

Software Livre para GIS: Entre o Mito e a Realidade

Gilberto Câmara

Em 2004, a cidade alemã de Munique tomou uma decisão histórica, ao selecionar o sistema operacional Linux para os 14.000 computadores pessoais e servidores usados pela Prefeitura. Apesar do valor do contrato ser modesto pelos padrões alemães (apenas US\$ 35 milhões), o CEO da Microsoft, Steve Ballmer, interrompeu suas férias para visitar Munique e pressionar o prefeito. A Microsoft tomou até uma atitude inusitada: baixou seus preços para igualar-se ao custo do ambiente Linux. Ofereceu seu software essencialmente de graça, ao cobrar apenas os serviços de configuração e instalação. Mesmo assim, os responsáveis pela prefeitura optaram pelo Linux por princípio. O argumento foi que Munique não poderia depender de uma tecnologia proprietária.

A decisão de Munique representa um marco fundamental na escolha entre software livre (ou "software aberto") e software proprietário, pois o preço não foi o fator definitivo para a escolha. A cidade decidiu pela tecnologia Linux por aspectos estratégicos, inclusive levando em conta que a oferta do ambiente Linux foi feita por uma empresa alemã (a SUSE). Afinal, neste mundo globalizado, é mau negócio exportar empregos de alta tecnologia.

O caso de Munique não é único. Ele acontece num momento em que governos no mundo inteiro estão cada vez mais interessados em software livre. Aqui no Brasil, a partir de exemplos como as prefeituras de Porto Alegre e do Recife, o governo federal tem anunciado apoio aos projetos de implantação de software livre. As expectativas do governo se baseiam em quatro hipóteses sobre o software livre: (a) menor custo; (b) independência de tecnologia proprietária; (c) disponibilidade de soluções de software livre eficientes e com boa qualidade; (d) existência de capacidade local de desenvolver soluções adaptadas para o cliente público brasileiro.

A maior parte dos analistas considera que o software livre é desenvolvido por equipes descentralizadas de programadores, distribuídas no mundo inteiro, atuando com altruísmo, e motivados não por interesses comerciais, mas pela pressão dos pares em fazer software de qualidade. A idéia é que um programador cujo código fica à vista de todos tem muito mais cuidado ao desenvolver software e o faz de forma mais confiável. O sistema operacional Linux e do servidor web Apache são veiculados como exemplos a ser imitados por todos os programadores de software livre.

Até aqui, estamos no plano das boas intenções. Mas, afinal, qual é a realidade do software livre? Quem é que faz software livre? Existe software livre de qualidade disponível para todos os gostos? Como se desenvolve a capacidade local de adaptação? Tentando responder a estas questões, o autor fez uma avaliação de 70 produtos de software livre no mercado GIS, a partir dos produtos listados no sítio <http://freegis.org>. Para entender o mercado, classificamos as equipes de desenvolvimento de software em três categorias: (a) *equipes individuais*, com 1 a 3 pessoas, essencialmente da mesma instituição; (b) *redes cooperativas*, com times de 15 ou mais, operando de forma distribuída e trabalhando muitas vezes fora do expediente normal, como no caso do Linux; (c) *times corporativos*, com equipes de 5-15 programadores, trabalhando para uma instituição privada ou pública, que por decisão própria decidiu desenvolver software livre. Para cada produto, atribuímos notas de 1 a 5 para as características de maturidade, suporte e funcionalidade. Os resultados, de certa forma surpreendentes, estão na Tabela a seguir.

GRAUS DE MATURIDADE, SUPORTE E FUNCIONALIDADE DE SOFTWARE LIVRE PARA GIS

	Total	Maturidade	Suporte	Funcionalidade
Equipes Individuais	37 (53%)	2.3	1.7	1.8
Redes cooperativas	4 (6%)	3.7	3.7	3.7
Corporações	29 (41%)	3.2	3.1	3.0

(valores Médios numa escala de 1-5, onde 1 – pior e 5 – melhor)

Os resultados indicam que apenas 6% de todos os produtos de software livre para GIS são desenvolvidos por redes cooperativas. Embora surpreendentes à primeira vista, estes resultados têm uma explicação lógica. Poucos são os produtos de informática que podem ser desenvolvidos de forma inteiramente descentralizada. É preciso que a estrutura do software seja muito modular para que seja possível distribuir tarefas, sem se preocupar muito com os custos de integração. Exemplos são: (a) GRASS(<http://grass.itc.it/>), conjunto de funções de análise espacial baseados em arquivos matriciais; (b) R (<http://www.r-project.org/>), um pacote de programas estatísticos, que inclui técnicas de estatística espacial. Outro aspecto é psicológico: não é fácil para programadores de culturas diferentes concordarem em trabalhar no mesmo software, principalmente se ele está ainda num estágio inicial. São poucos os produtos que conseguem vencer esta “barreira invisível” e empolgar corações e mentes no mundo inteiro.

Os softwares desenvolvidos por indivíduos ou times pequenos são mais da metade dos produtos consultados, com duas grandes variedades. Uma minoria são produtos que fazem bem apenas uma tarefa específica, como o “shapelib” (<http://gdal.velocet.ca/projects/shapelib/>), biblioteca para ler e escrever arquivos em formato ArcView, ou o Gstat (<http://www.gstat.org/>), software de geoestatística. Na maior parte, trata-se de idéias pessoais, colocadas na Web na esperança de conquistar adeptos, mas cuja funcionalidade é ainda muito limitada.

Os produtos institucionais são quase metade (41%) do software livre para GIS e apresentam qualidade muito superior aos softwares individuais. Isto decorre do fato que software de alta complexidade precisa de muita comunicação entre as equipes e está fora do alcance de um programador individual. Também mostra que a grande força do desenvolvimento do software livre não são *hackers* altruístas que trabalham de madrugada, mas programadores profissionais cujo emprego é produzir software livre. Este é o caso de produtos significativos na para GIS: (a) o PostgreSQL (<http://www.postgresql.com>) e seu complemento PostGIS (<http://postgis.refractions.net>), para criação de bancos de dados geográficos; (b) o TerraVision, para visualização 3D avançada na Internet (<http://www.tvgeo.com/>); (c) o GeoVista, conjunto de técnicas inovadoras de visualização geográfica desenvolvido na Penn State University (<http://www.geovista.psu.edu>); (d) o TerraLib (<http://www.terralib.org>), biblioteca para construção de aplicativos geográficos desenvolvida pelo INPE, TecGraf/PUC-RIO e Funcate.

Quais são as indicações de política pública que podemos tirar deste levantamento? Primeiro, há muito pouco software livre de qualidade baseado na “geração espontânea”. Programas de software livre com bom desempenho são essencialmente produtos institucionais. Não basta decretar: “queremos usar software livre”, pois pode não existir o software que queremos. Se a política adotada pelo governo brasileiro consistir apenas num conjunto de decretos e resoluções recomendando ou até obrigando o uso de software livre por instituições públicas, estaremos construindo uma grande frustração nacional. Portanto, se o governo brasileiro quiser efetivamente estabelecer uma política de uso de software livre, será preciso estabelecer programas nacionais de apoio ao desenvolvimento e manutenção de software livre, e de capacitação de empresas para adaptação deste software para os clientes locais. Sem isto, nosso discurso de independência tecnológica não se sustentará. Como dizia um filósofo alemão: “A História sozinha nada faz; ela não possui riquezas, nem participa de batalhas. São os homens que, por sua escolha, fazem tudo isso e a constroem.”