deste modo pretende-se mostrar as formas de interacção entre os membros das comunidades de *free software* e de *open source*, no Ciberespaço.

1. Free Software e Open Source

O termo *Free Software* foi popularizado por Richard Matthew Stallman, um programador e defensor da partilha do código fonte. Em 1984, Stallman deixou o laboratório de Inteligência Artificial do *Massachusetts Institute of Technology* para fundar a *Free Software Foundation*¹ e criar um sistema operativo completamente livre, com o intuito de recuperar o espírito de partilha e alteração livre do código, que existia antes de existirem as licenças proprietárias. A esse projecto chamou *GNU*². Stallman lançou assim as bases para o movimento ideológico e político *free software*.

Sendo o código fonte de um software uma forma de conhecimento científico, e como os cientistas publicam os seus trabalhos para que outros cientistas possam investigar através dos seus resultados, os cientistas de computação devem publicar o seu código fonte para que outros possam continuar a inovar na área da computação (Dempsey *et al*, 1999:2).

Com o objectivo de proteger o código fonte aberto do seu projecto, para que este fosse eternamente livre e que nenhuma empresa o usasse em software proprietário, Stallman criou uma licença: a *GNU General Public License (GPL)*³. Esta licença de *copy left*⁴, diz que o código pode ser copiado e modificado sem restrições, e que tanto as cópias como os trabalhos derivados (i.e., versões modificadas) têm de ser distribuídos sob a mesma licença que o original, sem restrições adicionais (Fogel, 2005:7). Ou seja, o free software rege-se sobre uma determinada base filosófica, consubstanciada em quatro liberdades em particular: a liberdade de executar o software para qualquer uso, de estudar o funcionamento de um programa e de adaptá-lo, de redistribuir cópias, de melhorar o programa e de tornar as modificações públicas para que a comunidade beneficie da melhoria.

No entanto, à medida que o *free software* despertava o interesse das empresas surgiu um problema com o termo *free software*, visto que a definição de *free* em inglês é livre ou gratuito. Segundo Richard Stallman «*software livre refere-se não ao preço mas sim à liberdade, por isso pense em liberdade de expressão e não em cerveja grátis*» (Revolution OS), mas muitos programadores continuavam a achar que o termo *free software* tinha uma conotação negativa. Em 1998, um grupo de programadores

1 <http://www.fsf.org>

2 GNU é um acrónimo recursivo que significa “GNU's Not Unix”

3 Ver anexo A

4 Copyleft é um trocadilho com a palavra Copyright. Este termo descreve um tipo de licença que, contrariamente ao Copyright, remove as restrições de distribuição de cópias e versões modificadas por outras pessoas e que requer que essas mesmas permissões sejam mantidas nessas cópias e versões modificadas.

resolveram adoptar o termo *open source*, formando assim um grupo chamado *The Open Source Initiative (OSI)*.

«*The Open Source Initiative é um programa de marketing para software livre. É um impulso para o free software em terreno sólido do pragmatismo em vez de um ataque ideológico agressivo. A atitude ganhadora não mudou, a atitude perdedora e simbolismo mudaram (...) Porque não chamamos free software, como fazíamos anteriormente? Uma razão directa é o facto do termo free software ser facilmente mal interpretado de maneiras de levam ao conflito (...) Estamos a tentar impulsionar o nosso conceito para o mundo corporativo. Temos um produto ganhador, mas o nosso posicionamento no passado era terrível. O termo free software tem sido mal interpretado pelos homens de negócios, que confundem o desejo de partilhar com anti-comercialismo, ou pior, como roubo.*» (qtd. in Fogel, 2005:12 a 13)

Segundo Karl Fogel, *developer* da plataforma de colaboração *Launchpad* e autor do livro *Producing open source software*, o que a *Open Source Initiative* criou foi um vocabulário novo para se poder falar de software livre como uma metodologia de desenvolvimento de software e como uma estratégia de negócio, em detrimento de uma cruzada moral. Ou seja, tanto o movimento *free software* como o movimento *open source* defendem que o código deve ser acessível e que o software pode ser comercializado ou distribuído gratuitamente e a diferença entre ambos os movimentos está muito mais marcada na questão ideológica. No entanto existe também uma diferença que vai para além da questão linguística e da questão ideológica: algumas das licenças admitidas pelo movimento *open source* não são admitidas pelo movimento *free software*, pois são consideradas pouco permissivas. A *Open Source Initiative* adoptou um conjunto de critérios aos quais chamaram *The Open Source Definition*⁵. Bruce Perens adaptou o seu documento *Debian Free Software Guidelines* para deixarem de ser específicas para o *Debian* e passaram a dizer respeito ao *open source* em geral. A *Open Source Definition* especifica uma serie de licenças que podem ser utilizadas pelo software *open source*: *GNU GPL*, *Berkeley Unix Project (BSD)*, *the X Consortium*, e outras.

Apesar de todas as diferenças que existem entre software livre e software com código aberto, ambos acentam num modelo de produção que inclui três princípios fundamentais: a permeabilidade entre os papéis do produtor e do consumidor, a partilha do conhecimento e a criação através de um processo de *patchwork* resultante dos contributos de várias pessoas de várias partes do planeta. Este processo de *patchwork* não representa um trabalho eternamente inacabado mas sim em permanente

5 Ver anexo B

Internet como meio de produção de Código Aberto

desenvolvimento graças ao contributos dos membros da comunidade, sejam estes em forma código, traduções ou suporte.

2. Comunidades no Ciberespaço

Para falar das comunidades que existem no ciberespaço (comunidades online) é importante definir o que é uma comunidade. Segundo a Infopédia, “comunidade” vem do latim *Communitate*, sendo «qualquer grupo social cujos membros vivem numa determinada área, sob um governo comum e partilhando uma herança cultural e histórica, sociedade». No entanto quando falamos de comunidades no ciberespaço esta definição pode parecer redutora, pois implica que os indivíduos têm de estar num determinado local para pertencer a uma determinada comunidade. Então torna-se também essencial definir o que é ciberespaço para conseguir definir esta comunidade tão particular.

Segundo Pierre Lévy, o ciberespaço «*constitui um campo vasto, aberto, ainda parcialmente indeterminado, que não deve ser reduzido a um único dos seus componentes.*» (Lévy, 1997: 153). Mais tarde o autor acrescenta, na sua obra *A inteligência colectiva : Para uma antropologia do ciberespaço* que o ciberespaço é o novo meio de comunicação que emerge da interligação mundial dos computadores e torna-se num espaço de comunicação, de sociabilidade, de organização e de transação, como também é um mercado da informação e de conhecimento. (2000).

Bruce Sterling, por sua vez, afirma que hoje em dia faz sentido falar em ciberespaço como um lugar por si só. (1993:11). Assim sendo, considerando o ciberespaço como um lugar, e não só como a ligação mediada pela rede, a definição de comunidade no ciberespaço começa a fazer mais sentido.

Torna-se também necessário distinguir estas comunidades que existem no ciberespaço das comunidades virtuais. Apesar de ambos os tipos de comunidades serem grupos de pessoas que se interligam entre si através de redes informáticas diferem nas suas raízes. Enquanto as comunidades *online* referem-se a recriações na Internet de comunidades já existentes no mundo físico, as comunidades virtuais correspondem a associações formadas na Internet sem ligação a um espaço físico preexistente. (Cardoso, 2003:85).

Como em qualquer comunidade, as comunidades *online* nascem porque várias pessoas têm um interesse em comum, e também alguns interesses que entram em conflito. (Agré, 1995). Estas pessoas estão ligadas umas às outras graças aos fluxos de comunicação contínuos que existem entre elas, fluxos que se mantêm mesmo quando a distância as separa.

«*Cyberspace is already the home of thousands of groups of people who meet to share information, discuss mutual interests, play games, and carry out business (...) Community is now conceptualized not*

in terms of physical proximity but in terms of social networks.» (Smith e Kollock, 1999:16).

O que distingue as comunidades que existem no ciberespaço das comunidades do mundo físico é o media que utilizam para se reunirem, porém mantém-se o elo principal entre os membros das comunidades: a comunicação. Esta comunicação é bidireccional, tem um *timing* flexível e este ambiente comunicacional cria um sentido de lugar. (Downes e McMillan, 2000:21).

2.1 Identidade, géneros e conteúdo

A questão da identidade também tem de ser considerada quando falamos de comunidades que surgem no ciberespaço. A identidade é um pilar da interacção social, pois em todas as interacções que temos com os outros sentimos necessidade de ter noção de com quem estamos a interagir.

Nas interacções cara a cara existem pistas que nos ajudam a descodificar a identidade da pessoa com a qual interagimos: as roupas, as expressões corporais, os indicadores de *status*, etc. Este reconhecimento da identidade do outro é fulcral para que saibamos o que esperar e como reagir. «*Our ability to identify others also allows us to hold individuals accountable for their actions.*» (Smith e Kollock, 1999:9).

Porém, no ciberespaço torna-se complicado descodificar a identidade dos indivíduos e das próprias comunidades. A melhor forma de reconhecer a identidade de uma comunidade é observar os géneros de comunicação que adoptam.

Os géneros são um tipo distinto de acção comunicativa caracterizado por um propósito comunicativo socialmente reconhecido e por aspectos formais comuns. «*Each genre implies a particular sort of audience and a particular sort of activity (Bazerman 1988). Who the audience actually is and what its members are actually doing are, of course, fairly difficult empirical matters.*» (Agre, 1998). Segundo Bakhtin os textos e os discursos só podem ser percebidos através da análise dos géneros e o processo criativo requer um uso efectivo destes. Este autor defendia que os géneros eram criados através de um processo histórico no qual os conceitos, o seu uso e o seu contexto prático co-evoluem (1987).

Cada comunidade adopta os géneros que melhor se adaptam aos diferentes propósitos da comunidade e à própria vida dos seus membros. Contudo, cada género não é exclusivo de uma só

comunidade, apesar da forma como cada comunidade utiliza um determinado género não ser igual. «*Of course, a given genre might be addressed to several different purposes simultaneously, or even to several different communities, but it stands to reason that a genre cannot be too many things to too many communities without diluting its usefulness for any one of them.*» (Agre, 1998). O facto de uma comunidade adoptar um determinado género para comunicar permite-nos entender como a comunidade se organiza e é muito importante para podermos entender a interacção no ciberespaço.

Na *web* existem múltiplos géneros: *e-mail*, *mailling list*, *chat*, *faq* (perguntas frequentes), etc. Muitos destes géneros são utilizados pela comunidade *free software* e *open source*, e serão estudados neste trabalho precisamente para perceber como funciona esta comunidade e como ela interage no ciberespaço.

Sally McMillan, uma autora que explora as formas como a comunicação interactiva leva à formação de comunidades virtuais, afirma que o conteúdo produzido pelas comunidades representa 57.3% do conteúdo *online*. «*Community Content sites can exist with minimal funding from community-minded organizations and with the donated time, efforts, and expertise of volunteers. The content of such sites is an ever-changing montage of ideas, information, and intelligence created and shared by the participants.*» (McMillan, 1998). Segundo a autora, os sites que reúnem conteúdo produzido por uma comunidade incorporam mais elementos de interactividade e os produtores desse conteúdo não dão muito valor à propriedade intelectual.

Os conteúdos produzidos pelas comunidades expressão a sua voz colectiva e o conteúdo produzido pelas comunidades *open source* e *free software* traduz-se muitas vezes em programas e documentação essenciais para o funcionamento de variadas ferramentas utilizadas pelos próprios membros da comunidade. Assim sendo, este conteúdo não só é produzido pela comunidade como também é produzido para a comunidade.

2.2 Interacção

A interacção é um conceito das ciências físicas que foi importado e modificado para as ciências sociais por Goffman. Segundo este autor a interacção é a «*influência recíproca dos indivíduos sobre as acções uns dos outros numa situação de presença física imediata.*» (Goffman, 1959:26). Contudo a

definição de Goffman torna-se redutora pois no ciberespaço também existem formas interação, que não são face-a-face, mas podem ser por exemplo face-a-dispositivo. Por isso, uma definição mais adequada seria que a interação é um processo de influência recíproca de indivíduos ou agentes em presença sobre as acções uns dos outros, numa sucessão contínua de acontecimentos interligados e ligados a uma estrutura espaço-temporal.

O tipo de interacção que ocorre no ciberespaço é muito diferente do tipo de interacção face-a-face, e que no entanto também dá aos seus intervenientes a sensação de co-presença, ou seja a sensação de estar junto do outro. De acordo com Shanyang Zhao as situações de co-presença podem ser organizadas de acordo com a localização do outro e se este está presente em pessoa ou via simulação. Portanto existe a co-presença física/corporal (o face-a-face), co-presença virtual (simulação física), telepresença física/corporal (face-dispositivo) e telepresença virtual/simulação digital (através de agentes). Tomando esta separação de Zhao, o tipo de presença que estará mais associado às comunidades *open source* e *free software* será a telepresença física/corporal.

A partir do momento em que foram criados determinados meios de interacção na *web* como o *e-mail*, o *chat* e os sistemas de conferência, as pessoas começaram a formar milhares de grupos para discutir uma variedade de temas, para jogar, para se entreter, e até para trabalhar em vários projectos colectivos. «*Each online communication system structures interaction in a particular way, in some cases with dramatic effect on the types of social organizations that emerge from people using them.*» (Smith e Kollock, 1999:3).

«*Interactivity is the relationship between two or more people who, in a given situation, mutually adapt their behavior and actions to each other.*» (Jensen, 1998:188). Segundo Straubhaar e LaRose a expressão interactividade deve ser usada para caracterizar situações em que existe um *feedback* em tempo real dos receptores da informação (1996: 12). A interactividade é apenas mais uma das dimensões da interacção, para além desta também existem outras dimensões igualmente importantes como a funcionalidade e a usabilidade.

As comunidades *opensource* e *free software* são um bom exemplo de comunidade que utiliza o ciberespaço como meio de interacção entre os seus membros. Estas não podem ser chamadas de comunidades virtuais pois são comunidades que já existiam no mundo físico e que alguns dos seus membros não se conhecem apenas através da rede. Para além disso, estas são comunidades que muitas vezes passam do ciberespaço para o mundo físico para reuniões e encontros face-a-face. Por exemplo,

Internet como meio de produção de Código Aberto

alguns membros da comunidade Ubuntu, um sistema operativos baseado em Linux, disponibilizam-se para instalar a distribuição e a ajudar a esclarecer dúvidas que as pessoas possam ter face-a-face⁶, para além das reuniões que as *LoCo-teams*⁷ têm todos os meses.

⁶ Anexo C

⁷ LoCo é uma abreviatura de Local Community (comunidade local). Estas sub-comunidades dedicadas a cada país ajudam com suporte, esclarecem dúvidas e muitas vezes até reúnem-se cara-a-cara.

3. A especificidade das comunidades free software e open source

As comunidades *free software* e *open source* destacam-se no ciberespaço por serem comunidades cujo conteúdo que produzem é maioritariamente código, código este que é necessário para o funcionamento do mais variado software. Estas comunidades são guiadas pelo princípio de que o código deve ser aberto e disponível para que qualquer pessoa o possa utilizar, alterar e distribuir. Segundo Vincent Freeh, Greg Madey e Renee Tynan o *open source* e o *free software* são culturas de desenvolvimento de software que juntam milhares de programadores que produzem, partilham e suportam o seu software voluntariamente sem receberem uma compensação monetária. (2002:1). Porém existem programadores pagos para trabalharem em projectos *open source* e *free software*, sem no entanto perderem o interesse, pois muitas vezes são pessoas que já trabalhavam nesse projecto voluntariamente e que depois são contratadas. «*Open-source developers are volunteers, self-selected for both interest and ability to contribute to the projects they work on (and this remains generally true even when they are being paid a salary to hack open source.)*» (Raymond, 2001:25).

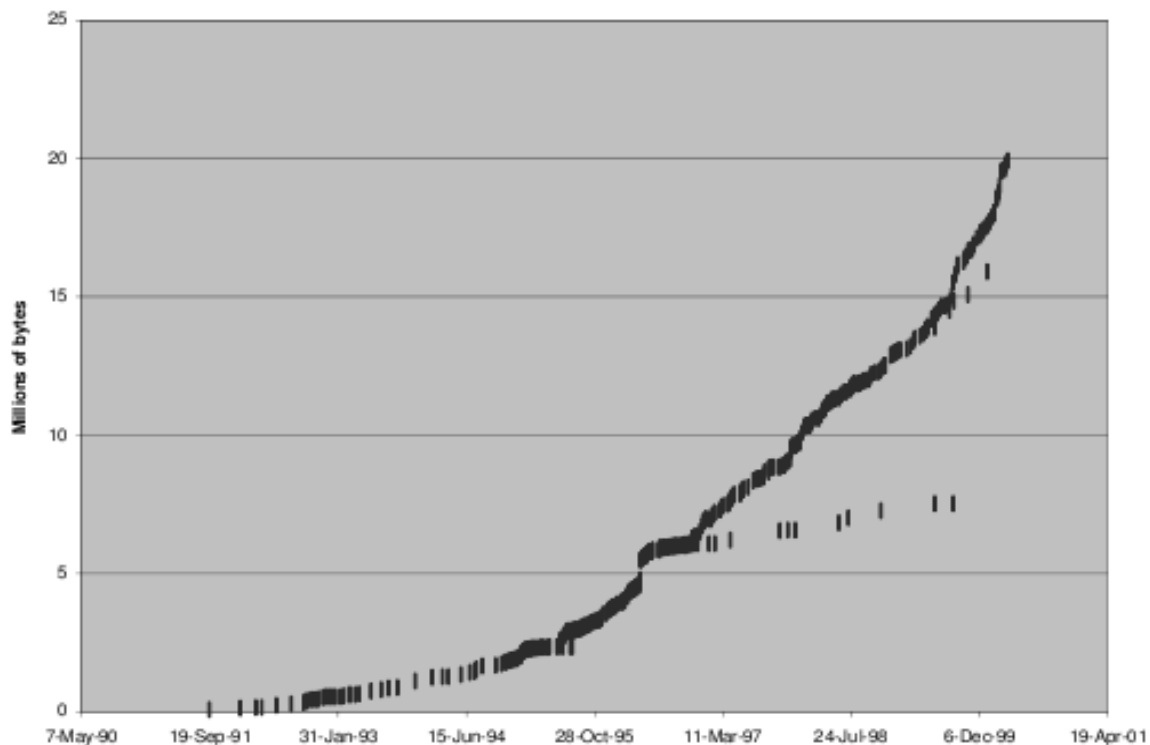
O modelo de desenvolvimento do software livre e do código aberto garante a validade do código através da *peer review*, ou seja, se alguém fizer uma alteração no código que prejudica o funcionamento do programa outra pessoa irá encontrar esse erro e corrigi-lo. «*Nonetheless, the guiding principle for open source software is that, by sharing source code, developers cooperate under a model of rigorous peer-review and take advantage of “parallel debugging” that leads to innovation and rapid advancement in developing and evolving software products.*» (Dempsey et al, 1999:2). Este ciclo de revisão de código garante um desenvolvimento não só rápido, mas que também garante a qualidade do software. Segundo Eric Raymond, autor de *The Cathedral and the Bazaar*, obra que descreve o modelo de desenvolvimento do software de código aberto, afirma que estes projectos só se extinguem quando os *developers* perdem o interesse. (Raymond, 2001:25). Contudo, o modelo de desenvolvimento de código aberto não é apenas um modelo de produção de software, mas também um modelo que produz interacção, aprendizagem e conhecimento. É este modelo que organiza a comunidade e as suas relações com as outras comunidades.

A proliferação do acesso à Internet permitiu que esta cultura de partilha, que já existia à muito tempo no meio académico, se expandisse. «*Internet connectivity has enabled the open-source notion of cooperative, peer-reviewed software development to be deployed on a global scale. Perhaps the*

most influential open-source project to-date has been and continues to be the Linux operating system.»
(Dempsey et al, 1999:3).

O *Linux* começou a ser desenvolvido em 1991 por Linus Torvalds, do Departamento de Ciência da Computação da Universidade de Helsínquia. Torvalds queria criar um sistema operativo baseado em Unix para o seu computador e depois de ter criado uma versão estável do *kernel*⁸ resolveu disponibilizar o código para que outras pessoas ajudassem no projecto. No início de 1994 o projecto já tinha evoluído imenso e a versão 1.0 foi oficialmente lançada gratuitamente e disponível para que mais alterações fossem feitas pelos seus utilizadores.

Figura 1 – Crescimento do *kernel* do *Linux*



«I don't think it's a coincidence that the gestation period of Linux coincided with the birth of the World Wide Web, and that Linux left its infancy during the same period in 1993–1994 that saw the takeoff of the ISP industry and the explosion of mainstream

8 Núcleo do sistema operativo

interest in the Internet. Linus was the first person who learned how to play by the new rules that pervasive Internet access made possible.» (Raymond, 2001:21).

Tal como Raymond também Torvalds e outros autores acreditam que a Internet é um meio que facilita a colaboração. No começo a Internet serviu de meio para Torvalds receber algumas sugestões, conselhos e desenvolver o *kernel*, depois à medida que as pessoas começavam a usar o *Linux* começaram a enviar *Bug⁹ Reports* e até a corrigir alguns erros. Com o desenvolvimento cada vez maior do *Linux* os programadores começaram a criar novos sistemas operativos utilizando o *kernel Linux* e assim ajudando a populariza-lo.

Segundo Torvalds o *Linux* foi desenvolvido com sucesso porque era um projecto interessante e que outros programas menos interessantes como um processador de texto não serão tão facilmente desenvolvidos pela comunidade. Contudo o *GNU Emacs¹⁰* é um exemplo de que mesmo um editor de texto pode ser um projecto aliciante para centenas de voluntários.

Apesar do sucesso de projectos *free software* como o *Linux*, ainda há quem encare com algum cepticismo este modelo de produção de software. Duvidam que a abordagem do *open source* e do *free software* poderá algum dia produzir software robusto, exigido pelas massas e pelas grandes corporações. (Dempsey et al, 1999:2). Entretanto já existem alguns casos de sucesso que fazem com que os sépticos pensem duas vezes: *Firefox, Sun Microsystems, MySQL, Apache, IBM, etc.*

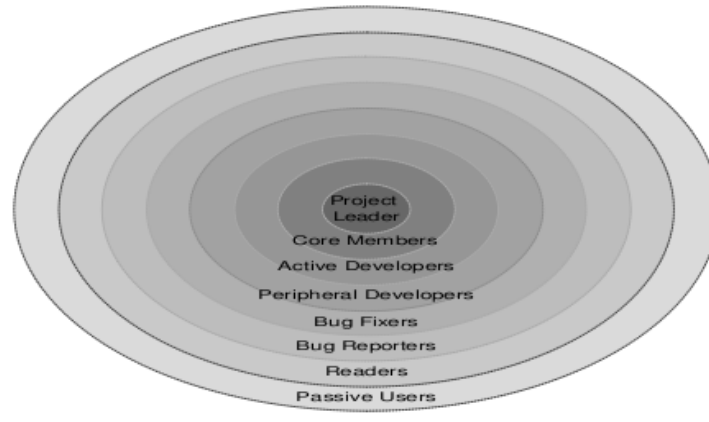
Segundo Eric Raymond o ponto forte do free software e do open source é o seu modelo de produção particular: o bazar. Em *The Cathedral and the Bazaar* Raymond utiliza a metáfora da catedral e do bazar para comparar o software proprietário com o software livre: a catedral é um local fechado e silencioso onde poucos trabalham sob um comando central e o bazar é um espaço aberto e ruidoso no qual muitos trabalham ao mesmo tempo de forma descentralizada. Na catedral o trabalho é feito por uma equipa fechada, na qual o projecto é desenvolvido por um pequeno grupo. Enquanto no bazar, depois de divulgada a primeira versão do programa, uma multidão de programadores começará a testar e a corrigir os erros encontrados voluntariamente. Como conta com uma comunidade ampla, composta por pessoas de diferentes lugares e como vivências diferentes, o projecto de software livre terá estatisticamente muito mais probabilidade de ter os seus *bugs* corrigidos, alcançando rapidamente uma estabilidade de funcionamento.

9 Erro ou falha na execução de um programa, prejudicando ou inviabilizando o seu funcionamento

10 Editor de texto do sistema operativo GNU

3.1 Estrutura da comunidade

Figura 2 – Estrutura de uma comunidade *open source*



Apesar de não haver uma estrutura hierárquica obrigatória nas comunidades *open source* e *free software* é óbvio que estas comunidades não são sistemas anárquicos. A influência que cada um dos membros têm sobre o projecto e sobre a comunidade são diferentes, dependendo do papel que elas desempenham. No centro, como espelha a Figura 2, que mostra uma estrutura comum nas comunidades *open source* e *software livre*, está o *Project Leader* (líder do projecto) que é a pessoa que tem maior influência no projecto, tendo um papel de regulador. Apesar de todos os membros da comunidade serem importantes para o seu funcionamento, um *Core Member* tem mais influência que um *Active Developer* (programador activo) e os *Passive Users* (utilizadores passivos) são os que têm menos influência no desenvolvimento do projecto. Porém os utilizadores só pela sua existência já desempenham um papel muito importante na comunidade, pois a sua existência contribui socialmente e psicologicamente para atrair e motivar outros membros mais activos. O papel que cada pessoa desempenha na comunidade não é fixo, porque para ter um papel mais importante na comunidade é apenas preciso contribuir mais. «*The roles and their associated influences in OSS communities are not associated with any attributes (such as age, title, etc.) of a member; instead, they can only be earned through contributions to the community.*» (Nakakoji et al).

Segundo os investigadores Kumiyo Nakakoji, Yasuhiro Yamamoto, Yoshiyuki Nishinaka¹, Kouichi

Kishida e Yunwen Ye, autores de *Evolution Patterns of Open-Source Software Systems and Communities*, é extremamente importante manter o equilíbrio dos papéis desempenhados pelos diferentes membros da comunidade, pois se existirem apenas utilizadores passivos o projecto nunca evoluirá e se todos os membros da comunidade estão encarregues de a gerir o projecto, este torna-se insustentável.

3.2 Motivações

«Intrinsic motivation is defined as the doing of an activity for its inherent satisfactions rather than for some separable consequence. When intrinsically motivated, a person is moved to act for the fun or challenge entailed rather than because of external prods, pressures, or rewards .» (Ryan e Deci, 2000:56)

Para perceber o funcionamento das comunidades *open source* e *free software* é necessário conhecer as suas motivações. «*Many are puzzled by what appears to be irrational and altruistic behavior by movement participants: giving code away, revealing proprietary information, and helping strangers solve their technical problems.*» (Lakhani e Wolf, 2005:3).

Existem muitas possíveis motivações para contribuir em projectos *free/open source* e para fazer parte destas comunidades, porém existem três características estruturais que são comuns a vários casos de comunidades *free software* e *open source* de sucesso, e que são características que fazem as comunidades *online* funcionarem. Estas características são: interacção continua, persistência da identidade e conhecimento de interacções anteriores (Smith e Kollock, 1999). Esta última é particularmente importante pois, se uma pessoa não tiver boas memórias de interacções anteriores, terá dificuldade em criar e manter uma comunidade cooperativa *online*.

Tendo em conta que as comunidades *open source* e *free software* reúnem as características de um movimento social podemos tentar entender as suas motivações através de modelos que tentam explicar as motivações dos movimentos sociais. Diversos autores das ciências sociais definem um movimento social como uma acção com uma intenção explícita de cooperação, colectiva e concentrada. As pessoas que participam nestes movimentos sociais defendem com convicção uma determinada causa.

Klandermans apresenta um modelo que tenta explicar a participação em movimentos sociais

Internet como meio de produção de Código Aberto

(1997). Segundo o seu modelo, a motivação depende dos custos e dos benefícios, dividindo-se três tipos de motivos: motivos colectivos, sociais e de recompensa. Os motivos colectivos estão relacionados com os objectivos do movimento social em questão, os motivos sociais estão relacionados com as expectativas dos outros (família, amigos, etc.) e os motivos de recompensa, como o próprio nome indica, estão relacionados com as possíveis retribuições que se pode obter. Klandermans acredita que, no seu conjunto, estas três classes de motivos determinam a vontade que uma pessoa sente em participar num determinado movimento social. Contudo existem outros teóricos que afirmam que as pessoas não aderem a movimentos sociais só por causa dos custos e benefícios, mas também por causa de aspectos de identificação colectiva. (Simon et al., 1998). Ou seja, as pessoas também sentem que fazem parte de um movimento social porque se identificam com as normas, ideologias e os standards desses grupo.

Existem vários motivos para pertencer às comunidades *open source* e *free software* e muitos destes motivos encaixam-se nas categorias supracitadas. Karim Lakhani e Robert Wolf elaboraram um estudo, sobre as motivações das pessoas que participam em projectos *free software* e *open source*, intitulado *Why Hackers Do What They Do: Understanding Motivation and Effort in Free/Open Source Software Projects*. Através de um inquérito realizado a 684 programadores, que contribuem activamente em projectos *open source* e *free software*, os dois investigadores tentaram perceber quais são as motivações que moviam esses programadores (Lakhani e Wolf, 2005).

Figura 3 – Motivações para contribuir em projectos Free Software e Open Source

<i>Motivation</i>	% of respondents indicating up to 3 statements that best reflect their reasons to contribute (%)	% volunteer contributors	% paid contributor	Significant difference (t statistic/p value)
<i>Enjoyment based Intrinsic Motivation</i>				
Code for project is intellectually stimulating to write	44.9	46.1	43.1	n.s.
Like working with this development team	20.3	21.5	18.5	n.s.
<i>Economic/Extrinsic based Motivations</i>				
Improve programming skills	41.3	45.8	33.2	3.56(p=0.0004)
Code needed for user need (work and/or non-work)*	58.7	-	-	-
- Work need only	33.8	19.3	55.7	10.53 (p=0.0000)
- Non-work need	29.7	37.0	18.9	5.16(p=0.0000)
Enhance professional status	17.5	13.9	22.8	3.01(p=0.0000)
<i>Obligation/Community based Intrinsic Motivations</i>				
Believe that source code should be open	33.1	34.8	30.6	n.s.
Feel personal obligation to contribute because use F/OSS	28.6	29.6	26.9	n.s.
Dislike proprietary software and want to defeat them	11.3	11.5	11.1	n.s.
Enhance reputation in F/OSS community	11.0	12.0	9.5	n.s.

Segundo a tabela de Lakhani e Wolf acima apresentada, tabela esta que representa as três motivações que cada programador apontou como razões de participar em projectos *free software* e *open source*, 58.7 por cento dos programadores afirmam que o que os motiva é o facto daquele código ser necessário para os utilizadores daquele software. «*In this case a possible motivation is need, that is, one may produce and contribute a public good for the simple reason that a person or the groups as a whole has a need for it.*» (Smith e Kollock, 1999). Esta motivação está relacionada com a ligação que o programador tem com a comunidade a que pertence, sente necessidade de produzir código para o bem da comunidade. «*In this case one contributes to the group because that is what is best for the group – individual and collective outcomes are thus merged and there is no social dilemma. However, complete devotion to a group is rare.*» (Smith e Kollock, 1999)

O estímulo intelectual que têm ao escrever código é uma das principais razões que 44.9 por cento dos programadores afirma ser a motivação para contribuírem em projectos *open source* e *free software*. «*We find in contrast, that enjoyment-based intrinsic motivation, namely how creative a person feels when working on the project, is the strongest and most pervasive driver.*» (Lakhani e Wolf, 2005:3). O facto dos programadores participarem nestes projectos porque se interessam e porque estes são estimulantes, divertidos, faz com que trabalhem com muito mais gosto, pois não há pressões. Os programadores também confessam que os projectos *open source* e *free software* dão-lhes mais liberdade para serem criativos.

«*Open source software development teams, are generally comprised of volunteers working not for monetary return, but for the enjoyment and pride of being part of a successful virtual software development project.*» (Freeh, Madey e Tynan, 2002:2). No entanto 43.1 por cento dos programadores inquiridos, e que são pagos para participar em projectos *free software* e *open source*, também apontam esta motivação como a razão que os faz trabalhar nestes projectos.

A terceira razão que os programadores apontam para justificar o que fazem é uma motivação de recompensa: melhorar as suas aptidões de programação. Nos projectos *open source* e *free software* os programadores recebem um importante contributo para a sua formação como programadores através da *peer review*, pois através deste contributo da comunidade na correcção de erros, no desenvolvimento do projecto, na criação de documentação, os programadores percebem melhor as suas falhas. (Raymond 1999). Esta razão pode gerar outro benefício, para além do benefício de se tornar um bom

programador: uma subida na carreira. Os programadores muitas vezes mostram as suas qualidades como programadores aos potenciais empregadores contribuindo activamente em projectos *open source* e *free software*. «*Similarly, firms looking for a particular skill in the labor market can easily find qualified programmers by examining code contributions in the F/OSS domain.*» (Lakhani e Wolf, 2005:7). Na verdade, 17,5 por cento dos programadores que participaram neste estudo, confessaram que esta era uma das principais razões para contribuírem nestes projectos.

Eric Raymond afirma que a principal razão pela qual os programadores participam em projectos *free software* e *open source* não é tanto por altruísmo, mas mais para satisfazer o ego. «*One may call their motivation “altruistic”, but this ignores the fact that altruism is itself a form of ego satisfaction for the altruist.*» (Raymond, 2001:22). Contribuir regularmente para a comunidade, com contribuições de qualidade e úteis, ajudam o programador a acreditar que tem um grande impacto na comunidade e isso aumenta a sua confiança. Nesse caso, tomando este altruísmo como uma forma de satisfação do ego pode-se dizer que também esta razão é uma motivação de recompensa. «*Both the fetchmail and Linux kernel projects show that by properly rewarding the egos of many other hackers, a strong developer/coordinator can use the Internet to capture the benefits of having lots of co-developers without having a project collapse into a chaotic mess.*» (Raymond, 2001:23). A reputação de um programador dentro da comunidade é muito importante, não só para que o programador seja respeitado dentro da comunidade mas também para que possa tornar-se líder de um projecto. «*High quality information, impressive technical details in one's answers, a willingness to help others, and elegant writing can all work to increase one's prestige in the community.*» (Smith e Kollock, 1999). Contudo apenas 11 por cento dos programadores confessaram que aumentar a sua reputação na comunidade *free software* e *open source* era uma das principais motivações para participarem neste tipo de projectos.

Uma motivação de recompensa que Lakhani e Wolf não equacionaram foi a motivação de contribuir na expectativa que, ao fazê-lo, quando precisarem também iram usufruir do código produzido pelos outros. De acordo com Smith e Kollock esta seria uma motivação de reciprocidade antecipada (1999).

3.3 Ferramentas

Quando os projectos *free software* e *open source* são pequenos e envolvem poucas pessoas a

comunicação flui facilmente entre todos os intervenientes. Porém, quando o projecto cresce e à volta deste projecto está uma comunidade com centenas de pessoas, torna-se complicado coordenar os participantes. «*If good free software project management is about making everyone feel like they're working together in the same room, the obvious question is: what happens when everyone in a crowded room tries to talk at once?*» (Fogel, 2005:45).

De acordo com Karl Fogel os projectos precisam de tecnologias que suportam a captura selectiva e a integração de informação para poderem ter um bom apoio da comunidade. As *mailing lists*, os *real time chat systems* como o *IRC*, os sites, os *wikis* e as plataformas de *hosting* e colaboração (*Launchpad* e *Sourceforge*) não são apenas meios de comunicação – são meios de grupo, que sustentam e suportam uma interacção de muitos para muitos (Smith e Kollock, 1999:3).

3.3.1 Mailing Lists

«*Mailing lists are the bread and butter of project communications.*» (Fogel, 2005:48)

O *e-mail* e as listas de discussão são a mais antiga e a mais popular forma de interacção na Internet, sendo através destes que os membros de uma comunidade *free software* ou *open source* comunicam mais activamente. Para Fogel as *mailing lists* são mais do que um meio de comunicação, são um *medium of record*, pois a comunicação fica registada.

O *e-mail* permite a um indivíduo enviar uma mensagem directamente para outra pessoa, numa interacção assíncrona de um-para-um. Porém a *mailing list* é a prova de que a interacção através do género *e-mail* não tem de ser apenas entre duas pessoas. Numa *mailing list* as mensagens de *e-mail* enviadas para o endereço do grupo são enviadas para todos os subscritores da lista. «*This may be the most common form of group interaction on the Internet, and a number of lists contain thousands or tens of thousands of members.*» (Smith e Kollock, 1999:5).

Segundo Smith e Kollock as *mailing lists* têm algumas qualidades importantes que as distinguem das outras formas de comunicação *online*. Uma dessas qualidades é o controlo significativo que o proprietário da lista tem sobre quem pode ou não contribuir para o seu grupo. «*List owners can personally review all requests to be added to a list, can forbid anyone from contributing to the list if they are not on the list themselves, and even censor specific messages that they do not want broadcast*

to the list as a whole.» (Smith e Kollock, 1999:5). Algumas *mailing lists* têm menos supervisão, pois a regulação destas é complicada de se fazer, e as pessoas podem as subscrever livremente. Ainda assim, mesmo as listas mais abertas podem ser moderadamente controladas pelos seus proprietários. A maioria das *mailing lists* funciona como “ditaduras benignas”, pois o proprietário exerce alguns limites sobre o grupo, fazendo com que a *mailing list* seja ordenada e centrada na sua actividade.

Existem muitos benefícios para a utilização de meios de comunicação *online* que permitem uma interacção assíncrona. Um desses benefícios é a capacidade de interagir sem ter de reunir todos os participantes num determinado momento e assim, pessoa que estejam em fusos horários diferentes podem trocar mensagens e manter discussões (Smith e Kollock, 1999:5).

A maior parte das discussões que ocorrem nas *mailing lists* não estão directamente relacionadas com a produção de código mas sim com novas *features* para o projecto, com dúvidas relacionadas com o seu funcionamento e alterações que melhorem a performance e a qualidade do programa. Em projectos de grandes dimensões é mais vantajoso ter várias *mailing lists*, com diferentes tópicos. Por exemplo no caso do *Ubuntu* existem várias *mailing lists*: *ubuntu-security-announce* (para anunciar updates), *ubuntu-announce* (para anunciar novas versões), *ubuntu-news* (notícias semanais), *ubuntu-users* (para discussões sobre suporte e sugestões), *ubuntu-devel* (para os programadores), *ubuntu-devel-discuss* (para discussões da comunidade e sugestões de novos desenvolvimentos), *package upload and automatic notification* (arquivo das notificações das versões) e *ubuntu-motu* (sobre *packaging* e desenvolvimento), etc. Para além de todas estas *mailing lists*, as *LoCo-teams* do *Ubuntu* também têm as suas *mailing lists*.

O *e-mail* pode não ser um substituto das reuniões, mas é uma ferramenta que explora as potencialidades do trabalho colectivo e media a interacção até que a reunião face-a-face se realize (Agre, 1998).

3.3.2 IRC/Real-time chat systems

Os sistemas de *Real-time chat*, como por exemplo o *IRC (Internet Relay Chat)*, são meios de comunicação *online* muito utilizados pelas comunidade *free software* e *open source* pois suportam comunicação síncrona. O *IRC* permitem que várias pessoas se juntem no mesmo canal e mantenham uma conversação em tempo real, facilitando assim a discussão. «*Most chat systems support a great*

number of "channels" dedicated to a vast array of subjects and interests.» (Smith e Kollock, 1999:6). Este meio também permite uma regulação ou moderação, ou seja, o proprietário do canal pode expulsar pessoas que violem as normas de funcionamento e mesmo impedir de entrar no canal.

Na rede *Freenode*¹¹ estão alojados vários canais de projectos *open source* e *free software*, nomeadamente canais do *Ubuntu*. O *IRC* é uma ferramenta muito importante para esta comunidade, não só para que os utilizadores possam obter ajuda personalizada e instantânea mas também para coordenar o trabalho que é desenvolvido. O *Ubuntu* tem um canal *#ubuntu-meeting* que é utilizado para as reuniões de projectos dentro da comunidade *Ubuntu*. Nestas reuniões é decidido o que tem de ser feito e os prazos, e tal como numa reunião no mundo físico têm hora marcada, um moderador e no fim da reunião é elaborada uma acta¹². Assim os programadores que estão em diferentes partes do mundo podem reunir-se para discutir e elaborar uma ordem de trabalhos.

3.3.3 Sites e Wikis

Os sites podem suportar comunicação assíncrona e síncrona. *«Through the use of various software tools, web sites can host asynchronous discussion groups as well as real-time text chat.»* (Smith e Kollock, 1999:7).

Devido à sua interface gráfica e à capacidade de integrar imagens e sons, os sites podem criar uma experiência mais rica e mais intuitiva para os utilizadores. Como navegar na Internet é uma experiência familiar para a maioria das pessoas torna-se mais intuitivo entrar numa discussão através de um site. Além disso, através de apresentação gráfica, imagens e sons, os sites dão o mote para o tipo de interacção que promovem. Neste meio a comunicação de um-para-um pode ser mais difícil, porém a interacção entre o grupo como um todo é potencializada.

Os *wikis*¹³ também permitem um método de interacção muito particular, pois permitem que os utilizadores possam editar o conteúdo dos sites. Os *wikis* foram inventados em 1995, no entanto só em

11 Rede de IRC (chat.freenode.net) usada para projectos de software livre e código aberto.

12 Ver anexo D

13 Wiki é uma página ou conjunto de páginas da Internet concebidos para permitir que as pessoas que acede ao site possa modificar o seu conteúdo, usando uma markup language simplificada. Os wikis são habitualmente usados para criar sites colaborativos e potencializar sites de comunidades. A palavra “wiki” significa rápido ou super rápido em Havaiano.

*Code*¹⁷, e outros. Neste momento o *Launchpad* tem alojados 10214 projectos entre os quais o *Ubuntu*, o *MySQL*¹⁸, etc. «*Launchpad is a supermarket of free software projects. As a project owner, you can tell the world everything they need to know about your work: where you track bugs, where you host your code, who's involved, the details of your release plans and more.*» (<https://launchpad.net/>)

O código é a principal mercadoria do *free software* e do *open source*, por isso, torna-se essencial criar estratégias para facilitar a produção e partilha de código. O *Launchpad* permite o alojamento do código de projectos *open source* e *free software*, e para facilitar as contribuições utiliza *Bazaar*, que é um sistema de *source control* distribuído. Ou seja, permite que a comunidade contribua para o projecto, criando um novo ramo (*branch*) de desenvolvimento separado, e se o líder do projecto quiser poderá incluir o código produzido pelos membros da comunidade no seu projecto. Essas ramificações podem ser sub-projectos que se destacam do projecto principal ou actualizações para o projecto principal que podem ser incluídas neste facilmente. Este método de interacção permite uma fácil regulação do código e facilita as contribuições.

A identificação e correcção dos *bugs* também é muito importante na produção de software, seja este livre ou não. O *Launchpad* fornece uma ferramenta que permite seguir um determinado *bug* que existe em vários projectos e assim, quando esse *bug* for corrigido, vários projectos podem beneficiar dessa correcção. A interface para reportar um *bug* é fácil de utilizar, permitindo que qualquer membro da comunidade, mesmo alguém que não tem conhecimentos de programação, possa informar que encontrou um erro. Basta descrever o que acontece, identificar se alguém já encontrou esse *bug*, e enviar¹⁹.

A comunidade também pode contribuir para os projectos com traduções. Sem uma interface *user-friendly* torna-se muito complicado fazer traduções de software²⁰, por isso é necessário criar interfaces mais simples. No *Launchpad* o método de realizar traduções é simplificado para que qualquer pessoa consiga contribuir²¹, havendo sugestões de tradução e um interface que promove a interacção.

O *Launchpad* também tem uma área de respostas (Answers) para quem tem dúvidas sobre algum do software que está alojado nesta plataforma. Para além do género *Faq* (perguntas frequentes), muito

open source.

17 O Google Code é o site da Google para developers interessados em projectos relacionados com o Google e com software open source.

18 O MySQL é um sistema de gestão de bases de dados.

19 Ver anexo G

20 Ver anexo H

21 Ver anexo I

comum na *web*, o utilizador pode também fazer perguntas²² facilmente que depois serão respondidas pela comunidade.

Todas estas ferramentas permitem a criação de uma autêntica rede colaborativa à volta dos projectos *open source* e *free software*. Para Freeh, Madey e Tynan redes colaborativas como estas são uma variação das redes sociais, onde as relações que ligam as pessoas não são as amizades mas sim as colaborações. Se numa rede social as pessoas são nós e as ligações entre elas são relações de conhecimento – *social network theory* – então, vendo as comunidades *open source* e *free software* como redes sociais colaborativas, os *developers* são nós e a ligação entre eles é o projecto.

Analisando o funcionamento do Launchpad é possível ver como as comunidades *open source* e *free software* são semelhantes às redes sociais. No site pode-se aceder ao perfil²³ de cada colaborador e saber mais sobre ele, incluindo o seu *Karma*²⁴ – por cada colaboração que uma pessoa faz num projecto (uma tradução, reportação de um bug, etc.) aumenta o seu *Karma*, assim pode-se distinguir facilmente os membros mais activos da comunidade.

3.4 O sucesso das comunidades *open source* e *free software*

As comunidades *open source* e *free software* reúnem um conjunto de características que as tornam em comunidades de sucesso: identidade, interacção, voluntários e um interesse em comum. Segundo Raymond é mais barato e eficaz recrutar voluntários *self-selected* na Internet do que organizar edifícios cheios de pessoas que preferiam estar a fazer outra coisa qualquer (2001:25). É principalmente o interesse que faz com que as pessoas contribuam voluntariamente num projecto. «*My friend, familiar with both the open-source world and large closed projects, believes that open source has been successful partly because its culture only accepts the most talented 5% or so of the programming population.*» (Raymond, 2001:25).

Poderia-se pensar que uma comunidade que tem programadores movidos pelo seu ego e desejo de se afirmar na comunidade seria uma comunidade fragmentada, no entanto acontece exactamente o contrário. As expectativas das pessoas que duvidam do sucesso dos mercados livres são contrariadas

22 Ver anexo J

23 Ver anexo K

24 Ver anexo L

Internet como meio de produção de Código Aberto

pela variedade, qualidade e empenho que existe nas comunidades *open source* e *free software* (Raymond, 2001:23). Outra crítica que os cépticos fazem é que só o trabalho que é mais apetecível e sedutor é que será feito pela comunidade e que tudo o resto será mal feito ou deixado para último plano. «...*unless it's churned out by money-motivated cubicle peons with managers cracking whips over them.*» (Raymond, 2001:25). Porém a comunidade *free software* e *open source* já deu várias provas de que isto não é verdade, pois projectos que parecem menos interessantes para uns são muito interessantes para outros, existindo assim um equilíbrio natural.

«The development of the GNU Emacs editor is an extreme and instructive example; it has absorbed the efforts of hundreds of contributors over 15 years into a unified architectural vision, despite high turnover and the fact that only one person (its author) has been continuously active during all that time.» (Raymond, 2001:24).

Apesar do *free software* e do *open source* terem cada vez mais expressão no ciberespaço, no mundo físico ainda há muito que fazer para que estes sejam popularizados. Os membros destas comunidades têm um importante papel na divulgação. Com autocolantes, *pins* e roupas alusivas ao software livre e ao código aberto tentam afirmar que pertencem a estas comunidades e assim cativar pessoas que desconhecem estas alternativas livres. As comunidades *free software* e *open source* destacam-se assim das outras comunidades virtuais pelo efeito que têm não só no ciberespaço mas também na vida quotidiana dos seus membros, que muitas vezes sentem necessidade de se conhecer e reunir fora da *web*.

Conclusão

Este trabalho teve o objectivo de fazer uma caracterização, à luz das ciências sociais, do fenómeno do software livre e do código aberto, focando a importância das comunidades associadas a estes movimentos e a importância da Internet como media que potencializou o seu aparecimento/desenvolvimento.

Segundo Eric Raymond, o futuro do software *open source* está nas mãos de quem consegue incorporar o sistema do bazar e deixar para trás o sistema de produção de software da catedral. «*This is not to say that individual vision and brilliance will no longer matter; rather, I think that the cutting edge of open-source software will belong to people who start from individual vision and brilliance, then amplify it through the effective construction of voluntary communities of interest.*» (Raymond, 2001:23). As comunidades terão cada vez mais importância no desenvolvimento de bom software.

Desde o início do movimento em prol do software livre e do código aberto que muitas pessoas se questionam sobre a qualidade e inovação do software que nasce no seio destes movimentos. Tendo em conta que o *free software* e o *open source* promovem a criatividade dos programadores, pois estes trabalham por livre e espontânea vontade, certo será dizer que promove a inovação.

«*There is no algorithm for community. That is, there is no step-by-step recipe that can be followed that will guarantee a specific outcome.*» (Kollock, 1996). Apesar de terem todas uma estrutura semelhante, cada comunidade *open source* e *free software* é única, pois a sua estrutura depende da natureza do sistema, dos membros da comunidade e das razões que os movem.

Bibliografia

- Agre, P. (1995). *Designing genres for new media: Social, economic, and political contexts*. Acedido a: 7 de Fevereiro de 2009, em: <http://polaris.gseis.ucla.edu/pagre/genre.html>
- Bakhtin, M. (1987). *Speech Genres and Other Late Essays*. University of Texas Press. Austin, Texas.
- Deci, E. e Ryan, R. (2000). *Intrinsic and Extrinsic Motivations: Classic Definitions and New Directions*. Contemporary Educational Psychology.
- Dempsey, B., Greenber, J., Jones, P. e Weiss, D. (1999). *A quantitative profile of a community of open source linux developers*. Acedido a: 27 de Janeiro de 2009, em: <http://www.ibiblio.org/osrt/develop.html>
- Downes, E. e McMillan, S. (2000). *Defining interactivity: A qualitative identification of key dimensions*. Volume 2, New Media & Society.
- Fogel, K. (2005). *Producing open source software: How to run a successful free software project*. 1ª edição, O'Reilly Media. Sebastopol.
- Freeh, V., Madey, G. e Tynan, R. (2002). *The open source development phenomenon: An analysis based on social network theory*. Acedido a: 26 de Janeiro de 2009, em: www.nd.edu/~oss/Papers/amcis_oss.pdf
- Jensen, J. (1998) *Interactivity: Tracing a new concept in media and communication studies*. Acedido a: 2 de Fevereiro de 2009, em www.nordicom.gu.se/common/publ_pdf/38_jensen.pdf
- Klandermans, B. (1997). *The social psychology of protest*. Basil Blackwell, Oxford.
- Kollock, P. 1996. *Design Principles for Online Communities*. Acedido a: 2 de Fevereiro de 2009, em <http://www.sscnet.ucla.edu/soc/faculty/kollock/papers/design.htm>
- Kollock, P. e Smith, M. (1999). *Communities in Cyberspace*. Acedido a: 2 de Fevereiro de 2009, em http://www.sscnet.ucla.edu/soc/faculty/kollock/papers/communities_01.htm
- Lakhani K. e Wolf R. (2005) *Why Hackers Do What They Do: Understanding Motivation and Effort in Free/Open Source Software Projects*. MIT Press
- Lévy, P., (1997). *A inteligência colectiva : Para uma antropologia do ciberespaço*. Colecção Epistemologia e sociedade, Instituto Piaget. Lisboa
- McMillan, S. (1998). *Who pays for content? Funding in interactive media*. Acedido a: 30 de

Internet como meio de produção de Código Aberto

Janeiro de 2009, em: <http://jcmc.indiana.edu/vol4/issue1/mcmillan.html>

- Nakakoji, K., Yamamoto, Y., Nishinaka, Y., Kishida, K. e Ye, Y. Evolution Patterns of Open-Source Software Systems and Communities. Acedido a: 3 de Fevereiro de 2009 em www.kid.rcast.u-tokyo.ac.jp/~kumiyo/mypapers/IWPSE2002.pdf
- Raymond, E. (2001). *The cathedral and the bazaar: Musings on linux and open source by an accidental revolutionary*. Edição revista, O'Reilly.
- Smith, M. e Kollock, P. (1999). *Communities in Cyberspace*. 1ª edição, Routledge. Londres.
- Sterling, B. (1994). *Hacker crackdown: Law and disorder on the electronic frontier*. Reedição, Bantam Books.
- Straubhaar, J. e LaRose, R. (1996). *Communications Media in the Information Society*. Wadsworth Press. Belmont, California

Anexos

Anexo A

GNU General Public License

Version 3, 29 June 2007

Terms and conditions

0. Definitions.

"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the

public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the

interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.

1. Source Code.

The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are

irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.

4. Conveying Verbatim Copies.

You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

Internet como meio de produção de Código Aberto

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

- a) The work must carry prominent notices stating that you modified it, and giving a relevant date.
- b) The work must carry prominent notices stating that it is released under this License and any conditions added under section

7. This requirement modifies the requirement in section 4 to "keep intact all notices".

c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.

d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in

one of these ways:

a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.

b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.

d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful

cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program

Internet como meio de produção de Código Aberto

shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

- a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or
- b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or
- c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or
- d) Limiting the use for publicity purposes of names of licensors or authors of the material; or
- e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or
- f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered "further restrictions" within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.

9. Acceptance Not Required for Having Copies.

You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment,

Internet como meio de produção de Código Aberto

however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.

12. No Surrender of Others' Freedom.

If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Internet como meio de produção de Código Aberto

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

There is no warranty for the program, to the extent permitted by applicable law. except when otherwise stated in writing the copyright holders and/or other parties provide the program "as is" without warranty of any kind, either expressed or implied, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. the entire risk as to the quality and performance of the program is with you. should the program prove defective, you assume the cost of all necessary servicing, repair or correction.

16. Limitation of Liability.

In no event unless required by applicable law or agreed to in writing will any copyright holder, or any other party who modifies and/or conveys the program as permitted above, be liable to you for damages, including any general, special, incidental or consequential damages arising out of the use or inability to use the program (including but not limited to loss of data or data being rendered inaccurate or losses sustained by you or third parties or a failure of the program to operate with any other programs), even if such holder or other party has been advised of the possibility of such damages.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

Anexo B

The Open Source Definition (Version 1.0)

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

1. Free Redistribution

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

2. Source Code

The program must include source code, and must allow distribution in source code as well as compiled form. Where some form of a product is not distributed with source code, there must be a well-publicized means of downloading the source code, without charge, via the Internet. The source code must be the preferred form in which a programmer would modify the program. Deliberately obfuscated source code is not allowed. Intermediate forms such as the output of a preprocessor or translator are not allowed.

3. Derived Works

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

4. Integrity of The Author's Source Code.

The license may restrict source-code from being distributed in modified form only if the license allows the distribution of "patch files" with the source code for the purpose of modifying the program at build time.

Internet como meio de produção de Código Aberto

The license must explicitly permit distribution of software built from modified source code. The license may require derived works to carry a different name or version number from the original software.

5. No Discrimination Against Persons or Groups.

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

6. No Discrimination Against Fields of Endeavor.

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

7. Distribution of License.

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

8. License Must Not Be Specific to a Product.

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

9. License Must Not Contaminate Other Software.

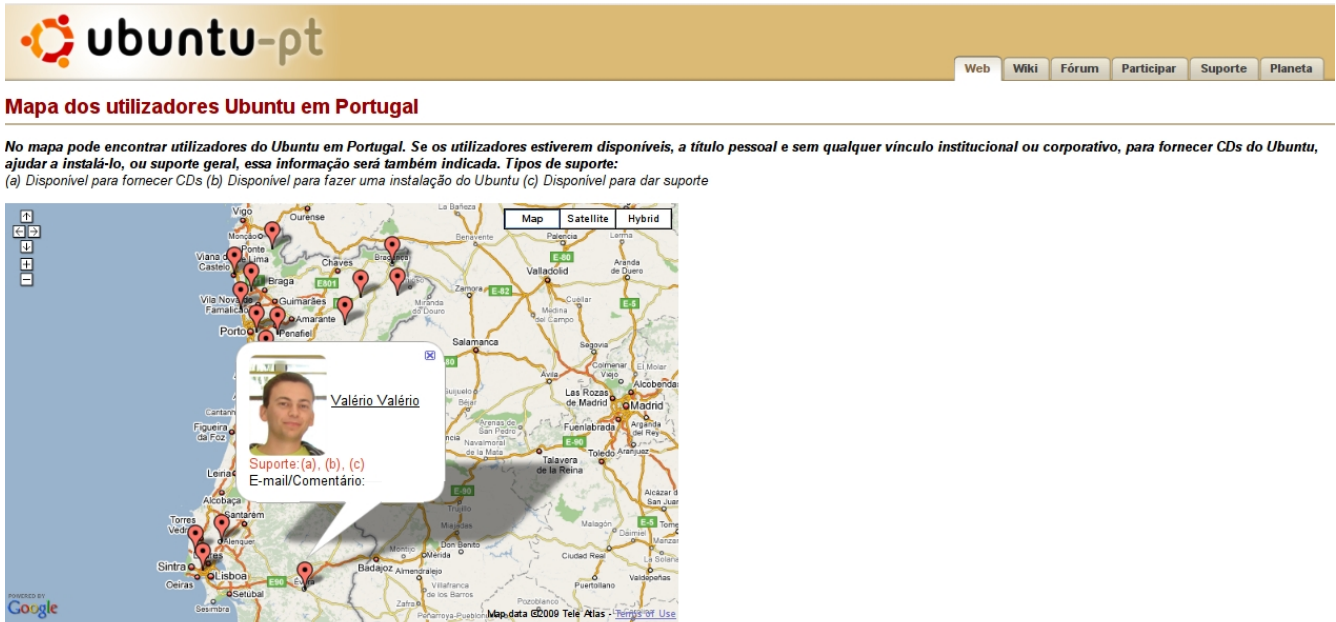
The license must not place restrictions on other software that is distributed along with the licensed software. For example, the license must not insist that all other programs distributed on the same medium must be open-source software.

10. Example Licenses.

The GNU GPL, BSD, X Consortium, and Artistic licenses are examples of licenses that we consider conformant to the Open Source Definition. So is the MPL.

Anexo C

Sistema de localização de membros da comunidade dispostos a dar apoio técnico face-a-face



The screenshot shows the 'ubuntu-pt' website header with navigation links: Web, Wiki, Fórum, Participar, Suporte, Planeta. Below the header is the title 'Mapa dos utilizadores Ubuntu em Portugal'. A paragraph of text explains that the map shows users available for support, with details on how to use the map and what the support types mean. The map itself shows various locations in Portugal with red pins. A callout box for 'Valério Valério' is visible, showing a profile picture and the text 'Suporte: (a), (b), (c)' and 'E-mail/Comentário:'. The map is powered by Google and includes a copyright notice for 2009.

ubuntu-pt Web Wiki Fórum Participar Suporte Planeta

Mapa dos utilizadores Ubuntu em Portugal

No mapa pode encontrar utilizadores do Ubuntu em Portugal. Se os utilizadores estiverem disponíveis, a título pessoal e sem qualquer vínculo institucional ou corporativo, para fornecer CDs do Ubuntu, ajudar a instalá-lo, ou suporte geral, essa informação será também indicada. Tipos de suporte:
(a) Disponível para fornecer CDs (b) Disponível para fazer uma instalação do Ubuntu (c) Disponível para dar suporte

Map Satellite Hybrid

Valério Valério

Suporte: (a), (b), (c)

E-mail/Comentário:

POWERED BY Google

Map data ©2009 Tele Atlas. Terms of Use

Anexo D

Acta de uma reunião no #ubuntu-meeting

Classroom/Meetings/Minutes2008Mar16

Meeting 2008 Mar 16th

- Meeting Topic - How to proceed for April

Team Page: <https://wiki.ubuntu.com/Classroom>

Meeting Minutes by: Trevor Sharpe

Present

The following were present during this meeting:

- visualperception
- Zelut
- Heartsbane
- herlo
- pleia2
- highvoltage

Agenda

<https://wiki.ubuntu.com/Classroom/MeetingAgenda2008Mar16>

Proceedings

- Meeting called to order at **1:59PM EST (GMT -0500)**, by Christer Edwards(Zelut)
- Review of Agenda
 - Nothing from US Teams
 - Solid date for Openweek
 - awaiting reply from Jono Bacon
 - New Wiki Layout: completed
 - Announcement banner needed
 - Marketing blog widgets
 - To be discussed on mailing list
 - artwork
 - slogans
 - Discuss Hiveminder for group assignments
 - Is it a barrier to a open disclosure
 - Output methods: RSS, atom, ical, plain text
 - continue to use gobby?: **gobby no longer in use except for group collaboration**
 - Ubuntu-training email do we need to contact Belinda Lopez to perhaps join us?
- Follow up meeting date/time was established
- Classroom Guidelines moved back to next meeting
- Meeting called to end at **3:05PM EST (GMT -0500)**, by Christer Edwards(Zelut)

Assignments

- Contact Belinda Lopez at Ubuntu Training: pleia2
- Announcement banner: heartsbane
- Widgets Layout: visualperception

Anexo E

Ubuntu wiki



Artwork » BugSquad/Meeting/Minutes » 2006-03-28 » Minutes2008Mar16 » Ubuntu Team Wiki »

Immutable Page [Info](#) [Attachments](#) More Actions:

[Find Page](#) [User Preferences](#)

Ubuntu Team Wiki

-  [Community Team Wiki](#)
[Ubuntu Help](#)
-  [Get Involved](#)
[Community](#)
[Teams](#)
-  [Events](#)
[Weekly Newsletter](#)
[Release Schedule](#)

Welcome to the Ubuntu Community Team Wiki!

This is **THE** definitive place for members of the Ubuntu community to discuss ideas and store team-related information.

Released (Current + Stable)			
Hardy Heron	8.04 LTS	April 24, 2008	Supported until April 2013 (Server) or April 2011 (Desktop)
Intrepid Ibex	8.10	October 30, 2008	Supported until April 2010
To be Released	Version	Release Schedule	Support Date
Jaunty Jackalope	9.04	April 23, 2009	Supported until October 2010

For the **FULL** Release Schedule, go to the [Releases](#) page.

Anexo F

Diferenças entre o Launchpad e o Sourceforge

Name	Code hosting	Code review	Bug tracking	Web hosting	Wiki	Translation system	Shell server	Mailing list	News groups (NNTP)	Forum	Personal branch	Private branch	Announce	Build system	Team	Other
Launchpad	Yes	Yes	Yes	No	No	Yes	No	Yes	No	No	Yes	Yes (security only)	Yes	Yes (Ubuntu)	Yes	Blueprints Kama Answer Mail interface Public API OpenID Provider Download area Non-project branch
SourceForge	Yes	No	Yes	Yes	Yes	No	Yes	Yes	No	Yes	No	No	Yes	No	No	Marketplace Jobs Help Wanted OpenID Relying Party Download area

Anexo G

Launchpad – Interface para reportar bugs

launchpad Ubuntu "firefox-3.0" package phreakia Log Out

Overview Code Bugs Blueprints Translations Answers

Is the bug you're reporting one of these?

- #228806 [MASTER] various unrelated crashes for firefox-3.0 and xulrunner-1.9
Confirmed, last updated 7 hours ago
- #243174 Firefox 3 crashes when Shockwave Flash is activated and visiting a site with flash inside
Invalid, last updated 2008-09-04
- #209634 firefox + flash video: crash playing second movie
Invalid, last updated 2008-06-27
- #280545 flash plugin crashed when screensaver interrupted full-screen video
Invalid, last updated 2008-12-22
- #275371 FF3 crash - belive flash related
Invalid, last updated 2008-12-22
- #205433 Firefox3 crashes frequently with flash-nonfree in use
Invalid, last updated 2008-03-26
- #213673 Flash page crashed Firefox 3.0 beta 5
Invalid, last updated 2008-12-22
- #203277 firefox crashed with SIGSEGV: running flash presentation
Invalid, last updated 2008-03-18
- #313557 Firefox crashes when Adobe Flash is utilized
Invalid, last updated 2009-01-04
- #203989 firefox 3-0 crashed with SIGSEGV and flash items
Invalid, last updated 2008-03-28

No, I'd like to report a new bug

In what package did you find this bug?

I don't know

(Choose...)

Summary:

Further information:

Ubuntu guidelines:

Please include, if possible:

The source package you found the bug in, for help see <https://wiki.ubuntu.com/Bugs/FindRightPackage> .

1) The release of Ubuntu you are using, via 'lsb_release -rd' or System -> About Ubuntu.

Anexo H

Método tradicional de tradução de software

```
1 # Translation of epiphany to Croatiann
2 # Copyright (C) Croatiann team
3 # Translators: Automatski Prijevod <>, Danijel Studen <dstudent@vuka.hr>, Denis Lackovic <delacko@fly.srk.fer.hr>, Ivan Knežević <kn3z@lycos.com>, Nikola Planinac <>, pr p
4 msgid ""
5 msgstr ""
6 "Project-Id-Version: epiphany 0\n"
7 "Report-Msgid-Bugs-To: \n"
8 "POT-Creation-Date: 2008-09-07 22:48+0200\n"
9 "PO-Revision-Date: 2008-04-15 06:09+0000\n"
10 "Last-Translator: Launchpad Translations Administrators <rosetta@launchpad."
11 "net>\n"
12 "Language-Team: Croatian <lokalizacija@linux.hr>\n"
13 "MIME-Version: 1.0\n"
14 "Content-Type: text/plain; charset=UTF-8\n"
15 "Content-Transfer-Encoding: 8bit\n"
16 "Plural-Forms: nplurals=3; plural=n%10==1 && n%100!=11 ? 0 : n%10>=2 && n%"
17 "10<=4 && (n%100<10 || n%100>=20) ? 1 : 2;\n"
18 "X-Launchpad-Export-Date: 2008-05-27 22:27+0000\n"
19 "X-Generator: Launchpad (build Unknown)\n"
20
21 #: ../data/bme.desktop.in.in.h:1
22 msgid "Browse and organize your bookmarks"
23 msgstr "Pregledaj i organiziraj svoje zabilješke"
24
25 #: ../data/bme.desktop.in.in.h:2
26 msgid "Epiphany Web Bookmarks"
```

Anexo I

Launchpad – Interface para traduções

launchpad > Ubuntu 9.04

Overview | Code | Bugs | Blueprints | **Translations** | Answers

Translation status by language Help for translations

Language	Contributors	Status	Untranslated	Need review	Changed
Achinese	39	<div style="width: 100%; height: 10px; background-color: red;"></div>	466646	—	—
Afar	10	<div style="width: 100%; height: 10px; background-color: red;"></div>	467714	—	—
Afrikaans	60	<div style="width: 95%; height: 10px; background-color: red;"></div>	381422	451	601
Akan	2	<div style="width: 100%; height: 10px; background-color: red;"></div>	467771	—	—
Albanian	38	<div style="width: 95%; height: 10px; background-color: red;"></div>	412840	615	945
Amharic	16	<div style="width: 100%; height: 10px; background-color: red;"></div>	456659	66	186
Arabic	145	<div style="width: 95%; height: 10px; background-color: red;"></div>	313147	1720	2383
Aragonese	4	<div style="width: 100%; height: 10px; background-color: red;"></div>	467761	—	—
Armenian	16	<div style="width: 100%; height: 10px; background-color: red;"></div>	461588	139	242
Assamese	10	<div style="width: 95%; height: 10px; background-color: red;"></div>	419094	76	—
Asturian	25	<div style="width: 95%; height: 10px; background-color: red;"></div>	318552	288	642
Avaric	2	<div style="width: 100%; height: 10px; background-color: red;"></div>	467694	—	—
Aymara	3	<div style="width: 100%; height: 10px; background-color: red;"></div>	467760	—	—
Azerbaijani	21	<div style="width: 95%; height: 10px; background-color: red;"></div>	427300	40	324
Baluchi	2	<div style="width: 100%; height: 10px; background-color: red;"></div>	467757	—	—
Basque	88	<div style="width: 95%; height: 10px; background-color: red;"></div>	221602	3793	5619
Belarusian	46	<div style="width: 95%; height: 10px; background-color: red;"></div>	370197	2456	3666
Bengali	53	<div style="width: 95%; height: 10px; background-color: red;"></div>	403820	2284	3631
Berber (Other)	2	<div style="width: 100%; height: 10px; background-color: red;"></div>	467790	—	—

> See import queue
> See language packs

Translation permissions

Translation Ubuntu group: Translators
Permissions: Structured

Your languages

Asturian
Portuguese

+ Select Languages...

launchpad > Ubuntu 8.10 "ubuntu-docs" package

Overview | Code | Bugs | Blueprints | **Translations** | Answers

Description | **Translate** | Download

Template "about-ubuntu" in Ubuntu Intrepid package "ubuntu-docs"

Translating into Portuguese

Translating using as a guide Search:

11 → 20 of 49 results First · Previous · Next · Last

11. **English:** Ubuntu includes the very best in translations and accessibility infrastructure that the free software community has to offer, to make Ubuntu usable for as many people as possible.

Current Portuguese: O Ubuntu inclui a melhor infra-estrutura de suporte a traduções e acessibilidade que a comunidade de aplicações informáticas livres ("Free Software") tem para oferecer, de modo a tornar o Ubuntu utilizável pelo maior número de pessoas possível.
Translated and reviewed by [leonorhungria](#) on 2007-04-04

Suggestions:

O Ubuntu inclui a melhor infra-estrutura de suporte a traduções e acessibilidade que a comunidade de aplicações informáticas livres ("Free Software") tem para oferecer, de modo a tornar o Ubuntu utilizável pelo maior número de pessoas.
Used in [aboutubuntu](#) in Ubuntu Dapper package "ubuntu-docs" by [Rui Az.](#) on 2006-04-14

New suggestion:

Anexo J

Launchpad – Answers

Language: ▼

The language in which this question is written. The languages marked with a star (*) are the languages spoken by at least one answer contact in the community.

Summary:

A one-line summary of the issue or problem.

Description:

Include as much detail as possible: what you're trying to achieve, what steps you take, what happens, and what you think should happen instead.

Anexo K

Launchpad – Perfil



Mark Shuttleworth

South African living in London, '73 edition. I'm the leader of the Ubuntu Project (<http://www.ubuntu.com/>) and one of the architects of Launchpad. More details on my home page at <http://www.markshuttleworth.com/>

See my Ubuntu wiki home page for FAQ's about my role in Ubuntu, and the general direction of the project.

Email: No public address provided. ([Contact this user](#))

Wiki: <https://wiki.ubuntu.com/MarkShuttleworth>

IRC: [sabdfl](#) on network [irc.freenode.net](#)

OpenPGP keys: [D54F0847](#)

SSH Keys: [mark@dell-desktop](#)

([view key text](#))

OpenID login: <https://launchpad.net/~sabdfl> ([What's this?](#))

Location **Time zone:** Europe/London



Most active in

- [gnome-control-center](#)
- [Launchpad itself](#)
- [Ubuntu](#)
- [Launchpad Foundations](#)
- [Soyuz - The Launchpad Pac...](#)

Working on...

[#1 Microsoft has a majority market share](#) [also in Clubdistro, Computer Science Ubuntu, Ichthux, JAK LINUX, OpenOffice, Launchpad Translations, Tabuntu, Baltix, Debian, Fluxbuntu, Tilix](#), reported on 2004-12-17 by [Mark Shuttleworth](#)

[#50699 Launchpad should be free software \(free as in freedom\)](#)
[also in Launchpad Itself, Launchpad Answers, Launchpad Bugs, Launchpad Translations, Ubuntu](#), reported on 2006-06-22 by [Kurt Kraut](#)

[» Show all bugs in progress](#)

Anexo L

Launchpad – Karma

Launchpad karma

This is a summary of the Launchpad karma earned by Mark Shuttleworth, organized by activity type.

Category	Points
Bazaar Branches	1201
Bug Management	112
Total: 1313	

Latest actions

Action	Date
Commented on a review	3 hours ago
Commented on a review	3 hours ago
Commented on a review	6 hours ago
Bug Comment Added	2009-02-10
Reviewer commented on a review	2009-01-31
Reviewer commented on a review	2009-01-30
Reviewer commented on a review	2009-01-29
Reviewer commented on a review	2009-01-29
Reviewer commented on a review	2009-01-29
Reviewer commented on a review	2009-01-29
Reviewer commented on a review	2009-01-29
Bug Marked as Duplicate	2009-01-13
Bug Comment Added	2009-01-13
New Bug Filed	2009-01-12
New Bug Filed	2009-01-12
Bug Comment Added	2009-01-12
Bug Comment Added	2009-01-07
Bug Comment Added	2009-01-07
Reviewer commented on a review	2008-11-18
Bug Comment Added	2008-11-11
Reviewer commented on a review	2008-11-07
Bug Comment Added	2008-11-06
Bug Comment Added	2008-11-06