

A Catedral e o Bazar

um trabalho seminal de Eric Raymond

Imre Simon
Universidade de São Paulo
São Paulo, Brasil

[<is@ime.usp.br>](mailto:is@ime.usp.br)

<http://www.ime.usp.br/~is/>

Foto da época em que o artigo foi escrito

nos idos de 1997:

- O Linux, iniciado 6 anos atrás, tem 3 milhões de usuários (?)
- O Software Livre tem 13 anos de vida e, mesmo na perspectiva do seu enorme sucesso, passa por uma inflexão: sua adoção está atingindo escalas sociais
- A Internet, iniciada 28 anos atrás, tem penetração social em x países e possui 40 milhões de hosts conectados
- não existem (ou são irrelevantes): Google, slashdot, SourceForge, termo open-source, amazon.com, Windows98, ...

ninguém entende como o software GNU/Linux não pára de ampliar o seu espaço, apesar da popularidade crescente do sistema Windows, apoiado por uma propaganda maciça

Eric Raymond

- observa o fenômeno,
- caracteriza a programação cooperativa em larga escala,
- contrasta este método com os métodos anteriormente praticados
- e descreve suas explicações num artigo para o qual encontra um título magicamente sugestivo: “A Catedral e o Bazar”

Sobre o autor: Eric Raymond

hacker, matemático, antropólogo

coordenador do jargon file,

um dos maiores estudiosos da cultura hacker (“the good guys”, rebeldes dentro do sistema, como a GPL, por exemplo), que deve ser contrastada com a cultura cracker (“the bad guys”, rebeldes fora do sistema)

um dos desenvolvedores do emacs

um dos autores da 2ª edição do livro Learning GNU Emacs, 1996, (com isto contraria o Stallman, que defende a FDL)

vai lançar o movimento open-source em 1998, procurando efetivamente envolver empresas na transição para o software de fonte aberto

escreve muito bem

Transições de fase relevantes a este trabalho

Internet : 1969 -> 1997 -> hoje -> ...

Métodos de Programação : Catedral -> Bazar

- Catedral -> Catedral + Bazar, mas precisamente
- transições anteriores:
 - ▷ *linguagem de máquina (mulheres programadoras) -> linguagem compilada*
 - ▷ *método do “Mythical Man Month” -> Programação em pequenas equipes*
 - ▷ *software totalmente amarrado no hardware -> software é um valor independente*

Surgimento espontâneo e acidental do Método Bazar de programação

Transições futuras ou em curso (em relação a 1997)

- O Metabazar (SourceForge): 50.000 bazares de software livre
- Um ecossistema de software livre: O projeto Debian com 9.000 pacotes de software livre, continuamente atualizadas, com alto grau de independência, com as dependências codificadas e administradas pela família apt-get de software

Nasce o Linux em 1991

existem sistemas GNU/Solaris, GNU/HP-UX, GNU/...

existe o Minix, um sistema operacional relativamente aberto usado para fins didáticos (Andrew Tenenbaum)

existe a Internet, já bem difundida no meio acadêmico do mundo inteiro

existe e vai bem a licença GPL e o compilador gcc

em janeiro Linus Torvalds, um estudante de Computação da Universidade de Helsinki, ganha um 386 e inicia a programação de um kernel de Unix para o x86, de força industrial

numa das suas decisões mais importantes escolhe a licença GPL para o kernel

em agosto de 1991 funda o primeiro bazar com uma mensagem para a Usenet

O método Catedral de escrever software

o software é escrito por um pequeno grupo muito coeso

o software é inacessível durante a fase da elaboração

quando aparece o software já tem um alto (ou pelo menos razoável) grau de confiabilidade

a comunidade de contribuintes é altamente hierarquizada, a maior parte das contribuições são feitas por um pequeno grupo de programadores de alta confiança do coordenador, o arquiteto-mor da Catedral

o debugging pode ser feito por uma comunidade mais numerosa, como no caso do TeX e MetaFont

Funcionamento de um Bazar de Software

Alguns princípios importantes

- Objetivo bem focado e capaz de motivar um grupo de interessados bem qualificados para contribuírem
- Trate os seus usuários como co-desenvolvedores
- Se Você tratar os seus beta-testadores como o seu recurso de maior valor então eles corresponderão e tornarão-se em seu recurso mais valioso
- Linus' Law: *“Given enough eyeballs, all bugs are shallow”*
- Release early, release often and listen to your costumers
- Reconheça as boas idéias dos seus usuários

Resumindo: *considere a programação como uma atividade social, numa sociedade espontaneamente hierarquizada*

- “Egoless programming” de Gerald Weinberg (1971)
- “Extreme Programming” de Kent Beck (1995 ?)

O Bazar é um Sistema baseado numa comunidade hierarquizada

O tamanho da comunidade pode variar muito

- desde 3 a 4 desenvolvedores (1?) preparando uma versão inicial de algum sistema
- até um ecossistema de software livre, como o Debian, contendo
 - ▷ *10.000 pacotes de software*
 - ▷ *elaborados por 20.000 bazares de tamanhos muito variados*
 - ▷ *compreendendo um total de um bilhão (?) de linhas de programa fonte*
 - ▷ *distribuídos para uma dezena de arquiteturas de computadores*

o bazar é uma estrutura hierarquizada espontaneamente, de acordo com o grau de competência dos seus membros

- coordenador,
- desenvolvedores,
- documentadores,
- testadores,
- usuários

Uma questão muito interessante, a discutir

A hierarquia do bazar inclui ou não os usuários?

normalmente os usuários são considerados puros beneficiados com o trabalho dos desenvolvedores

eles recebem de presente o software funcionando (“gift culture”)

esta visão contraria a tese de Maurice Mauss, criador da antropologia, com a obra “Essay sur le don”, 1930 (?)

Conclusão

“Talvez no fim a cultura de código aberto irá triunfar não porque a cooperação é moralmente correta e a 'proteção' do software (software 'hoarding') é moralmente errada (assumindo que você acredita na última, o que nem Linus nem eu fazemos), mas simplesmente porque

o mundo do software de código fechado não pode vencer uma corrida evolucionária, disputada com as comunidades de código aberto, que podem disponibilizar horas de dedicação qualificada ordens de grandeza maiores num dado problema.”

Dois conceitos (mais ou menos) nas entrelinhas do trabalho

o bazar é um sistema social que permite uma *evolução* mais segura e mais eficiente do software (evolução=melhora, adequação, capacidade de adaptação)

o mecanismo de interação do bazar tem grandes similaridades com o processo de *peer-review* utilizado no desenvolvimento e documentação das áreas científicas, ao mesmo tempo em que tem enormes discrepâncias com o peer-review

Grandes Expoentes dos dois métodos

Método Catedral

- Ken Thompson e Dennis Ritchie (Unix)
- Donald E. Knuth (TeX, MetaFont)
- Richard Stallman (Projeto GNU, FSF, licença GPL, gcc, emacs, ...)

Método Bazar

- Linus Torvalds, seu inventor e primeiro implementador
- Bruce Parens, um dos primeiros coordenadores do projeto Debian

Alguns desenvolvimentos embasados no CatB

Linux: a Bazaar at the Edge of Chaos, de Ko Kuwabara

Coase's Penguin, or Linux and the Nature of the Firm, de Yochai Benkler

Homesteading the Noosphere e The Magic Cauldron, do próprio Eric Raymond

O movimento open-source

Mais três temas para discussão

Quais as características principais do bazar e quais os pré-requisitos para que ele funcione bem?

Quais as controvérsias e paradoxos levantados pelo bazar em torno da informação aberta e quais os compromissos atuais?

Quais as similaridades e diferenças entre o bazar de software e o bazar de enciclopédia (wikipedia)?