Avaliação como Forma de Incentivo ao Desenvolvimento de Software Livre em Ambiente Acadêmico

Francisco Reverbel

Departamento de Ciência da Computação
Instituto de Matemática e Estatística da Universidade de São Paulo

reverbel@ime.usp.br

Abstract. *This paper discusses criteria for evaluating the work of free software developers in an academic environment. It presents a set of parameters that are applicable to free software projects and can be used as indicators of significance. We suggest that traditional academic evaluation should be complemented by these parameters. This would be an effective way of fostering the innovative creation of free software in academia.*

Resumo. *Este artigo discute critérios de avaliação de atividades de desenvolvimento de software livre realizadas em ambiente acadêmico. É apresentado um conjunto de indicadores de relevância de projetos de software livre. Sugerimos que o sistema de avaliação acadêmica tradicional seja complementado com esses indicadores. Isso seria uma forma efetiva de incentivar a criação inovadora de software livre em ambiente acadêmico.*

1. Introdução

É crescente o reconhecimento da importância econômica e tecnológica do software livre. Formas de incentivo a projetos de pesquisa avançada que produzam software livre e/ou de fonte aberto estão sendo estudadas tanto no âmbito da Comissão Européia [European Working Group on Libre Software, 2000] como no do governo dos Estados Unidos [President's Information Technology Advisory Committee, 2000].

O software livre tem o potencial de contribuir para a redução do fosso tecnológico entre ricos e pobres [Kon, 2001]. Isto sugere que seu impacto positivo pode ser ainda mais forte sobre nações menos desenvolvidas. Para essas nações, o software livre não é apenas um modo de se evitar os custos do software fechado e a dependência de fornecedores monopolistas. Ele é, sobretudo, uma forma de acesso a tecnologias avançadas que dá a países menos desenvolvidos a possibilidade de participar efetivamente do processo de construção dessas tecnologias. É oportuno, portanto, que a questão dos incentivos ao software livre tenha começado a ser discutida em nosso país. Em particular, interessa-nos aqui a questão dos incentivos ao software livre no ambiente acadêmico [Simon, 2002].

O novo modelo de criação e distribuição de software coloca novos desafios para a Universidade brasileira: a difusão do software livre tanto no ambiente acadêmico como fora dele, a formação de recursos humanos capacitados para trabalhar com software livre e o desenvolvimento de software livre que contenha inovações tecnológicas. Este artigo aborda um tópico relacionado com o terceiro desses desafios: a avaliação como forma de incentivo ao desenvolvimento de software livre no ambiente acadêmico.

A criação de software tecnologicamente inovador é uma atividade de pesquisa que requer níveis muito altos de esforço e dedicação. Dependendo da natureza do software que se está desenvolvendo, os trabalhos de implementação podem ser inteiramente realizados por alunos de

pós-graduação ou de graduação, devidamente orientados por docentes universitários. Sistemas de software mais complexos, entretanto, requerem a intervenção direta de pessoas com maior conhecimento e experiência (os docentes) nos trabalhos de implementação. Surge aqui um problema: não há, no Brasil, nehum incentivo para que docentes universitários desempenhem tais tarefas, que tendem a consumir muito tempo. A avaliação docente é baseda na produção acadêmica tradicional, medida geralmente em termos de artigos publicados em revistas e em conferências científicas. O software inovador que um docente desenvolveu não é levado em conta por agências de fomento à pesquisa como CNPq e FAPESP, para efeito de concessão de verbas de pesquisa ao docente, nem tampouco por Universidades como a USP, para efeito da progressão do docente em sua carreira.

Nos países desenvolvidos é bem mais comum o envolvimento de docentes universitários em atividades de implementação de software livre.[1] Universidades americanas avaliam docentes das áreas de Ciência da Computação e Engenharia da Computação segundo os critérios explicitados em [Patterson et al., 1999], que levam em conta os "artefatos computacionais" (software, chips, etc.) produzidos pelo docente. Esse documento menciona alguns "indicadores de impacto" que podem ser usados na avaliação desse tipo de produção: número de *downloads* de um artefato de software, número de usuários, numero de visitas numa página Web, etc. No Brasil, entretanto, as agências de fomento à pesquisa e as Universidades relutam em atualizar seus critérios de avaliação. Duas dúvidas são recorrentes entre os avaliadores:

- Como reconhecer um projeto de software livre relevante e inovador?
- Como saber se uma pessoa efetivamente contribuiu para determinado projeto?

O autor se baseia em sua experiência de participação em dois projetos internacionais de software livre — o JBoss [Fleury and Reverbel, 2003, JBoss Group, 2003] e o JacORB [JacORB Team, 2003], ambos muito bem sucedidos — para oferecer respostas parciais a essas questões no restante deste artigo.

2. Indicadores de Relevância de Projetos de Software Livre

A relevância de um projeto de software livre é geralmente associada a um conjunto de indicadores. O vínculo entre a relevância do projeto e os indicadores relacionados a seguir é fortemente sugerido pela análise, *a posteriori*, dos projetos de software livre mais influentes e bem sucedidos, como Linux, Apache, GNU C/C++, GNU Emacs, Perl, PostgreSQL, FreeBSD, Samba, Sendmail e Bind.

Comunidade de participantes ativa e geograficamente dispersa. Ao longo de sua vida um projeto relevante tende a atrair um grupo de participantes ativos, com capacidade e disponibilidade para oferecer contribuições efetivas. O critério da dispersão geográfica reduz a probabilidade dos participantes terem sido motivados por fatores não necessariamente relacionados com a relevância do projeto. Ele elimina projetos cujos participantes são todos funcionários de uma mesma empresa (que pode ter interesse específico no projeto) ou alunos de um mesmo programa de pós-graduação (possivelmente orientados por um professor que participa do projeto), etc.

Dimensão do projeto. Projetos relevantes tendem a ter certo porte. O número de linhas de código fonte é em geral maior que cem mil e freqüentemente maior que um milhão.

Usuários. Todo projeto relevante tem usuários. A existência de usuários, além de gerar realimentação vital para o ciclo evolutivo do projeto, indica que ele atende a uma necessidade real e que ultrapassou o estado de protótipo. Tipicamente não existe uma contabilização precisa dos usuários de um programa livre. O volume de usuários é inferido a partir de outros

[1] A propósito: a versão final deste artigo foi gerada por um sistema de processamento de texto — o TEX — implementado pessoalmente pelo Prof. Donald Knuth, da Universidade de Stanford.

indicadores, como número de downloads e tráfego em listas de usuários. A publicação e comercialização de livros descrevendo o software desenvolvido (freqüentemente escritos por participantes do projeto) e a existência de estruturas comerciais de apoio aos usuários são também fatores relacionados com o número de usuários.

Tráfego em listas de discussão. Todo projeto relevante tem pelo menos uma lista de discussão na Internet. As mensagens enviadas para a lista são mantidas num repositório público e acessível a quem quer que seja. Muitos projetos tem mais de uma lista de discussão. Um arranjo típico utiliza uma lista para participantes do projeto, exclusivamente voltada para a discussão de aspectos internos do software coletivamente construído, e outra lista para os usuários desse software. Projetos relevantes têm um certo tráfego de mensagens em suas listas de discussão. Este tráfego varia de projeto para projeto, de dezenas de mensagens por mês até centenas de mensagens por dia.

Existência de literatura. A partir de um certo ponto de sua vida um projeto relevante se torna assunto de livros cujo mercado potencial é o conjunto de usuários do software produzido.

Comercialização de serviços de apoio. A partir de um certo ponto da vida de um projeto relevante surgem empresas que comercializam serviços de apoio aos usuários do software desenvolvido.

De acordo com esses indicadores, um projeto de software livre pode ser considerado relevante mesmo que não esteja alinhado com o estado da arte em sua área. (O Linux é freqüentemente enquadrado nessa categoria, por não empregar uma arquitetura tipo microkernel.) Como não contempla a questão da inovação tecnológica, esse conjunto de parâmetros é certamente incompleto para a avaliação da relevância de atividades de desenvolvimento de software livre desenvolvidas num ambiente acadêmico. Ele deve ser complementado por uma análise do grau de inovação e das contribuições para o estado da arte presentes no software desenvolvido, ou por parâmetros adicionais, especificamente voltados para uma avaliação acadêmica. Segundo [Patterson et al., 1999], a utilização de determinado software por outros pesquisadores, como instrumento ou objeto de pesquisa, é uma boa indicação da relevância acadêmica e da influência do projeto que gerou esse software.

3. Atribuição de Créditos em Projetos de Software Livre

O modo como os participantes de um projeto de software livre se organizam internamente varia bastante de projeto para projeto. Praticamente todos os projetos bem sucedidos adotaram como organização social alguma forma de meritocracia. O mérito de cada participante é baseado no valor de suas contribuições para o projeto. O funcionamento da organização repousa portanto sobre o reconhecimento coletivo (que necessariamente inclui uma avaliação) da contribuição de cada um.

A correta atribuição de créditos pelas contribuições dos participantes é uma condição imprescindível para a harmonia interna da estrutura social do projeto. É extremamente difícil uma apropriação indevida do crédito pelo trabalho realizado por outrem. Na improvável hipótese de que tal apropriação ocorra, é virtualmente impossível que ela passe despercebida. Além disso, num projeto com certa importância, mesmo a cessão voluntária de crédito (prática muito difícil de se eliminar num ambiente acadêmico tradicional) é inviabilizada pelo fato dos trabalhos serem realizados em público. A história completa das modificações nos arquivos fonte é mantida num repositório acessível a todos os participantes (e a toda e qualquer pessoa, em muitos projetos). Esse repositório é utilizado diariamente pelos membros ativos do projeto. Em muitos projetos o repositório de arquivos fonte é configurado para enviar automaticamente uma mensagem informativa a todos os participantes sempre que algum arquivo for modificado. Tais mensagens contém os nomes dos arquivos modificados, as alterações efetuadas e o participante responsável pelas alterações.

A quase totalidade dos projetos tem pelo menos duas formas básicas de reconhecimento pelas contribuições de seus participantes:

Direito de escrita no repositório de arquivos fonte. É a forma mais elementar de reconhecimento. Um "*committer*" é um participante que tem o direito de efetivar alterações ("*commit*") no repositório de fontes do projeto. Novos participantes somente se tornam *committers* depois de demonstrarem um certo nível (que varia bastante entre projetos) de dedicação e de competência. Do ponto de vista de segurança, é indesejável a existência de *committers* que não estejam efetivamente contribuindo para o projeto. Por esse motivo, muitos projetos adotam alguma política de revogação do direito de escrita de participantes inativos.

Divulgação do nome do participante. Os nomes dos participantes são geralmente publicados na página do projeto, com grau de destaque proporcional à importância de suas contribuições.

Além dessas formas elementares de reconhecimento, projetos com uma estrutura organizacional mais complexa adotam esquemas mais elaborados para definir algum tipo de diferenciação hierárquica entre seus participantes.

4. Conclusão

A criação inovadora de software livre em ambiente acadêmico pode ser incentivada significativamente com uma acão das agências de fomento à pesquisa e das Universidades: a atualização dos critérios de avaliação de pesquisadores e docentes, de modo a levar em conta o software livre por eles produzido. Este artigo é uma contribuição nesse sentido. Não temos, no entanto, a pretensão de ter identificado um conjunto de parâmetros adequado para avaliar todo e qualquer projeto de software livre. Por estar fortemente relacionada com a existência de usuários, boa parte dos parâmetros identificados neste artigo não capta a relevância de projetos que ainda se encontrem nos estágios iniciais de seu ciclo evolutivo. Mesmo assim, acreditamos que um conjunto de indicadores nos moldes do aqui proposto pode enriquecer muito o processo de avaliação da produção acadêmica.

Referências

European Working Group on Libre Software (2000). "Free Software / Open Source: Information Society Opportunities for Europe?". European Commission Information Society. `http://eu.conecta.it/paper.pdf`.

Fleury, M. and Reverbel, F. (2003). The JBoss Extensible Server. In *Middleware 2003 — ACM/IFIP/USENIX International Middleware Conference*, volume 2672 of *LNCS*, pages 345–374. Springer-Verlag. `http://www.ime.usp.br/~reverbel/papers/jboss-mw2003.pdf`.

JacORB Team (2003). JacORB web site. `http://www.jacorb.org`.

JBoss Group (2003). JBoss web site. `http://www.jboss.org`.

Kon, F. (2001). O Software Aberto e a Questão Social. Relatório Técnico RT-MAC-2001-07, Departamento de Ciência da Computação, IME-USP. `http://www.ime.usp.br/~kon/papers/RT-SoftwareAberto.pdf`.

Patterson, D., Snyder, L., and Ullman, J. (1999). "Evaluating Computer Scientists and Engineers for Promotion and Tenure". Best Practices Memo, Computer Research Association. `http://www.cra.org/reports/tenure_review.html`.

President's Information Technology Advisory Committee (2000). "Developing Open Source Software to Advance High End Computing". Report to the President, Executive Office of the President of the United States. `http://www.itrd.gov/pubs/pitac/pres-oss-11sep00.pdf`.

Simon, I. (2002). "O Software Livre precisa de incentivos? Aonde? De que tipo? Como proceder?". Palestra proferida no I Forum Nacional de Software Livre em Instituições de Ensino Superior (SOLIES), Universidade Federal de São Carlos. `http://www.ime.usp.br/~is/aula/incentivos/`.