

REVISTA

# espírito livre

LIBERDADE E  
INFORMAÇÃO

<http://revista.espiritolivre.org> | #001 | Abril 2009



## PADRÃO ABERTO

Você sabe o que é um?



## EDUCAÇÃO

O Linux em sala de aula



## BRIGA BOA

O julgamento do  
The Pirate Bay

# COMPUTAÇÃO EM NUVEM

De aplicativos a sistemas  
operacionais, tudo vai pra web

## ENTREVISTA

Pau Garcia-Milà fala de  
sua criação: o eyeOS

## LINGUAGEM LUA

É da terra *brasilis* esta  
poderosa linguagem de  
programação

## É LEGAL

Software Livre faz  
sucesso nos tribunais  
brasileiros



## Atribuição-Uso Não-Comercial-Vedada a Criação de Obras Derivadas 3.0 Unported

Você pode:



copiar, distribuir, exibir e executar a obra

Sob as seguintes condições:



**Atribuição.** Você deve dar crédito ao autor original, da forma especificada pelo autor ou licenciante.



**Uso Não-Comercial.** Você não pode utilizar esta obra com finalidades comerciais.



**Vedada a Criação de Obras Derivadas.** Você não pode alterar, transformar ou criar outra obra com base nesta.

- Para cada novo uso ou distribuição, você deve deixar claro para outros os termos da licença desta obra.
- Qualquer uma destas condições podem ser renunciadas, desde que Você obtenha permissão do autor.
- Nothing in this license impairs or restricts the author's moral rights.

Termo de exoneração de responsabilidade

Qualquer direito de uso legítimo (ou "fair use") concedido por lei, ou qualquer outro direito protegido pela legislação local, não são em hipótese alguma afetados pelo disposto acima.

Este é um sumário para leigos da Licença Jurídica (na íntegra).

# BOAS VINDAS

Sonho. Uma palavra que para muitos representa algo impossível e para outros se reporta a esperança em algo, um desejo, um pensamento. No caso da Revista Espírito Livre não podia ser diferente. A revista nasceu de uma idéia pessoal já antiga, mas que amadureceu durante o feriado de carnaval deste ano de 2009 e percorreu longos passos até chegar a edição de estréia. A revista tem como alicerce a Iniciativa Espírito Livre, uma proposta que busca reunir colaboradores, técnicos, profissionais liberais, entusiastas, estudantes, empresários e funcionários públicos em atividades, como em eventos afim de estreitar os laços destes com iniciativas livres, entre elas o próprio software livre.

Muito debate e discussão até termos o formato de revista que você está lendo neste momento, e o processo de produção foi caminhando, os responsáveis por matérias de um lado, eu de outro e Hélio Ferreira na outra ponta, tornando todas essas idéias realidade, algo legível. Aliás, reunir um time tão maduro e respeitado não foi tarefa fácil, muitos tinham compromissos, outros simplesmente dispensaram o convite de participar. Todos temos compromissos particulares e não poderia ser diferente com os colaboradores que assumiram a empreitada. A estes, "tiro meu chapéu", acredito que os leitores também tirarão...

Nas próximas páginas vocês terão muita informação de qualidade e que passou por um crivo grande até chegar aqui. A entrevista de estréia é com Pau Milà-Garcia [Barcelona/Espanha], criador do eyeOS, considerado por muitos o mais bem sucedido sistema operacional web. Vários nomes conhecidos dentro e fora do Brasil estarão passando aqui pela revista. Este é só o começo...

Além da revista em formato PDF, já está no ar o site da revista [<http://revista.espiritolivre.org>], que trará trechos das recentes e futuras matérias. Estejam ligados no site até porque as promoções pipocarão lá também. É só ficar atento.

Bom, é isso. Estarei com vocês nas próximas edições a frente dos trabalhos e sintam-se a vontade para opinar e dizer o que acharam de nossa revista.

A todos vocês leitores, a Revista Espírito Livre dá as boas vindas!

Nos vemos por aí... e vamo que vamo! 

**João Fernando Costa Júnior**  
**Editor**



## EXPEDIENTE

### Diretor Geral

João Fernando Costa Júnior

### Editor

João Fernando Costa Júnior

### Revisão

Marcelo Tonieto

### Arte e Diagramação

João Fernando Costa Júnior

Hélio S. Ferreira

### Capa

Hélio S. Ferreira

### Contribuiram nesta edição

Alexandre Oliva

Cárlisson Galdino

Cezar Taurion

David Ferreira

Edgard Costa

Evaldo Junior

Filipe Saraiva

Giselle Cristina S. D. Costa

Jomar Silva

Kárlisson Bezerra

Lázaro Reinã

Orlando Lopes

Pau Garcia-Milà

Paulino Michellazo

Roberto Salomon

Rodrigo Leão

Sinara Duarte

### Contato

[revista@espiritolivre.org](mailto:revista@espiritolivre.org)

# SUMÁRIO

## CAPA

### **26 Aspectos estratégicos**

O espaço virtual não tem dono nem fronteiras

### **29 Entrevista**

Pau Garcia-Milà fala do sistema operacional eyeOS

**PÁG. 22**

## COLUMNAS

### **09 Síndrome de Peter Pan Digital**

Sonhe, voe, mas não deixe de crescer

### **11 O movimento foi traído**

Para coçar e traer basta começar

### **13 Para quem vai tirar a LPI**

Dicas para quem vai encarar a prova

### **15 Uma noite na Taverna Silício**

Esse papo vai longe...

### **18 O que aprendemos até agora**

É preciso aprender a colaborar melhor

### **20 SL e o sistema de leis**

As leis existentes não são aplicadas

## Computação em nuvem

Muito mais que um massivo conjunto de computadores



**56 AGENDA**



**06 NOTÍCIAS**



---

## DESENVOLVIMENTO

---

- 33** **Lua**  
Linguagem de programação poderosa
- 37** **Crie software livre**  
Dicas pra quem está iniciando

---

## TECNOLOGIA

---

- 41** **Padrões abertos**  
Não leve gato por lebre
- 46** **The Pirate Bay**  
O compartilhamento digital em debate

---

## EDUCAÇÃO

---

- 49** **Linux nas escolas**  
Formando uma geração com poder de escolha

---

## HUMOR

---

- 52** **Lino e Wino**  
Nerdson não vai à escola

---

## EVENTOS

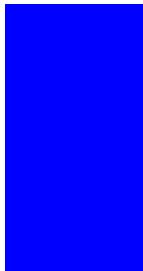
---

- 53** **Document Freedom Day**  
Veja o que rolou em Vitória/ES

# NOTÍCIAS

Por João Fernando, Giselle Cristina e Lázaro Reinã

**Lei anti-pirataria é derrubada na França por ausência de deputados**



**Mozilla Labs Concept Series Design Challenge Spring 2009**



No último dia 9, a Assembléia Nacional da França esteve votando contra o polêmico projeto da "Lei de Proteção da Criação na Internet", que tem como autores partidários do governo de Sarkozy. É previsto no texto as já famosas punições a internautas que fizerem downloads de conteúdos protegidos por direito autoral.

"Esta lei é o resultado de um acordo alcançado entre artistas, produtores e empresas de telecomunicações", diz Sarkozy, sugerindo que pretende prosseguir com o projeto de lei. O ocorrido agradou também aos internautas espanhóis, uma vez que tal modelo do projeto apoiado por Sarkozy já foi escolhido pela Indústria Cultural e Ministério da Cultura espanhol afim de se tornar realidade também em solo espanhol.

Um desafio aos audaciosos, criativos e afins, o Mozilla Labs Concept Series Design Challenge Spring 2009 é um evento que promete reorganizar tanto os navegadores quanto a própria Web. Este é seu primeiro ano e propõe a mentores e estudantes, ideias e/ou projetos a partir do zero, do nada.

A linha de desenvolvimento sugerida aos designers é que se produza um navegador sem interface, sem janelas, onde apenas existe a web. Alguns conceitos já foram apresentados e poder ser vistos no seguinte link <http://labs.mozilla.com/projects/concept-series>.

**Lançado CDLivre 4.0**

# CDLivre

Versão 3.0.0

Jogos  
Multimídia  
Produtividade  
Desenvolvimento



Design  
Internet  
Utilitários  
Wallpapers



Festival Latinoamericano de Instalação de Software Livre

Sua Liberdade de Escolha! - Use Software Livre!

No dia 29 de março foi lançado o CDLivre 4.0 uma iniciativa independente, que disponibiliza diversos softwares livres no intento de além de disseminar a filosofia livre, desmistificar ou pelo menos amenizar o impacto do primeiro contato com o uso de um S.O livre até porque a maioria dos softwares contidos no CD, senão todos, são aplicativos facilmente encontrado nas maiores distribuições Linux.

No CD são encontrados softwares tais como: Firefox, Pidgin, Xchat e Amsn, TrueDownloader, 7zip, Songbird, etc.

Então não perca tempo e acesse já <http://cdlivre.dukitan.com> para baixar o CD.

### Gastos com TI cairão 3,8 % este ano, diz Gartner



Devido ao presente cenário econômico mundial, é quase inevitável que se faça especulações quanto ao mercado de TI, um dos que mais movimentam a economia mundial. Afinal, nada funciona hoje sem que haja a influência da informática. O Instituto Gartner preve uma forte queda em comparação com 2008. A redução afetará os quatro principais segmentos do setor: hardware, software, serviços e telecomunicações.

A retomada dos investimentos no setor só acontecerá quando os mercados financeiros se estabilizarem. Enquanto isso, segundo a companhia, tanto os consumidores quanto as empresas irão migrar para produtos mais baratos e estender a vida útil dos equipamentos que já têm.

Essa última informação pode passar despercebida por alguns, porém, quando se trabalha com Software Livre, as expressões "migrar" e "produtos mais baratos", significam muito. Definitivamente esse é o momento do Software Livre.

### Senador Azeredo aumenta a pressão para a aprovação de seu projeto



Se você ainda não sabe a respeito do projeto senador Eduardo Azeredo é bom tomar nota pois a liberdade à informação em nosso país pode estar em risco. O projeto de lei do Senador Azeredo é polêmico e impõe o fim da comunicação anônima na internet, além de criminalizar práticas cotidianas na rede. Abre espaço para atacar as redes P2P, como tem ocorrido em todo o mundo (veja o exemplo do julgamento do Pirate Bay nesta edição).

Caso queira se juntar à luta pela privacidade na internet, envie um e-mail para o deputado do seu Estado pedindo reforços. Procure no site da Câmara o e-mail do deputado do seu Estado: <http://www2.camara.gov.br/canalinteracao/falededeputado>. Existe ainda uma petição em resposta ao projeto do senador Azeredo. Segue o link : <http://www.petitiononline.com/veto2008/petition.html>.

### Próxima versão do Firefox contará com abertura de páginas em 50 milissegundos



A Mozilla, responsável pelo famoso navegador Firefox divulgou algumas informações sobre o lançamento da próxima versão de seu browser: O nome de código escolhido é Namoroka e será lançado no início de 2010. Ainda não se sabe se ele vai se chamar Firefox 3.6. ou Firefox 4. Segundo a PC Pro, a Mozilla pretende apostar pesado na capacidade de resposta desta nova versão.

Algumas metas já foram fixadas, entre elas a capacidade de iniciar o programa ou abrir tabs em até 50 milissegundos, atualização de localizações e reprodução multimídia. Até à data, a Mozilla prefere se referir a esta futura versão como Firefox.next.

Está previsto ainda que as ferramentas Ubiquity e Prism sejam integradas a esta nova versão.

### Lemote YeeLoong Open Net-book: Um open notebook



A popularidade dos notebooks cresce e com ela as novidades vão surgindo a cada semana. Esse ai é o [Lemote YeeLoong](#). Vem recheado com Debian GNU/Linux + Lemote Education Suite. Sim, o primeiro notebook totalmente "aberto". Todos os seus softwares possuem código livre. Nenhum partezinha de código que seja proprietário. Enfim, um "open notebook"...

Nada revolucionário em termos de hardware e configuração, mas o preço promete ser bem competitivo. Em breve o novo brinquedinho estará nas lojas. O aparelho vem com duas opções de armazenamento: disco rígido (que pode ter 160GB) ou SSD (4 ou 8GB). Ele vem com uma tela 8.9" TFT LCD com 1024x600 de resolução. Quanto a memória RAM tem opções com 512MB e 1GB, enfim atentendo a todos os gostos. Traz ainda leitor de cartões e webcam de 300K.

Visite o site oficial do [Lemote YeeLoong](#) para saber maiores detalhes.

### Mandriva ajuda a 'portar' k3b para Qt4



O k3b é, provavelmente, a melhor aplicação com interface gráfica para gravar CDs e DVDs em GNU/Linux e nas diversas BSDs. Apesar de ser uma aplicação muito popular, ela ainda não havia sido portada para o toolkit Qt4, o mesmo usado pelo KDE 4. Felizmente, a Mandriva decidiu dar uma ajuda com mão-de-obra, passando a trabalhar em conjunto com Sebastian Trueg para que o processo seja acelerado, como refere no seu blog oficial.

O port do k3b pra Qt4 deverá ser concluído a ainda a tempo do lançamento da próxima versão da Mandriva Spring. Além de se integrar muito melhor com o novo KDE, os usuários de Windows também deverão poder utilizar esta poderosa aplicação, haja vista que o Qt4 é conhecido por ser multi-plataforma e ter versão para Windows.

Ponto para Mandriva!

### Enorme doação alemã marca evolução do Wikipédia



**WIKIPEDIA**  
*The Free Encyclopedia*

Em dezembro foi discutido um acordo entre a Wikipédia Alemã e o Arquivo Federal Germânico encorajado a doar 100.000 imagens à Wikimedia sob uma licença Creative Commons Attribution-ShareAlike (CC-BY-SA). Naquela ocasião, a maior doação de imagens de todos os tempos para a Wikipédia, e a maior na história do movimento pela cultura livre. Agora a Wikipédia Alemã está atingindo acordo similar com o Estado Saxão e Biblioteca Universitária, que vão doar 250.000 (duzentos e cinquenta mil) imagens à Wikipédia sob uma licença CC-BY-SA. Coincidência ou não, a Microsoft anunciou que irá descontinuar sua enciclopédia Encarta. 

**Quer comentar sobre algo desta primeira edição? O que gostou? O que não gostou? Participe! Envie seu comentário e/ou notícia para [revista@espiritolivre.org](mailto:revista@espiritolivre.org).**

# Síndrome de Peter Pan Digital



Por Alexandre Oliva

Minha filha entrou no último ano da pré-escola que frequenta. Já está sendo preparada pra grande transição no final do ano. Em breve vai conseguir ler por si mesma. Ao contrário de alguns usuários de software, está animadíssima pra dar esse passo em direção à independência, pra não mais precisar de uma Wendy pra ler os contos que ela adora. Também, não há nenhum Peter Pan insistindo pra ela ir pra Terra do Nunca, pra jamais crescer.

Liberdade, independência e soberania são conceitos relacionados, muitas vezes mal interpretados. Não significam poder fazer qualquer coisa que se queira. Significam não estar dominado, subjugado, controlado. Significam poder fazer suas próprias escolhas, limitado por não mais que sua própria natureza e o respeito ao próximo. Afinal, a liberdade de um termina onde começa a do outro.

Software é Livre para um usuário se ele tem as liberdades

des de (0) executar o programa para qualquer propósito, (1) estudar seu código fonte e adaptá-lo para que se comporte como desejado, (2) copiá-lo e distribuí-lo da forma que foi recebido e (3) melhorá-lo e distribuir as modificações. Havendo restrições substanciais a qualquer dessas liberdades, o Software não é Livre, assim como o usuário. Maneiras de impor restrições há muitas: negação de acesso ao código fonte, verificação de origem no hardware ou em software de baixo nível, direito autoral, contratos, ordens judiciais, por aí vai. O diálogo entre usuário e os vilões da história (são muitos) é mais ou menos assim:

- Posso instalar o programa no meu outro computador?
- Nunca! Você assinou um pacto com o sangue do seu mouse!
- Posso corrigir esse erro que está me incomodando?
- Nunca! Quem decide o que é certo no programa sou eu!

- Posso fazer uma cópia pro meu amigo?

- Nunca! Vai virar tripulante do Capitão Gancho!

- Posso liberar esse programa que eu escrevi?

- Nunca! A patente de capitão é só minha! Já pra prancha!

- Posso estudar como o programa funciona?

- Nunca! Enquanto estiver nesta terra, nunca vai crescer!

Curioso é que tem gente que parece que não só acha isso normal; até diz que gosta assim e luta pela suposta liberdade de aceitar esses abusos. De fato, se essas escolhas infelizes não tivessem repercussão negativa sobre ninguém mais, se prejudicassem somente quem se joga na armadilha, seria uma liberdade legítima.

Mas quando o usuário aceita que outros tomem decisões sobre sua vida digital, colocando nas mãos deles o controle sobre o que fazem os programas que usa, além de

sacrificar a própria liberdade ainda oferece vantagens aos que impõem as restrições: pagamento pelo "privilegio", receita indireta proveniente de recomendações, popularidade, efeitos de rede, entre outros.

Isso realimenta o processo: o agente agressor, percebendo vantagens, conclui que o comportamento é favorável e bem aceito e mantém ou até aumenta as restrições. O usuário, já subjugado, por vezes alienado, encontra-se em posição de fácil controle pelo agressor.

O pior é que, com o poder conferido pelas vítimas anteriores, o agressor faz novas vítimas. Essa repercussão prejudicial sobre os demais é que faz com que a decisão de realimentar o processo ultrapasse os limites da liberdade individual.

Mas o usuário que não quer crescer, que quer deixar nas mãos de outros as decisões que afetam diretamente sua própria vida digital (e a nossa), também não aceita a responsabilidade social sobre como seus atos afetam a mim, você e os demais. Ao contrário, clama por uma liberdade absoluta, sem limites provenientes da preocupação com o próximo, da ética: é a Síndrome de Peter Pan Digital.

Tal qual Peter Pan, insistem que outros trilhem o mesmo caminho dos meninos perdidos, tornando-os todos de-

## “ ” O movimento Software Livre é um movimento social, não técnico.

Alexandre Oliva

pendentes de alguma Wendy que lhes leia porque não conseguem ler. Decidem continuar não conseguindo. Não querem crescer, para preservar sua ilusão de liberdade absoluta.

O movimento Software Livre é um movimento social, não técnico. Sua filosofia está fundada no princípio moral universal da reciprocidade: trate os outros como gostaria de ser tratado. Aconselhe seus amigos a que, ao invés de prejudicar ao próximo, desrespeitando suas liberdades e/ou ajudando outros a desrespeitá-las, ajudem-no a crescer consigo.

O pozinho mágico da Tinker Bell (no meu tempo era Sininho) é pó de liberdade: é com ele e com pensamentos e sonhos felizes que a gente pode voar, crescer e alcançar o que quiser, sem prejudicar ninguém. Vai uma pitadinha de liberdade aí?



Copyright 2009 Alexandre Oliva

Cópia literal, distribuição e publicação da íntegra deste artigo são permitidas em qualquer meio, em todo o mundo, desde que sejam preservadas a nota de copyright, a URL oficial do documento e esta nota de permissão.

<http://www.fsfla.org/blogs/lxo/pub/peter-pan-digital>



**ALEXANDRE OLIVA** é engenheiro de Computação e Mestre em Ciências da Computação. Usuário, desenvolvedor, evangelizador de Software Livre e contribuidor do projeto GNU desde a década de 90. Engenheiro de Compiladores na Red Hat desde fevereiro de 2000. Co-fundador e conselheiro da FSFLA - Fundação Software Livre América Latina.



Sigurd Decroos - www.sxc.hu

# Você traiu o movimento, véio!

Por Paulino Michellazo

Bendito dia! Final de FISL [1], todo mundo cansado e chega aquele nunca visto com o discurso de traição. É verdade, trair e coçar é só começar. E se é pra começar, nenhum lugar melhor que o mais famoso evento da comunidade brasileira de software livre.

— Eu traí? Por quê?

— Você traiu o movimento, véio. Você usa Mac!

— Hááá, é por isso? Mas o Mac roda Linux, roda OpenOffice, roda...

— NÃO IMPORTA!, você é impuro!

Vou saindo de fininho, escada rolante abaixo para não ser açoitado pelo zebu vitima-

do pela Síndrome Larms [2]. Infelizmente as pernas de meu pinguim são curtas e não dão conta de desviar dos chifres do GNU com olhos vermelhos pelo sangue da vingança. Com a barulheira chegam mais vítimas da mesma síndrome (será que a ANVISA não cuidou do problema?) que se juntam ao primeiro na caça as bruxas, ou melhor, ao pinguim.

Tento argumentar que não é o hardware que faz o pensamento mas não tem conversa. Querem a todo custo (agora uma manada toda) pegar o Macbook, depená-lo e deixar somente o disco rígido. Pior; sem nenhum dado pois dizem estar infectado pelo vírus

do hardware proprietário. E vírus agora também leva a denominação de proprietário? Na mesma hora tiro o celular do bolso para pedir auxílio ao IBAMA e não ser pisoteado quando vêem mais um símbolo do pecado em minhas mãos; um iPhone preto reluzente (ou seria preto piano?), peça totalmente herege.

— Um iPhone? Seu vendido! Vamos excomungá-lo!

— Calma! É simplesmente um aparelho.

— Não é não! É obra do diabo dos empregos (Jobs).

E vira alvoroço! É mouse para um lado, cabo de força para outro, roteador, fone e até

mesmo uma placa de 286 apareceu no meio. A muvuca vai se formando e chegam outros de várias tribos para participar do julgamento e execução. Até a fogueira armaram com CD's da Microsoft e manuais de Oracle. Jogam álcool (porque gasolina é dos porcos imperialistas) e ameaçam a cremação diante das beldades da PUC. Neste momento, a salvação. O velho cachorro louco aparece trazendo sob seus braços seu inseparável laptop IBM e profetiza: "aqueles que nunca foram proprietários que atirem a primeira pedra."

A multidão se contém (para alegria de poucos e tristeza de muitos) e o sermão continua: "comunidades devem ser democráticas onde todos tem

direito de opinar e decidir o que é melhor para si mesmos. Precisamos conter atitudes que mancham a imagem de todos aqueles que acreditam no software livre. Não são máquinas ou equipamentos que modificam as pessoas, mas sim suas atitudes de tolerância e compreensão diante dos outros."

No meio da patota alguém mostra uma foto autografada e adquirida por cinco reais para dizer: "este aqui nunca foi proprietário!" E outro retruca: "nunca foi e também nunca toma banho!" Pronto, lá está a pancadaria armada novamente.

E eu? Escorrego pela lateral agradecendo o cachorrão:

thank's man! Today, you're God! 

[1] Fórum Internacional de Software Livre

[2] Síndrome Larms — A la RMS



**PAULINO MICHELAZZO** é diretor da Fábrica Livre ([www.fabricalivre.com.br](http://www.fabricalivre.com.br)), empresa especializada em soluções para Internet com ferramentas de gestão livres. Foi diretor mundial da Mambo Foundation, System Develop Specialist da ONU no Timor Leste. É instrutor de CMS's (Drupal, Joomla e Magento). Escreve regularmente para diversos canais na Internet e publicações técnicas no Brasil e em Portugal.



**fisl10**  
10º Fórum Internacional  
**Software Livre**  
A tecnologia que liberta

**Edição Especial**

24 a 27 de Junho de 2009 - Centro de Eventos PUCRS  
Porto Alegre/RS - Brasil

Maiores informações em:  
**[www.fisl.org.br](http://www.fisl.org.br)**





# Quer tirar a LPI?

Não se apresse, estude e use bastante o sistema antes de encarar a prova.

Por Evaldo Júnior

Você quer tirar a certificação LPI? Está estudando bastante para isso? Já conhece bem o sistema Linux e as ferramentas GNU? Faz simulados e mais simulados? Comprou livros sobre o assunto? Pois eu tenho uma grande novidade para você! Estamos no mesmo barco! E mais, tenho umas dicas também! Mas primeiro vamos voltar ao passado.

É engraçado pensar que quando iniciei no mundo do software livre eu só pensava na LPI, eu lia artigos, estudava pelo Guia Foca Linux, treinava no computador e sempre me

sentia despreparado. Quando eu fazia algum simulado eu não ia muito bem, acertava cerca de 30% das questões. Pensei várias vezes em entrar para algum curso preparatório para a LPI, mas os preços, na época, acabaram me desestimulando.

Mas foi ai que a maré começou a virar à meu favor. Acabei conseguindo dois estágios para trabalhar com servidores Gnu/Linux, principalmente Debian, depois disso ainda trabalhei mais 2 anos administrando uma pequena rede de escritório e desenvolvendo um

sistema em PHP. Como vocês podem imaginar eu acabei resolvendo vários problemas reais, ou seja, as soluções precisavam funcionar, não eram provas de certificação que, se eu não passasse, poderia fazer novamente.

Durante este período eu acabei deixando essa tal certificação de lado, não que ela não me interessasse mais, mas o sistema em si e a solução de problemas eram muito mais interessantes. Aprendi, com esforço e paciência, a montar um servidor web, mysql, proxy, samba, nfs e

“ Não guarde o que você aprendeu. Passe para frente o conhecimento.”

Evaldo Junior

aprendi bastante também sobre a administração de pacotes Debian.

Fiz outras coisas que me ajudaram muito a evoluir. Me integrei à comunidade de software livre, ajudei a organizar duas edições do Flisol em Santos/SP, onde conheci um pessoal bem legal. Participei, e ainda participo, do fórum Debian, tentando ajudar o pessoal lá, parece que não, mas isso é um belo de um treinamento. Desenvolvi inúmeros scripts para automatizar várias tarefas do meu cotidiano. Desenvolvi um front end para o eSpeak, o GeSpeak. Penei para configurar a placa wireless do meu notebook, mas hoje ela já é suportada. E fiz outras tantas coisas que nem lembro, mas que me ajudaram a aprender mais e entender como o GNU/Linux funciona.

Agora voltamos ao presente... Há cerca de duas semanas resolvi voltar à estudar para a LPI, acabei estudando um dia só, aliás, algumas horas, com alguns amigos, e esta

semana eu recebi, de um desses amigos, um programinha que é um simulado da LPI. Resolvi testar, rodei o programa, escolhi a prova 101 e comecei a responder as perguntas. E qual foi o resultado? Acertei quase todas as questões, só errei mesmo as referentes ao RPM, já que ainda não usei uma distribuição descendente do Red Hat.

Desta história toda eu consegui concluir algumas coisas, talvez bem óbvias, mas interessantes.

Primeiro, a experiência prática ajuda muito, por isso não fique só na teoria, arrume problemas para resolver, que tal montar um pequeno servidor em casa? Pegue um computador mais antigo e faça uma transformação completa! Ele pode ter um proxy que impede seu irmão mais novo de ver sites adultos, ele pode ser um servidor de arquivos ou um repositório subversion, e que tal um servidor de impressão? As possibilidades são muitas, explore isso.

Segundo, não guarde o que você aprendeu, faça diferente, passe para frente o conhecimento, quando você ensina alguém você está aprendendo também. Monte um blog, participe de fóruns, dê palestras, vá a eventos. Essas atividades fazem você aprender muito, afinal você não vai fazer um texto em um blog sobre algo que você não conhece bem, certo?

Terceiro, aprenda mesmo, não se limite a seguir tutoriais, faça mais, leia os manuais e entenda como as coisas realmente funcionam e não simplesmente como funcionaram para alguém.

Então é isso, estude, resolva problemas e ganhe experiência, você vai ver que afinal de contas a prova nem é tão complicada. Boa sorte! 

## **Maiores informações:**

### **LPI Internacional:**

<http://www.lpi.org>

### **LPI Brasil:**

<http://www.lpi.org.br>

### **Blog do InFog:**

<http://infog.casoft.info>



**EVALDO JÚNIOR**  
(InFog) é formado pela Fatec em processamento de dados e atualmente é desenvolvedor, administrador de sistemas, membro da comunidade de software livre e também dá aulas no projeto Hackerteen.



Claudio Pozas - www.enworld.org

# Uma noite na Taverna Silício

Por Carlisson Gaudino

Nosso mundo  
não tem mais  
fronteiras...

Saudações, amigo leitor da Revista Espírito Livre. Eu sou Cárlisson Galdino e este vai ser meu palco pelos próximos meses. Aqui vocês verão um pouco do que geralmente não se vê em revistas voltadas à tecnologia.

Talvez você não me conheça, mas... Tá, tá bem, prova-

velmente você não me conhece. Talvez você conheça o Cordel do Software Livre, que publiquei há vários meses através da Internet. E talvez não. Sou o Bardo autor do cordel, caso o conheça (e caso não o conheça também, de certo).

A proposta deste modes-

to espaço na Revista Espírito Livre é falar sobre Tecnologia e Software Livre usando Arte, ou falar sobre Arte e Software Livre usando Tecnologia, ou o contrário. Enfim, qualquer coisa do tipo.

Nosso mundo não tem mais fronteiras. Como disse Oswaldo Montenegro recentemente em seu blog, "voando de avião, não vemos as linhas separando estados e países. Tudo besteira!" Este é nosso mundo hoje.

Cordel... Falar em não termos fronteiras me lembrou que talvez você que acompanha estas linhas nem saiba do que estou falando, já que cordel é tipicamente nordestino. Cordel é uma poesia enorme colocada num livrinho e pendurada em cordões para a venda. Isso antigamente, já que hoje é possível vender até mesmo pela Internet (ainda não cheguei a isso porque não acho que faça sentido para alguém pagar R\$ 1,50 em um produto e precisar pagar quase R\$ 30,00 de frete). Afinal, cordéis típicos não têm menos que 15, nem mais que 60 estrofes (aqueles bloquinhos de texto)...

É, escrevo poesia, contos, cordel (que é poesia), romances... Não desenho, mas pra quê? Pra fazer isso já temos outro Karlisson.

Quanto a este Cárlisson que vos fala, sou formado em Ciência da Computação pela UFAL (onde hoje trabalho) e

fiz parte da primeira turma do curso de Pós-Graduação em Software Livre pela UFLA. Já tive vários projetos, que por uma razão ou por outra não deram certo: IaraJS, Enciclopédia Omega, Losango, Academia Código Livre... Também já ajudei alguns projetos. Sou membro da Academia Arapiraquense de Letras e Artes. Ajudei meu xará em algumas paródias-charges que ele fez no Nerdson (acho que em todas até o momento, mas sempre foi apenas ajuda, na forma de uma ou outra correção métrica/rítmica ou alguma sugestão. O maior trabalho nas paródias foi sempre dele mesmo).

Neste espaço você vai encontrar poesias de temas raros, didáticas, informativas ou apenas entretenimento geek / nerd / hacker. Também contos ou causos...

"Depois de baixar um arquivo no antigo Netscape até concluir 140% do download, a barata parou um pouco para tocar gaita enquanto recompilava o GCC. (Na reversal russa o código-fonte que você escreve é que compila o GCC?) O que viu na tela lhe trouxe a mais pura insanidade. E ela voou pela janela para bem longe e até hoje não há notícias do pobre inseto. Na tela, apenas aquelas palavras assustadoras. Palavras capazes de gelar até mesmo o coração do mais corajoso desenvolvedor: Kernel Panic."

Enfim, espero que vocês gostem da proposta.

### Uma Noite na Taverna Silício

Num vale entre montanhas inacessíveis, numa pequena vila ao redor do verde manto de árvores que recobre aquelas montanhas, muita coisa acontece. Um lugar muito pequeno na imensidão do mundo, mas com uma imensidão de coisas para contar. Um lugar importante.

São sete horas da noite na Taverna do Silício quando entra um homem de terno colorido e chapéu. Ele se senta ao balcão. Apreensivo, olha para os lados, com seu queixo pontiagudo. Tira do bolso o relógio de ouro preso à corrente no cinto e confere as horas.

O bar está quase vazio a essa hora. Pelo menos vazio das pessoas que ele conhece, com quem lidava no passado. Sempre foi encrenqueiro, mas raramente perdia uma disputa. Nem que precisasse pagar o xerife, o barman e juntar amigos para isso com sua imensa fortuna. Não era à toa que ele, mesmo em um lugar tão escondido, era um dos homens mais ricos de todo o planeta.

- Me traz uma coca-cola e um whisky xyz.

- Tá aqui, meu patrão.

- A vida tá uma droga, Jonas!

- Vão mal os negócios? A

rede de suporte para escritórios e a fábrica de janelas estão com problemas?

- De jeito algum! Vai tudo muito bem!

- Então o quê?

- Eles estão deixando de gostar de mim, aos poucos...

- Como assim?

- Ainda lembro daquele marinheiro, lembra? Acho que se chamava Netune Skaeyph.

- Lembro sim. Era antigo na casa. Mas o senhor me desculpe, mas o senhor judiou muito dele.

- É, eu sei... Mas ele estava invadindo meu território e eu tinha que tomar uma providência.

- Sei como é, patrão. Até hoje ninguém sabe quem incendiou o barco dele, não é? Foi uma confusão tão grande que até o xerife foi expulso e botaram outro. E ele dizia que era você que tinha queimado, vejam só!

- Intriga, Jonas. Intriga...

- É sim. E ouvi dizer que ele está muito bem estabelecido lá na Libéria. Que agora adotou outro nome, acho que foi Fênix ou Minotauro, algo assim...

- É, estou sabendo. Na

verdade esses são miseráveis filhos dele, que hoje têm outros nomes, mas não vem ao caso. O caso é que estão todos indo embora, Jonas! Todos!

- Libéria... O que tem de mais essa terra?

- É, Jonas, não sei. E tem muitos clientes querendo que eu me mude pra lá.

- E o senhor vai, senhor Michel?

- Não! Já mandei uns funcionários lá, mas querem que eu vá. E eu sei que se eu for vou terminar destruindo tudo o que fiz até hoje...

- Então fica por aqui mesmo! A Taverna Silício do Vale estará sempre aberta para o senhor.

- É, mas todos estão indo para lá... Já foi o Sam e com seu filho, o Breno Alves, lá ele abriu uma empresa que está querendo me derrubar na assitência aos escritórios... E eu soube que até o filho dele, o Djavan, também se mudou pra Libéria.

- É, parece que a Libéria tá ficando animada, patrão!

- Sem contar um pessoal que fica me atazanando de lá. Tem um povo estranho, um índio apache, um dançarino de

samba...

- E o Linus?

- Não fale esse nome, Jonas! Não fale esse nome!

- Tá, desculpa, chefe!

Ele abaixa a cabeça um pouco.

- Chefe? Por que o senhor não vai pra lá então? O senhor daqui e eles de lá, eles podem ganhar o mercado do senhor praquelas bandas!

- Não dá, Jonas. O governo de lá tem regras diferentes do governo daqui. Mas eu ainda derrubo o governo deles. Presidente Gopp Lefft...

Eu acabo com ele, Jonas! Ele e aquelas leis estúpidas! Acabo com ele ou não me chamo Michel Sófocles! 



**CARLISSON GAUDINO** é Bacharel em Ciência da Computação e pós-graduado em Produção de Software com ênfase em Software Livre. Já manteve projetos como IaraJS, Encyclopédia Omega e Losango. Hoje mantém pequenos projetos em seu blog Cyaneus. Membro da Academia Arapiraquense de Letras e Artes, é autor do Cordel do Software Livre e do Cordel do BrOffice.

Em Salvador - Bahia,  
Dois Grandes Eventos  
29 e 30 de Maio.  
[Entrada Grátis]



**III Encontro Nordestino de Software Livre**  
**IV Festival de Software Livre da Bahia**



# O que aprendemos até agora?

Por Roberto Salomon

Faz 25 anos que o projeto GNU foi lançado.

Faz 18 anos que o kernel Linux foi anunciado na forma daquela mensagem do Linus para a lista do Minix.

Faz 8 anos que a IBM se comprometeu a investir um bilhão de Dólares no desenvolvimento do Linux.

Neste último quarto de século muita coisa aconteceu na área de tecnologia. Muito da evolução se deve a avanços em materiais: máquinas mais potentes e mais baratas que permitiram que hoje possamos

sair de casa e comprar um netbook por menos de R\$ 1.000,00 em suaves prestações mensais em lojas de eletrônicos ditas “populares”.

Por conta deste mesmo avanço, nasceu o mercado de software. Um mercado diferente do tradicional pois é um mercado onde a escassez pode ser controlada. Um mercado do qual dependemos cada vez mais para controlar as complexidades dos demais mercados que criamos. Um mercado que gera altos lucros e movimentos sociais de grande alcance.

Nestes últimos 25 anos passamos por uma experiência social única: aprendemos a usar a tecnologia para nos comunicarmos de forma mais eficiente e, por conta disso, aprendemos a colaborar. Aprendemos que colaborar não implica em interagir com pessoas mas sim com as idéias que elas produzem. No mundo da colaboração que gerou os movimentos do Software Livre e do Código Aberto, não há cor, credo nem idade. Veteranos discutem idéias com novatos (alguns com mais ou menos destreza social) ao

mesmo tempo em que engenheiros experientes vislumbram novos caminhos sugeridos por estudantes no início da graduação.

Aprendemos a pensar de forma diferente, a pensar no problema de forma cada vez mais abstrata e a considerar cada vez mais as idéias e não os atributos físicos dos nossos interlocutores.

Começamos a aprender a aprender.

No entanto, até agora, somos apenas exploradores, caranguejos, colonizando as margens de um novo continente que ainda não temos condições de sequer imaginar as riquezas nele escondidas. Somos uma geração ainda encantada pelas caravelas que nos trouxeram até estas praias. Nossa missão é a de colonizar as margens e preparar a expansão que as próximas gerações farão. Um pouco disso começa a aparecer hoje.

A cada novo passo que damos, a cada nova tecnologia desenvolvida ou melhorada, uma coisa fica cada vez mais clara: os mecanismos de colaboração, criados ou expandidos pelo movimento do Software Livre, se tornam cada vez mais necessários para que possamos continuar nossa caminhada.

O desenvolvimento isolado, aquele realizado para resolver um único problema de uma única pessoa, é cada vez mais

**Aprendemos bastante, mas ainda precisamos aprender a colaborar melhor.**

Roberto Salomon

raro. O que vemos hoje são novas iniciativas que usam de novas formas as tecnologias que desenvolvemos até agora. A evolução é tão rápida que passamos a considerar normal o que até duas gerações atrás seria considerado assombroso. E tudo isso se deve ao poder da colaboração que se consolidou com o movimento do Software Livre.

### Nos ombros de pessoas comuns

Se antes precisavamos de gênios nos ombros de gigantes para podermos evoluir, os mecanismos de colaboração que desenvolvemos permitiram que avancemos colaborando e nos ajudando uns com os ombros dos outros. Algumas empresas já chegaram à conclusão que este é o caminho para o desenvolvimento sustentado de novos produtos e incentivam os seus funcionários a adotar modelos de colaboração como os que vemos nas diversas comunidades de proje-

tos de Software Livre e de Código Aberto.

Até agora, começamos a aprender a colaborar mas apesar dos avanços, pessoas resistem em colaborar com empresas, esquecendo que empresas são apenas grupos de pessoas. Cada vez mais, precisaremos de ferramentas de colaboração que nos permitam ajudarmos uns aos outros. Aprendemos bastante mas ainda precisamos aprender a colaborar melhor.



### Maiores informações:

**Blog do Roberto Salomon:**  
<http://rfsalomon.blogspot.com>



**ROBERTO SALOMON** é arquiteto de software na IBM e voluntário do projeto BrOffice.org.



# Software Livre e o nosso Sistema de Leis

Por Edgard Costa

Em função do meu trabalho e das pesquisas que tenho realizado, tenho percebido que nossos legisladores tem grande preocupação e percepção da importância que o Software Livre tem para um país com as desigualdades sociais, dimensões continentais e econômicas que caracterizam o Brasil. O arcabouço de leis que regulam, regimentam o uso do SL, nas diversas esferas de governo, não são tão tímidas quanto poderíamos pré supor. Porém, o que mais chama atenção é o fato do desconhecimento e da não aplicação destas leis por

parte do executivo em várias esferas de poder e do desconhecimento, por parte do legislador, do que é, efetivamente, Software Livre.

Depois que começamos a estudar o assunto mais a fundo descobrimos que Software não é vendido e sim tem seu código licenciado para utilização em determinadas circunstâncias. O criador do código permite, por licença, paga ou não, o uso do seu código compilado em programa executável – o que de forma leiga e imprecisa denominamos Software. Não vou tecer comentários sobre

os mais variados tipos de licença. Isto ficará para um outro artigo.

No âmbito legal, percebe-se que o legislador, principalmente, está preocupado com custos, e determina o uso do Software Livre porque considera qualquer Software, sob esta bandeira, grátis. Ele não foi instruído sobre o real significado do que é código livre e suas formas de licenciamento. Que a utilização de softwares de código aberto tem condições diferentes e portanto são mais flexíveis, adaptáveis, tendo seu desempenho melhorado

continuamente por sua maior velocidade de atualização, maior correção de erros (bugs) aumentando a versatilidade e confiabilidade deste Software, melhorando, significativamente, o seu desempenho e sua segurança.

Percebemos, também, que o legislador não inclui e nem regulamenta a necessidade de colaboração nos textos regulatórios.

O uso de Software de Código Aberto pressupõe a devolução de algo em troca do uso de alguma ferramenta. Seja em código, seja em correção, em tradução, em publicidade. Algo há que retornar. Não basta

ter a boa vontade de indicar para uso a Plataforma Livre porque é bom para os cofres públicos e para a lei de Responsabilidade Fiscal. Tem que devolver o conhecimento recebido. E isto tem que ser contemplado em lei. Se isto gera custos, então, é melhor repensar a sua utilização.

Os órgãos governamentais e empresas que entendem esta situação são, na sua maioria, dirigidos por pessoas que são membros militantes do mundo Software Livre e contaminados até o mais íntimo da alma com os princípios colaborati-

vos. Os demais não desejam nem ter a noção do que isto representa. A sua necessidade básica foi satisfeita, pronto. Não temos mais nenhuma responsabilidade com isto.

Nas nossas andanças já vimos vários bons exemplos de como o SL tem ajudado os brasileiros. A justiça trabalhista de Minas Gerais e de alguns Estados do Nordeste sabem o quanto foram beneficiados. O desenvolvimento de software

portância da adoção da plataforma SL para o aprimoramento do arcabouço legal existente e do cumprimento da legislação em vigor que obriga estados e municípios a adotarem o SL sobre pena de serem punidos pelos seus Tribunais de Contas. De quantos empregos a plataforma gera, dos impostos e contribuições sociais que são depositados pelos prestadores de serviços. E o mais importante. Que as receitas geradas pelo SL, ficam, efetivamente, no Brasil. Não são remetidos, em certa época do ano para a matriz, que perde parte dos ganhos enviados, pelas filiais, na ciranda financeira de seu país de origem. 

### Seja em código, correção, tradução ou publicidade. Algo há que retornar.

Edgard Costa

para Prototipagem Médica. As crianças beneficiadas por internet e por softwares educacionais confeccionados pela Celepar, do Paraná. Dos paulistas que usam o Acessa São Paulo.

Podemos considerar, para encerrarmos este artigo, que este assunto é, ainda, bastante novo e polêmico. Mas, quero aqui incentivar você leitor, amante do Software Livre, para conversar com seu representante no legislativo local, no legislativo estadual bem como no federal, para expor as diferenças, as necessidades e a im-

### Maiores informações:

**Caso de Sucesso do OpenOffice.org no Tribunal Regional do Trabalho da 4º Região:**  
[www.softwarelivre.org/news/3414](http://www.softwarelivre.org/news/3414)

**Celepar:**

[www.celepar.pr.gov.br](http://www.celepar.pr.gov.br)



**EDGARD COSTA** é membro do Grupo de Usuários BrOffice.org do Estado de S.Paulo, Assurer Cacert – Certificação Digital e autor do Livro BrOffice da Teoria à Prática.

# COMPUTAÇÃO EM NUVEM

Por Cezar Taurion

De aplicativos a sistemas operacionais, tudo vai pra web

Um tema exaustivamente debatido na mídia e em eventos é cloud computing ou computação em nuvem. Ok, mas o que é mesmo computação em nuvem? Ainda existe pouca informação sobre o assunto e mesmo uma consulta à Wikipedia ([http://en.wikipedia.org/wiki/Cloud\\_computing](http://en.wikipedia.org/wiki/Cloud_computing)) nos traz pouca coisa. Bem, podemos dizer que a computação em nuvem é um termo para descrever um ambiente de computação baseado em uma rede massiva de servidores, sejam estes virtuais ou físicos (a nuvem). Na minha opinião a com-

putação em nuvem pode ser visto como o estágio mais evoluído do conceito de virtualização.

Uma arquitetura em nuvem é muito mais que apenas um conjunto (embora massivo) de computadores. Ele deve dispor de uma infra-estrutura de gerenciamento que inclua funções como provisionamento de recursos computacionais, balanceamento dinâmico do workload e monitoração do desempenho.

O conceito de computação em nuvem já é comum em

algumas das empresas mais famosas da Internet como o Google, o Yahoo e a Amazon, que mantém parques computacionais com centenas de milhares de máquinas. Tem um artigo interessante sobre o assunto, que saiu há algum tempo na Wired, chamado The Information Factories, que pode ser acessado em <http://www.wired.com/wired/archive/14.10/cloudware.html>. Para se ter uma idéia do que são estas nuvens, estima-se que as cinco maiores empresas de busca na Internet tenham ao todo um parque computacional de cerca de 2 milhões de servidores!

E que se ganha com esta arquitetura? O primeiro benefício é uma melhor utilização dos recursos computacionais, potencializando os conceitos de consolidação e virtualização. Além disso, reduz sensivelmente o time-to-market para aplicações de e-business e Web 2.0 (onde incluímos aplicações de mundos virtuais e jogos MMPOG), que demandam conceitos do modelo computacional sob demanda (alocar recursos à medida que for necessário, de forma dinâmica).

Tem um case interessante, que pode ser visto no blog de Nicholas Carr, em [http://www.roughtype.com/archives/2007/11/look\\_ma\\_no\\_serv.php](http://www.roughtype.com/archives/2007/11/look_ma_no_serv.php), onde ele relata o surgimento das “serverless Internet company”, empresas



que se lançam na Web sem ter a necessidade de manter seu próprio parque de servidores. No caso ele cita a Mogulus (<http://www.mogulus.com/>), um site para usuários produzirem e exibirem programas de vídeo.

Hoje algumas destas empresas do mundo Internet como o Google e a Amazon já oferecem seu imenso parque computacional para outras empresas.

A IBM, com o Blue Cloud, entra neste segmento com uma oferta que ajuda as empresas a construirem sua própria nuvem. O Blue Cloud é um mix de tecnologias proprietárias com diversos componentes Open Source, como o Xen e o

Hadoop. O Hadoop é um framework open source (sob licenciamento Apache) que permite executar aplicações em ambientes de nuvem. O Hadoop é um sub-projeto do Lucene (<http://lucene.apache.org/hadoop/>) que implementa o modelo de programação MapReduce, desenvolvido pelo Google (<http://labs.google.com/papers/mapreduce.html>).

E já aparecem as primeiras experiências de computação em nuvem baseado em Open Source. Um é o projeto Eucalyptus, desenvolvido pela Universidade da Califórnia, em Santa Barbara, EUA. Este projeto, ainda acadêmico, emula o interface de computação em nuvem da Amazon e pode ser

visto em <http://eucalyptus.cs.ucsb.edu/>. Um outro projeto Open Source para computação em nuvem é o Enomaly ([www.enomaly.com](http://www.enomaly.com)). Este último permite que você construa um ambiente similar ao do Amazon, dentro de seu próprio data center. Os objetivos destes projetos são exploratórios, ou seja, permitir que pesquisadores trabalhem diretamente com tecnologias de computação em nuvem, coisa difícil de fazer quando a maioria das tecnologias são fechadas e proprietárias.

Existe uma clara simbiose entre computação em nuvem e Open Source. Se visualizarmos um massivo data center com milhares de servidores comoditizados e um ambiente virtualizado, parece nítido que o uso de softwares de código aberto permite reduzir mais ainda os custos operacionais e reduzir a dependência dos provedores de nuvem dos fornecedores de tecnologia. Open Source já é usado amplamente por várias das implementações de computação em nuvem, como o Linux (base das nuvens do Google, Salesforce e Amazon), o Xen (usado pela Amazon e na oferta Blue Cloud da IBM), o Hadoop (no Blue Cloud) e mesmo os pioneiros projetos de código aberto para computação em nuvem como o Eucalyptus.

Mas, existe também uma variável importante que aumenta a simbiose entre a computa-

ção em nuvem e o Open Source. A computação em nuvem é basicamente serviço. E serviço é fundamentalmente diferente do modelo tradicional de comercialização de software, que é feita pelo modelo de li-

usuário. Se não houver cumprimento do acordo, o pagamento simplesmente não é efetuado. É diferente da licença de uso do software, uma vez que o usuário paga antecipadamente pelo seu uso e

 Existe uma clara simbiose entre computação em nuvem e Open Source 

Cesar Taurion

cença de uso. Quando um usuário “compra” um software, ele adquire o direito de usá-lo por algum tempo, geralmente pela duração do contrato, embora este direito não lhe ceda a propriedade e patentes do software, que continua nas mãos do fornecedor. Este contrato de licença de uso governa os termos e condições de uso do software pelo usuário e pode ser modificado pelo fornecedor à sua descrição. Na prática, a licença de uso determina basicamente as obrigações do usuário.

Serviços, por outro lado, são governados por contratos ou acordos de níveis de serviço, que definem as obrigações a que o fornecedor acorda, em troca dos pagamentos pelo

mesmo que o software deixe de funcionar a contento, ele não tem direito a nenhum resarcimento. Vemos que a diferença fundamental é que acordos de nível de serviço definem as obrigações do provedor, enquanto licença de software determina as obrigações dos usuários.

Vamos nos colocar na posição de uma empresa, que vai utilizar softwares de outros fornecedores, comercializados pelo modelo de licença e que pretende prestar serviços de nuvem ao mercado. Ela tem que contratar os softwares para repassar seu uso aos seus clientes. Mas, embora estes clientes assinem com ela um acordo de nível de serviço, ela por ter assinado um contrato

de licenciamento com outra empresa, não pode garantir este nível de serviço. O software pode deixar de funcionar ou o contrato pode ser modificado pelo fornecedor para um valor a mais.

Com Open Source, o provedor não está sujeito as estas variáveis, pois pode modificar o software, corrigi-lo ou adaptá-lo às suas necessidades, de acordo com suas próprias demandas. E, mesmo em caso de discordância com os rumos de evolução da comunidade, ele pode, inclusive, criar um fork (árvore de código alternativa) e evoluir o software de acordo com suas necessidades.

Além disso, o provedor pode se concentrar em serviços e conteúdo e não em código em si. Sua receita não vem do software. Um exemplo é o Google que abriu um derivativo do

seu Google File System, que é o Hadoop, para a comunidade de Open Source. Aliás, o Hadoop deverá ser a base tecnológica para o desenvolvimento de novas aplicações Open Source orientadas para operar especificamente em nuvens.

Computação em nuvem e Open Source alteram os princípios básicos dos modelos econômicos da indústria de TI. Há um círculo virtuoso entre eles. A computação em nuvem se beneficia do maior uso de Open Source, porque seu uso elimina dependências críticas que afetariam a entrega de seus serviços. E quanto mais a computação em nuvem utilizar Open Source, mais maduros e evoluídos estes softwares se tornarão, amplificando seu uso. Na verdade, Open Source pode e deve encontrar na computação em nuvem mais um ca-

nal de distribuição.

O resultado final é que a simbiose entre computação em nuvem e Open Source vão pressionar a indústria de TI como um todo, afetando a margem dos modelos de comercialização tradicionais. 

Importante: As opiniões expressas neste artigo, não devem em absoluto serem consideradas como opiniões, visões e idéias de meu empregador, a IBM, nem de seus funcionários. Em nenhum momento, neste artigo, falo em nome da IBM, apenas e exclusivamente, em meu nome.



**CEZAR TAURION** é Gerente de Novas Tecnologias da IBM Brasil. Seu blog está disponível em [www.ibm.com/developerworks/blogs/page/ctaurion](http://www.ibm.com/developerworks/blogs/page/ctaurion)



**Festival Latino-americano de Instalação de Software Livre**

**25 DE ABRIL**  
Em várias cidades da América Latina

**Maiores informações:**  
<http://www.flisol.info>

<http://flisol.espiritolivre.org>

# Alguns aspectos estratégicos da computação em nuvem

Por Orlando Lopes



Kriss Szkurlatowski - www.sxc.hu

A possibilidade da utilização efetiva da Informática como tecnologia passa necessariamente pela compreensão de sua instrumentalização e aplicação na vida cotidiana, mas algumas atividades parecem distantes ou mesmo opostas, como a de uma educação ainda excessivamente identificada com a aprendizagem mediada pela leitura e pela escrita, e de uma cultura que tende a desconfiar da tecnologia, que parece ameaçar a própria condição humana. Até aqui, a ênfase tem sido a das aplicações de origem comercial, mas é óbvio que qual-

quer rede complexa de relacionamento social, como as redes sociais e culturais, precisam considerar suas possibilidades de incorporação dos formatos da computação em nuvem e localizá-la no conjunto dos recursos tecnológicos disponíveis.

Desde os anos 70, quando o conceito de computação em nuvem surgiu, começou-se a definir um espaço virtual 'sem dono e sem fronteiras' onde circulariam software e informações que poderiam ser acessadas por qualquer usuário: o cloud computing, definido como um modelo no qual a

computação (processamento, armazenamento e softwares) não está no computador que estamos usando, mas em algum ponto remoto, acessado via internet. De um lado, essa tendência de arquitetura informacional significa que boa parte das funções de TI não serão mais 'responsabilidade' dos usuários finais, e que as redes de computadores poderão, finalmente, ser simplesmente 'usadas'. De outro, concretiza-se uma das maiores anti-profissões da história da tecnologia: se houve um tempo em que não se pensaria em mais que cinco ou seis computadores para suprir as necessidades do mundo inteiro, hoje torna-se possível imaginar que estamos todos passando a ter supercomputadores em casa, a custos cada vez mais baixos.

É importante lembrar, porém, que os possíveis beneficiários da computação em nuvem não se restringem aos usuários domésticos e às empresas de tecnologia (como o Google, por exemplo) e serviços comerciais (como a Amazon). Alguns dos espaços mais estratégicos para a sua utilização talvez sejam o das diversas figuras jurídicas associadas aos poderes públicos, e aos órgãos e instituições não-governamentais, mas não seria possível falar sobre todos num único artigo. Por hora, apontaremos apenas algumas questões para evidenciar o valor estratégico da computação distribuída em

sistemas educacionais e redes de produção e difusão cultural.

Um dos primeiros pontos a se ressaltar provavelmente será o diminuição de custos com manutenção e melhor aproveitamento dos investimentos em hardware. Num ambiente de escalas tão monumentais quanto o da educação brasileira, por exemplo, qualquer economia na aquisição de equipamentos e serviços pode gerar fundos para melhoria de

sos (como projetos, avaliações de alunos e instituições, etc.) dos sistemas públicos de educação e cultura.

Assim, por exemplo, como os sistemas e redes das empresas precisam suportar a demanda dos períodos de pico em vendas – como o Natal – as unidades de serviços educacionais e culturais precisam atender a picos de processamento (em períodos de matrícula, fechamento de pautas,



Possíveis beneficiários da computação em nuvem não se restringem aos usuários domésticos e às empresas de tecnologia e serviços comerciais.

Orlando Lopes



infra-estrutura e qualificação da mão-de-obra dos professores e técnicos para a utilização dos recursos disponíveis. Gostando-se menos com máquinas e software, por exemplo, fica mais fácil ampliar os investimentos públicos para solucionar certos pontos delicados do desenvolvimento da tecnologia, como a estabilidade do armazenamento e a proteção aos dados restritos (por exemplo os dados administrativos e jurídicos das escolas) e sigilo-

geração de avaliações e estatísticas educacionais, realização de festivais e encontros culturais etc.), demandando uma estrutura que não teria porque ser mantida no decorrer do ano letivo regular. Uma outra grande potencialidade da nuvem é a da atualização das configurações e disponibilização de recursos computacionais sem a interferência na rotina do usuário final, como costumam fazer, por exemplo, sistemas como o Gmail, Yahoo

ou Hotmail.

Especificamente no campo da Educação, há uma série de regulamentações a serem estabelecidas, regulamentações para as quais a consideração sobre as diversas licenças de direitos autorais e difusão desenvolvidas nos últimos anos devem ser tomadas como um necessário ponto de apoio. É preciso estabelecer protocolos e procedimentos para oferecer segurança e confiabilidade aos gestores públicos e aos cidadãos. Talvez não estejamos ainda dimensionando o quanto será importante no futuro (no presente?) o desenvolvimento de soluções em software livre para mediar – ou, mesmo, viabilizar – a integração dos dados e serviços das redes e sistemas educacionais e culturais. Por sua própria natureza e diversidade, a educação e a cultura precisam recorrer a recursos que garantam sua mobilidade, flexibilidade e legitimidade, recursos que não sejam de ninguém em particular, mas que ao mesmo tempo sejam de todos.

Pessoalmente, senti muita, mas muita necessidade mesmo de um sistema em nuvem no início da implementação do Programa Nacional de Cultura Cultura Viva, mantido pelo MinC e em atividade desde 2004. A falta de estratégias para ajudar os participantes a organizarem seus processos de gestão e de um sistema eficiente de relacionamentos quase

puseram a perder um dos mais importantes programas de fomento à cultura das últimas décadas. Chegou-se a criar um sistema baseado em Drupal, mas na prática ele nunca venceu o estágio Beta, e ao que parece foi abandonado. Hoje o MinC está experimentando o Wordpress para desenvolver seus blogs institucionais, e certamente os grandes volumes de dados algum dia estarão (já estão?) circulando em data centers.

O que temos aí, claro, é apenas a ponta do iceberg, sua face mais evidente e nítida. A neutralidade da lógica do software livre torna-o um candidato natural para a concepção, desenvolvimento e implementação de estratégias e soluções que aprimorem e garantam os direitos civis. A natureza coletiva - e portanto necessariamente pública - da produção de software livre praticamente exige que ele se envolva e participe do desenvolvimento de ferramentas educacionais não apenas para ensinar, mas também para processar e distribuir dados e estatísticas que possam ajudar a dinamizar a vida administrativa e pedagógica das escolas públicas e privadas. Ainda que não se possa impedir a participação das empresas transnacionais de tecnologia no processamento (na 'manipulação') de dados estratégicos, os ativistas do software livre parecem assumir cada vez mais a função de construir apa-

ratos jurídicos e gadgets de monitoramento e validação dos dados dos sistemas educacional e cultural (assim como o dos outros sistemas, como o da Saúde, da Habitação...), enquanto cabe aos ativistas sociais e culturais promover a avaliação qualitativa desses dados, confrontando quando necessário com o poder público e com a sociedade.

Aliás, a criação de 'sensores' para acompanhar 'em tempo real' o comportamento da máquina pública seria uma outra contra-profecia genial da informática: ao invés de haver (apenas) um BigBrother acompanhando o que cada cidadão faz em sua vida pública e privada, o ativismo SL pode permitir que a computação em nuvem trabalhe para monitorar os dados e ações daqueles que nos vigiam. Atualmente, se há alguém que precisa de um Big Brother, é exatamente o Estado... 

### Maiores informações:

**MinC - Ministério da Cultura:**

<http://www.minc.gov.br>

**Wordpress:**

<http://www.wordpress.org>



**ORLANDO LOPES**  
Núcleo de Estudos  
em Tecnologias de  
Gestão e  
Subjetividades  
Consultor em  
Responsabilidade  
Cultural e Social



# PAU GARCIA-MILÀ

“Chamar o que nós fazemos de um ‘sistema operacional web’, rompe com a definição tradicional de um sistema operacional.”

João Fernando Costa Júnior entrevistou o criador do eyeOS.

A Revista Espírito Livre nesta edição de estreia conversou com Pau Garcia-Milà, o criador do eyeOS, um dos mais, senão o mais, bem sucedido Sistema Operacional Web.

## Revista Espírito Livre: Quem é Pau Garcia-Milà? Se apresente aos leitores!

Pau Garcia-Milà: Nasci em Barcelona em 1987. Fundei o projeto eyeOS como um hobby aos 17 anos, com Marc Cercós. 3 anos depois nós o convertemos em uma empresa e atualmente estamos trabalhando na Espanha, França, Alemanha e em outros 15 países.

**REL: Em linhas gerais, defina o eyeOS. O que é e como ele funciona? O que é necessário para se usar o eyeOS? O que você tem a dizer sobre cloud computing?**

PGM: eyeOS é um web desktop ou sistema operacional de nuvem. Para usar o eyeOS, tudo que o usuário precisa é um navegador que irá carregar o desktop do eyeOS e que lhe permitirá trabalhar com todas as aplicações instaladas em eyeOS neste servidor. eyeOS é gratuito e de código aberto, assim qualquer um pode ter seu próprio servidor eyeOS. Sobre cloud computing, eu acredito que é o próximo passo na nossa forma de trabalhar com o computador, ce-



eyeOS rodando aplicativo de apresentação

lular e outros dispositivos com acesso a Internet, mas há um problema nesta definição: Existe uma enorme preocupação quanto a privacidade no envio à distância de nossos arquivos privados, fotos, músicas, documentos... E todos usando serviços de terceiros devem se preocupar com isso.

**REL: Existem várias definições e cada um fala uma coisa, mas segundo Pau Garcia-Milà, o que seria um sistema operacional web? Eles trazem benefícios reais para que tipo de usuário?**

PGM: Chamar o que nós fazemos de um sistema operacional web, rompe com a definição tradicional de um sistema operacional. Tentamos focar

nos recursos que podem apresentados aos usuários ao invés de defini-la de uma maneira melhor ou pior. E do meu ponto de vista, um sistema operacional web é útil — diria essencial — para alguém que normalmente utiliza mais de um computador (por exemplo, um laptop e um computador no trabalho e/ou na universidade) e, por exemplo, um telefone móvel. Nesse caso, é possível acessar toda a nossa vida digital em toda parte, com qualquer computador ou telefone móvel.

**REL: Como teve a ideia de criar o eyeOS? No que estava pensando? Qual a ideia por trás do eyeOS?**

PGM: Nós inicialmente cri-

amos o eyeOS pensando em um pequeno pedaço de software que pudesse nos ajudar a manter alguns arquivos em um único local. Tivemos que melhorar e, eventualmente, o publicamos no SourceForge. A partir desse momento (1 de agosto de 2005), o eyeOS começou a crescer como um projeto e como uma alternativa em software livre aos serviços online disponibilizados pelas gigantes companhias de software.

**REL: Ao criar o eyeOS você se inspirou em algum outros sistema operacional? Porque?**

PGM:Tentamos não focar no Windows, Mac ou Linux para criar o design do eyeOS. Marc, que criou o estilo visual projetou algo limpo e fácil de usar. Isso foi provavelmente o que transformou o eyeOS em um sucesso, uma vez que tive-

mos algo incrivelmente bem concebido desde os seus primeiros dias.

**REL: Para que eyeOS funcione, existe a necessidade de um navegador, que por sua vez funciona sob um sistema operacional primário. Seguindo essa linha de pensamento, seria correto chamar uma aplicação que roda sob um browser de sistema operacional? O que pensa a respeito?**

PGM: Acreditamos que o termo sistema operacional deve ser redefinido. Ou talvez, precisamos de encontrar um melhor termo para projetos como o eyeOS. O nosso ponto de vista é que deve haver um, centralizado, sistema operacional e lotes de dispositivos portáteis que rodaria as informações do SO. É muito mais fácil transformar este modelo em uma realidade se todos nós precisamos apenas de

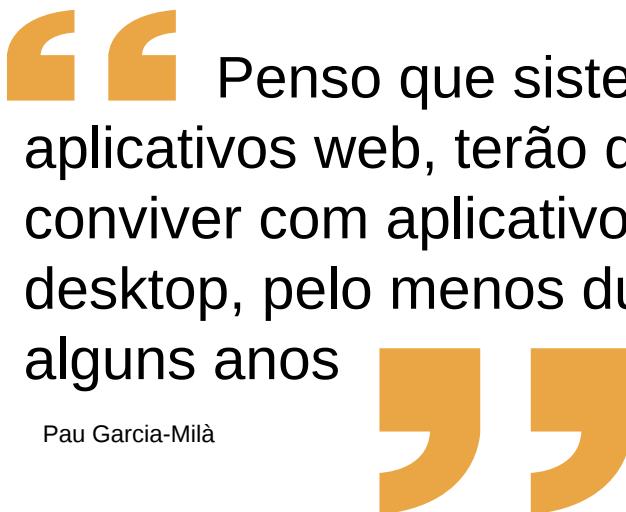
um navegador web e sem a necessidade de instalar qualquer coisa em qualquer dispositivo para trabalhar.

**REL: O que torna o eyeOS diferente dos demais sistemas operacionais web? O que nunca veremos no eyeOS?**

PGM: Quando começamos, não haviam sistemas operacionais web no mercado. Hoje, existem cerca de 55 empresas e projetos dentro deste mercado. Quase todos eles são feitos em flash, e quase todas elas são serviços, e não produtos. Sendo um produto, e sendo software livre, o eyeOS se diferencia entre os demais.

**REL: Você acredita que no futuro teremos apenas sistemas operacionais via web? Como você acha que aconteceria isso? O que esperar dos sistemas operacionais web no futuro? O que podemos esperar do eyeOS em sua versão 2.0 ou 3.0?**

PGM: Penso que sistemas e aplicativos web, terão que conviver com aplicativos desktop, pelo menos durante alguns anos. eyeOS, por exemplo, não serve para rodar programas como o Autocad ou Photoshop. É centrado em trabalho de escritório, trabalho colaborativo com os aplicativos do Office, compartilhamento de arquivos, etc. Se eventual-

Penso que sistemas e aplicativos web, terão que conviver com aplicativos desktop, pelo menos durante alguns anos

Pau Garcia-Milà

mente as ferramentas web e navegadores forem poderosos o suficiente para bater em ferramentas desktop veremos este tipo de aplicações web, e arquitetos e designers usando esses programas através de um navegador web (sem virtualizá-los, o que já pode ser feito).

**REL: Por ter seu código disponível, outros sistemas operacionais web surgem a cada momento, tendo como base o eyeOS. O que você pensa disso? Você acha que eles poderiam contribuir mais ou se integrar ao seu projeto?**

PGM: Existem muitos tipos de forks. Vimos pessoas quebrar a licença e remover o aviso de direitos autorais de eyeOS. Já vimos forks que reproduzem exatamente eyeOS alterando o seu nome, sem contribuir em nada. E finalmente já vimos forks que incidem sobre um mercado específico, que têm contribuído com novas aplicações, widgets e pedaços de

código para o eyeOS.

**REL: Além do eyeOS, você tem outros projetos de código aberto?**

PGM: Atualmente somente o eyeOS.

**REL: Onde os leitores podem saber mais informações sobre o eyeOS?**

PGM: No site oficial do eyeOS, [www.eyeOS.org](http://www.eyeOS.org), tem links para todas as principais comunidades e sites de notícias do eyeOS. O blog oficial ([blog.eyeOS.org](http://blog.eyeOS.org)) tem sempre as mais atualizadas notícias sobre o projeto e sua evolução.

**REL: Qual a mensagem que você gostaria de deixar para quem estiver lendo a revista?**

PGM: Gostaria de convidá-los a descobrir o eyeOS. Se não estão interessados na programação, podem perguntar-se como o eyeOS seria útil

em sua vida cotidiana. Se assim for, gostaria de convidá-los para testá-lo (existe um servidor gratuito em [eyeOS.info](http://eyeOS.info)). Para aqueles que são programadores, gostaria de convidá-los a descobrir o eyeOS toolkit em [dev.eyeOS.org](http://dev.eyeOS.org) e [doc.eyeOS.org](http://doc.eyeOS.org), para que eles possam descobrir o que eyeOS pode fazer por eles.

E se gostarem, mesmo que outros não, gostaria de lhes pedir para deixar as pessoas saberem que existe uma alternativa em software livre para os famosos serviços de editar documentos, compartilhar arquivos e assim por diante. Todos eles estão roubando nossa privacidade, e nós deveríamos estar preocupados com isso. Instalando um servidor eyeOS é bastante fácil (apenas um servidor web Apache +PHP é essencialmente necessário, além de alguns addons se quiserem que o eyeOS para abrir e editar arquivos do Microsoft Office e do Open Office.org). 



<http://www.flisoldf.blog.br>



- Traga seu computador
- Compartilhe e colabore
- Instale Software Livre
- Volte para casa muito mais FELIZ!



# VIRADO PRA LUA

Por Lázaro Reinã

Esse artigo tem como principal objetivo apresentar o uso da linguagem de programação Lua, suas características, especificações, sintaxe e a aplicação desses conceitos, especialmente com relação ao desenvolvimento de jogos, que é um dos grandes trunfos dessa linguagem rápida e eficiente. Sem mais delongas, vamos ao que interessa. Senhoras e senhores passageiros, afivelem os cintos a viagem com destino à Lua começa em... 3... 2... 1...!!!!

## O que é a linguagem Lua?

Eu tenho certeza de que a cada 10 programadores brasileiros 99,9% têm certeza que ‘sabem que não sabem’ o que é a linguagem Lua. O que é uma pena porquê Lua é uma linguagem muito usada lá fora, e sua principal utilização tem sido para o desenvolvimento de games, tais como, World War Craft, Far Cry, Ragnarok, etc. E eu na verdade fico bastante triste em olhar de novo para esses 99,9% programadores, que nem fazem idéia do que seja essa linguagem, e ver que mais uma vez o potencial das terras tupiniquins tem sido explorado em larga escala no exterior e nós aqui não damos a mínima idéia pra isso. O quê? Não entendeu? Vamos lá então, apertem os cintos.

Lua é uma linguagem de programação interpretada, baseada na ‘santa’ linguagem C, bastante leve — quando eu falo leve, é leve mesmo! Com uma sintaxe que na minha opinião é a mais simples de todas. Sintaxe essa que tem algumas características de Pascal, entretanto é inegável o seu parentesco com a mãe de quase todas, linguagem C.

Lua foi projetada para trabalhar em conjunto com outras linguagens, embarcada em um programa principal. Isso acontece através de uma API, destaco a API-C que é a nativa, presente na documentação oficial, e a API-Python, que por

muitos dos que a usam é considerada mais simples que a API-C. Porém, entretanto, toda, é perfeitamente possível criar um script e executá-lo sem que o mesmo esteja ligado a um programa principal, isso é possível graças a um interpretador que está incluso na distribuição de Lua. Mas, quando falamos de desenvolvimento de jogos é essencial que Lua esteja embarcada em um programa hospedeiro, e não contida em um script.

É a única linguagem fora do que nos podemos chamar de ‘eixo’ EUA–Europa–Japão, sendo um projeto 100% brasileiro, que é altamente utilizada dentro do mercado mundial de jogos. Entende minha tristeza? Lua é totalmente brasileira mas, os brasileiros desconhecem sua importância. Os que a conhecem, me desculpem mas, eu tinha que desabafar.

## História

Lua foi gerada no Tecgraf (Puc-Rio) — por Roberto Lerusalmischy, Waldemar Celes e Luiz Henrique de Figueiredo e visava atender a um projeto da Petrobras, no entanto, como demonstrou ser muitíssimo eficiente, leve, etc., ganhou uma abrangência maior, se tornando hoje a linguagem de script mais usada no desenvolvimento de jogos, representando 20% do total de linguagens de script, enquanto que python, a segunda mais utilizada, representa apenas 7%.



(Dados retirados de uma pesquisa realizada pelo site gamedev.net).

A primeira versão lançada em 1993 ainda não era pública, ou seja, apenas a Tecgraf a utilizava. Em 1994 foi lançada a segunda versão e a partir da então Lua foi licenciada por uma licença similar à licença BSD porém, a partir da versão 5.0 passou a ser licenciada pela licença MIT.

## Qual é a sua importância?

Bom... agora que já sabemos o que é a linguagem Lua, surgem algumas dúvidas: Ela realmente é importante? Dá pra levar a vida programando com Lua? Calma! Vamos entender melhor esse assunto.

Quando falamos da importância de uma linguagem de programação, estamos falando da sua repercussão de modo geral, as empresas que a usam, os projetos que usaram essa linguagem, a quantidade de desenvolvedores que a utilizam, correto? Pois bem, veja-

mos se Lua é realmente importante.

Se eu disser que Lua foi para o espaço, literalmente, ela passa ter uma importância maior? Creio que sim! Pois bem, até o ano de 2003 a Nasa usava a linguagem Lua para controlar o compartimento que armazena os gases perigosos dos ônibus espaciais. Bom né?! Vale lembrar que em 2003 Lua completava 10 anos de existência, então desde ‘novinha’ ela já conquistava a atenção de ninguém mais ninguém menos que a NASA!

Agora sem precisar ir tão longe, aqui mesmo em terras brasileiras, lembra dos outros 0,1% que sabem o que é Lua?! Pois bem, o INCOR — Instituto do Coração — radicalizou, usando Lua para monitorar suas UTIs via web. Lua então também ajuda a salvar vidas.

Contra fatos não há argumentos, não é assim que o povo fala? Então, estes fatos falam por si só.

## Mercado de Trabalho

Com todas essas demonstrações de importância fica fácil prever como deve ser concorrido o mercado de Lua, certo? Errado! Como a maioria dos desenvolvedores está mais voltada para as linguagens que estão mais ‘na moda’, nosso mercado em ascensão enfrenta uma escassez de profissionais. Apenas fa-

zendo uma ressalva, a maioria das oportunidades de emprego para desenvolvedores Lua, é para o exterior, vide [www.gamasutra.com](http://www.gamasutra.com). Graças à nossa falta de patriotismo, o Brasil não é o paraíso de Lua, ainda! Podemos perceber que existem algumas empresas que passaram a dar uma importância maior ao potencial de Lua então, as coisas tendem a melhorar por aqui! Quem sabe chegue a ser tão bom quanto é lá fora!

## Por onde dou o ‘start’?

Depois de toda essa parte mais burocrática, vamos então à labuta. Marchemos aos campos de programação!

Para começarmos vejamos o que precisaremos:

— Se você usa Debian, ou algum derivado, basta abrir um terminal e digitar o comando:

```
~$ sudo apt-get install\\
lua50
```

Basta esperar o término da instalação e pronto, é só começar;

— Se você usa outra distro basta acessar o endereço [www.lua.org/ftp](http://www.lua.org/ftp), escolher a versão e baixar o source. Depois é só seguir as instruções de instalação presentes no pacote do source.

— Caso você ainda use software proprietário, todos aqui esperam que não, basta acessar

[www.lua.org/download.html](http://www.lua.org/download.html) ou <http://lua-users.org/wiki/LuaBinaries> e baixar o instalador adequado ao seu SO.

Depois disso...

Vamos agora começar a trabalhar de fato. Depois de instalar o pacote abra o terminal ou prompt, que seja, e digite ‘lua’ para abrir o interpretador.

No terminal façamos assim:

```
~$ lua
Lua 5.0.3 Copyright\
(C) 1994-2006 Tecgraf,\ 
PUC-Rio
> teste = "funcionou!"
> print (teste)
funcionou!
```

Se você fizer aí e acontecer exatamente assim, está tudo ok!

Nesse exemplo podemos ver o interpretador Lua em ação, perceba que nós não escrevemos em um arquivo e sim digitamos os comandos diretamente no shell.

Veremos mais alguns exemplos daqui a pouco.

## Pondo a mão na massa — Início

A partir de agora começaremos a tratar da linguagem aplicada, por assim dizer. Porém visando atender a todos os leitores, não importando o nível de conhecimento que a pessoa tenha, esse início pode ser muito introdutório, porém

lembre-se de que seguindo a filosofia do Software Livre não podemos fazer acepção de pessoas. Portanto seja paciente que a gente chega lá!

Começaremos pelo exemplo clássico de toda linguagem , Hello World!:

Abra o terminal e digite o seguinte:

```
~$ vim primeiro_ex.lua
```

Com esse comando abriremos no editor de textos VIM, que é uma versão melhorada do famoso VI, um aquivo onde editaremos nosso código.

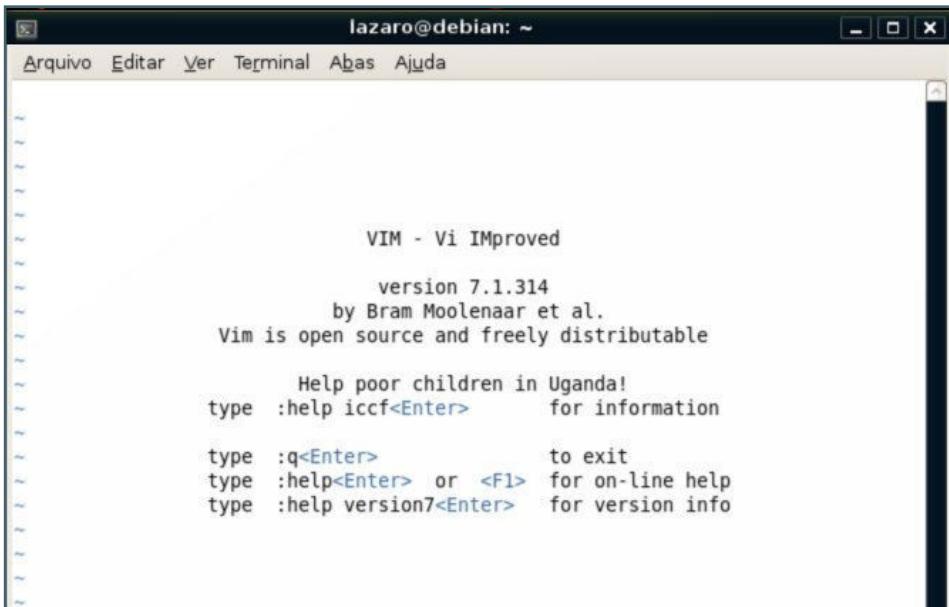
Essa tela é o VIM em execução [Figura 1]. Para ativar o modo de edição digite i, e então escreva o seguinte:

```
print "Hello World!";
```

Cetifique-se de finalizar a linha com ;. Agora precione a tecla Esc e digite :wq, e vejamos o resultado. Para tal devemos digitar o comando lua seguido do nome do arquivo que criamos, cuja extensão obrigatoriamente deve ser '.lua'.

```
~$ lua primeiro_ex.lua  
Hello World!
```

Eis então, nosso 'Hello World!' rodando. A partir de agora iremos nos aprofundar no aspectos mais técnicos da linguagem, tais como, valores e tipos, sintaxe, API, etc. Até chegarmos no que eu acredito



▲ Figura 1 - Editor de texto VIM

ser a parte mais interessante, como se dá a aplicação desses conceitos no desenvolvimento de jogos.

Porém esses assuntos ficarão para a próxima edição! Não percam, até porquê será a partir dela que a coisa começa a esquentar.

Até logo!

**Site Lua-Users:**  
<http://www.lua-users.org>

**Site Gamasutra:**  
<http://www.gamasutra.com>

**LUA conquista Microsoft:**  
<http://tinyurl.com/cklm4q>

**JB Online: Nasa, Intel e Incor empregam Lua:**  
<http://tinyurl.com/c5mfc7>

## Maiores informações:

**Site Oficial:**  
<http://www.lua.org>

**FTP Oficial:**  
<http://www.lua.org/ftp>

**Download da Linguagem Lua:**  
<http://www.lua.org/download.html>  
<http://lua-users.org/wiki/LuaBinaries>



**LÁZARO REINÃ** é usuário Linux, estudante C/C++, Lua, CSS, PHP. Integrante do EESL, ministra palestras e mini-cursos em diversos eventos de Software Livre.

# DESENVOLVIMENTO DE SOFTWARE LIVRE

Por David de Almeida Ferreira

Dicas  
importantes  
para que o  
futuro  
desenvolvedor  
de softwares  
possa iniciar  
uma promissora  
carreira

É muito comum as pessoas ingressarem no universo do Software Livre com anseios de desenvolver ou customizar softwares, porém a realidade que percebemos é outra, da grande maioria das pessoas que entra neste universo uma boa parte não possui experiência em desenvolvimento ou quando possui acaba não sendo suficiente devido à complexidade dos softwares, visto que o grau de clareza, entendimento e experiência dos desenvolvedores varia imensamente de um projeto para outro, o que pode dificultar que candidatos no-

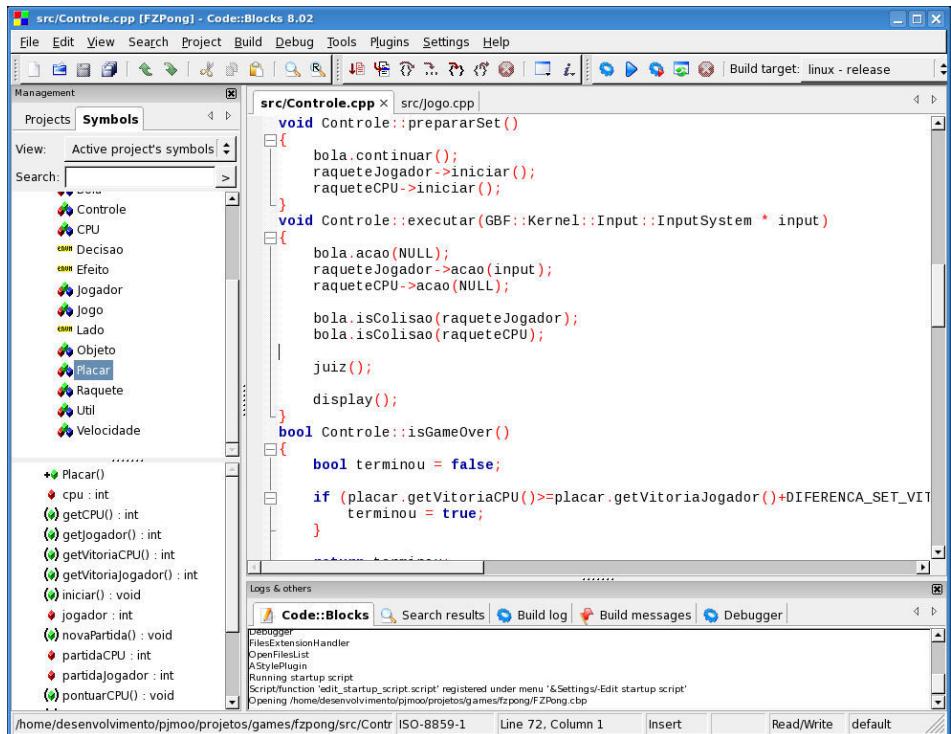
vatos a desenvolvedores possam de fato customizar os tão desejados softwares, abrindo então a oportunidade para que novos softwares semelhantes sejam desenvolvidos, variando entre melhores ou piores decisões arquiteturais, técnicas de programação, funcionalidades e tecnologias.

Além deste fato, outro que devemos ter em mente é que desenvolver software não é uma tarefa tão simples, ela é complexa, porém nada de outro mundo, um bom desenvolvedor de software deve possuir uma boa capacidade

de raciocínio lógico aliado ao conhecimento de uma ferramenta de desenvolvimento e somando a uma necessidade que deseja satisfazer, já que softwares são criados para atender uma necessidade.

O primeiro passo do futuro desenvolvedor de software é iniciar-se em lógica de programação, a qual pode ser vista na grande maioria dos cursos superiores de computação (nas suas mais variadas versões: processamento de dados, informática, análise de sistemas e etc...) ou em cursos técnicos oferecidos por diversas empresas, mas tome alguns cuidados, procure cursos onde os professores estão preocupados em desenvolver seu raciocínio lógico, geralmente isso é feito com pseudo linguagens de programação como o ‘portugol’, pois permite que o aprendiz de desenvolvedor aprenda a pensar sem se atrelar a uma linguagem de programação, permitindo assim que ele domine os aspectos básicos utilizados nas linguagens de programação, afinal um bom software é feito graças as habilidades de um bom desenvolvedor e não de uma ferramenta.

Como segundo passo, o desenvolvedor deve adquirir experiência em uma linguagem de desenvolvimento, tradicionalmente isso se adquire por meio de estudos (livros, artigos, tutoriais, cursos) e no desenvolvimento de softwares (exercícios, prática, testes). Es-



▲ Codeblocks

ta é uma tarefa evolutiva, o desenvolvedor precisa treinar bastante para ampliar seu conhecimento na linguagem para poder desenvolver softwares melhores e mais complexos.

O terceiro passo, poderíamos dizer que seria o qual o desenvolvedor, além de saber lógica de programação, ter conhecimento suficientes de uma linguagem, precisa conhecer novas técnicas e ferramentas, como por exemplo:

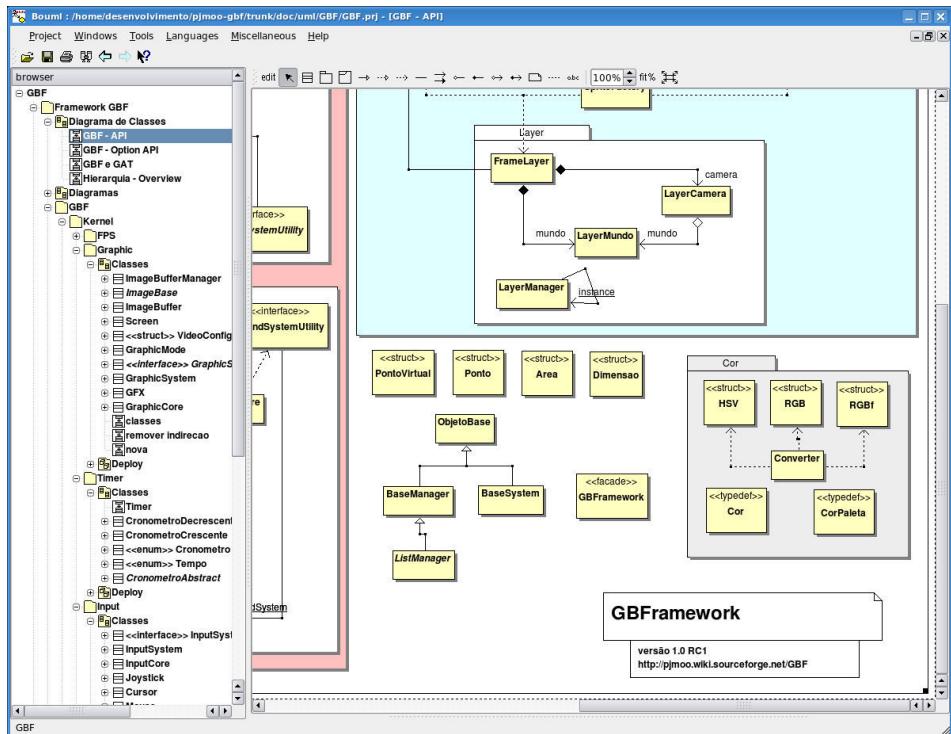
- Application Programming Interface (API) ou Interface de Programação de Aplicativos, podemos entender como uma extensão da linguagem de programação fornecendo novas funcionalidade (capacidades) para a criação de softwares mais elaborados e complexos. As APIs são res-

ponsáveis por permitir a integração de seu software a sistemas operacionais, banco de dados, multimídia e até mesmo integração com outros softwares. Por meio das APIs podemos, por exemplo, desenvolver softwares, como jogos, os quais possuem uma boa complexidade e requerem acesso a diversos recursos (teclado, mouse, joystick, vídeo, internet, som).

- Integrated Development Environment (IDE) ou Ambiente Integrado de Desenvolvimento, é um software que reúne em si diversos outros softwares com a intenção de auxiliar na tarefa de desenvolvimento de softwares. Podemos imaginar uma IDE como, por exemplo, um grande editor, possuindo acesso fácil a edi-

ção de Código Fonte, Documentação da Linguagem e API, Compilador, Depurador, Navegador de Código e em alguns casos editores de formulários(interfaces com o usuário). Geralmente as IDE atendem apenas uma determinada linguagem, porém uma mesma linguagem pode possuir inúmeras IDEs. Onde assim temos a opção de escolher a IDE de acordo com nosso sistema operacional, ou de acordo com as facilidade(recursos) ou a que melhor se adeque a nossa necessidade.

- Programação Estruturada, é um paradigma clássico da programação, nele a estrutura do software é focada na divisão de dados e funções / procedimentos (function e procedure), onde assim pensamos de forma separada o que queremos armazenar (na memória) e nas funções para manipular esses dados. Porém, não existe muito controle sobre quem pode manipular os dados ou suas rotinas de processamento, criando assim um modelo de programação onde



▲ BoUML

o desenvolvedor deve ter cuidado redobrado ao acessar os dados, principalmente quando qualquer parte do software pode manipular erroneamente os dados e prejudicar assim os resultados. Apesar desse cuidado redobrado, a programação estruturada está presente na grande maioria das linguagens e ainda é muito utilizada no

desenvolvimento de diversos softwares ou por questões tradicionais (experiência, maturidade, gosto) ou por limitações da tecnologia utilizada.

- Programação Orientada a Objetos, é o paradigma da programação que surgiu para aperfeiçoar a programação estruturada, onde agora o seu foco não é na divisão de dados e processamento e sim, na idéia de unidades lógicas, assim temos dados e funções altamente agregadas simplificando assim diversas questões, pois agora ao criarmos essas unidades lógicas (chamadas de objetos) pensamos em que dados cada objeto precisa e de que forma ele irá manipulá-los, assim também como quem pode e como vai manipular esses objetos. É uma gigantesca mu-

**“ Desenvolver é a arte de transformar rabiscos em idéias e soluções para serem executadas no computador. ”**

David A. Ferreira

dança da forma de se pensar em desenvolver software e nem todas as linguagens de programação dão subsídios para este paradigma.

• Unified Modeling Language (UML), é uma linguagem de modelagem (entenda-se como uma espécie de desenho), que auxilia o desenvolvedor a visualizar como os objetos estão relacionados e como eles se comportam. Esta modelagem é definida por diversos diagramas os quais fornecem informações detalhadas sobre diversos aspectos da construção do software, o que simplifica muito a comunicação entre os desenvolvedores, pois permite mostrar como o software foi montado, quais são suas partes (chamaremos de classes), o que cada uma fornece (chamaremos de métodos), quais partes podem ser substituídas,

removidas ou customizadas.

Como passo complementar indicaremos algumas ferramentas livres que devem fazer parte do dia a dia do desenvolvedor de software:

#### • Linguagem ou Plataforma

GNU Compiler Collection:

<http://gcc.gnu.org>

Minimalist GNU for Windows:

<http://www.mingw.org>

#### • IDE

CodeBlocks: <http://www.code-blocks.org>

Lazarus: <http://www.lazarus.freepascal.org>

Eclipse: <http://www.eclipse.org>

#### • API

SDL: <http://www.libsdl.org>

OpenGL: <http://www.opengl.org>

GTK: <http://www.gtk.org>

Qt: <http://www.qtsoftware.com>

#### • Modelagem UML

BoUML: <http://bouml.free.fr>

Com essas dicas o futuro desenvolvedor de software pode começar sua preparação para esta jornada quase espiritual, que é a arte de transformar rabiscos em idéias e soluções para serem executadas no computador, tendo em vista a criação de ferramentas para simplificar a sua vida, ou a dos outros, ou simplesmente pelo fator diversão e entretenimento. Fique claro que não será uma tarefa fácil, porém seus estudos aliados a sua determinação provarão para você sua verdadeira capacidade a qual está muito além do que você jamais imaginou, esperamos ver muitos e muitos software livres (utilitários, aplicativos, sistemas e jogos) produzidos muito em breve.

Nas próximas edições falaremos sobre Desenvolvimento de Jogos com Software Livre.



▲ FZPong



**DAVID ALMEIDA FERREIRA**  
<http://www.davidferreira.com.br>  
Especialista em Engenharia de Software com ênfase em Padrões de Software(UECE), É Arquiteto de Software do Banco do Nordeste do Brasil (BNB). É integrante do Projeto de Software Livre Ceará(PSL-CE) e do Grupo de Desenvolvedores de Jogos do Ceará(GDJCE).

# VOCÊ SABE MESMO O QUE É UM PADRÃO ABERTO?

Por Jomar Silva

**Não leve gato por lebre, saiba como identificar um padrão aberto**

Como todo bom nerd, me divirto de verdade vendo o mercado de TI adotar (e às vezes até criar) determinadas ‘buzzwords’ e, mesmo sem compreender seu real significado, acabam dando um jeito de produzir diversas pérolas de marketing com a nova palavrinha da moda.

Foi assim no início do ano 2000, quando o termo ‘IP’ (protocolo da Internet) chegou ao mercado de telecomunicações. Me lembro de ter participado de um evento internacional naquele ano e da noite para o dia tudo era IP (e

cá entre nós, como eu vinha da indústria de software e redes IP, quando eu queria encerrar uma discussão qualquer era só usar a frase “Isso se resolve via IP” e pronto, meus interlocutores faziam ‘cara de conteúdo’ e estava encerrada a discussão... o IP era mágico).

Em nossos dias, temos que conviver com três buzzwords, muitas vezes extremamente mal utilizadas: Interoperabilidade, ‘Open’ (vale também sua versão em português ‘Aberto’, apesar de que a versão em inglês sempre tem mais apelo de marketing) e a

que quero falar hoje, Padrão Aberto.

Percebi que estava tudo quase perdido em relação aos padrões abertos quando em uma das minhas viagens no ano passado, li numa revista de bordo da empresa aérea um anúncio de uma pequena empresa de software, dizendo que produzia sistemas de gestão (ERP) em padrões abertos. Fiquei curioso com o anúncio e anotei o site da empresa.

Quando cheguei em terra firme, visitei o site da empresa e até hoje eu não consigo entender com clareza o motivo de utilizarem ‘Padrões Abertos’ no seu anúncio, pois a empresa desenvolve um sistema proprietário, baseado em softwares proprietários e de aberto não tem absolutamente nada. Foi para mim a constatação de que o termo se tornou de vez uma buzzword.

Apenas por curiosidade, fui dar uma olhada nos parceiros de tecnologia da empresa e adivinha? Um dos parceiros é exatamente a empresa que mais falou bobagem sobre padrões abertos e interoperabilidade nos últimos dois anos (acho que vocês já estão bem grandinhos para saber de quem estou falando). Sendo assim, a pequena empresa de software estava apenas vendendo o peixe que comprou, e que muitas vezes é vendido: Especificação disponível na Internet é um padrão aberto (note que usei propositadamente a pala-



Artwork do Document Freedom Day. Leia mais a respeito nesta edição.

vra disponível e não acessível... na revenda da Audi aqui perto de casa existe um A4 zerinho disponível, mas não consigo acessá-lo... entendeu?).

Desde que vi este anúncio, aumentei ainda mais meu empenho em escrever sobre padrões abertos e em falar sempre no tema antes de iniciar qualquer discussão sobre o ODF (OpenDocument Format), padrão aberto que ajudo a desenvolver no comitê do OASIS (OASIS ODF TC).

Existem diversas definições sobre Padrão Aberto e, a falta de existir uma que seja considerada como ‘a definitiva’ acaba abrindo brecha para que o termo seja muito distorcido.

A versão que é mais citada e comumente utilizada por todo mundo (e a que eu considero como a mais completa), é a definição dada pela União Europeia em seu documento de definição de arquitetura de Interoperabilidade (European Interoperability Framework for pan-European eGovernment Services), similar ao nosso e-Ping (que inclusive cita a definição europeia).

Ela define que um padrão aberto é aquele que cumpre com quatro requisitos:

- O padrão foi adotado e será mantido por uma organização sem fins lucrativos, e seu desenvolvimento contínuo irá ocorrer com base em um processo de tomada de decisão

aberto e acessível a todas as partes interessadas (através de consenso ou decisão por maioria, etc.)

Destaco a importância da entidade que desenvolve o padrão ser uma entidade sem fins lucrativos, pois este tipo de entidade é sempre mantida por diversas empresas, e isso acaba sendo fundamental para que a agenda de trabalho dessa entidade não seja direcionada por uma única empresa. Além disso, a estrutura e os processos jurídicos e administrativos deste tipo de entidade normalmente são muito transparentes.

O processo de decisão aberto e acessível a todas as partes interessadas é fundamental para que o trabalho realizado dentro dos comitês de desenvolvimento dos padrões possa ser acompanhado de perto por qualquer interessado, permitindo ainda a participação de qualquer um.

Obviamente devem existir regras para que tudo isso seja feito, e cada vez mais eu acredito que o acompanhamento popular é uma excelente vacina contra qualquer falta de bom senso que possa existir em algum membro do comitê (e digo isso com conhecimento de causa, pois tenho observado este tipo de coisa dentro do comitê do ODF no OASIS — [http://www.oasis-open.org/committees/tc\\_home.php?wg\\_abrev=office](http://www.oasis-open.org/committees/tc_home.php?wg_abrev=office)). Todos os e-mails trocados pelo comitê são públi-

cos e a blogosfera tem se encarregado com efetividade de questionar cada ‘escorregada’ que alguém dá dentro do comitê. As pessoas podem ainda contribuir com nosso trabalho por lá através da lista pública de comentários).

- A norma foi publicada e seu documento de especificação está disponível gratuitamente, ou através de uma taxa simbólica. Deve ser permitido a todos copiar, distribuir e utili-

der o ‘mal exemplo’, a especificação do ODF na versão ISO (ISO/IEC 26.300, que é o mesmo documento do OASIS com uma folha de rosto da ISO), só está disponível para venda no site da ISO. Antes que alguém me chame de desinformado, a versão que pode ser encontrada gratuitamente no site da ISO deve somente ser utilizada pelos membros dos comitês da ISO. Quem não for membro de comitês da

**“ Qual é a utilidade de uma especificação aberta se as pessoas não tiverem acesso a sua documentação ? ”**

Jomar Silva

zá-la gratuitamente ou a uma taxa simbólica.

Qual é a utilidade de uma especificação aberta se as pessoas não tiverem acesso a sua documentação ?

Usando novamente o ODF como exemplo, seu documento de especificação pode ser encontrado gratuitamente no site do comitê do OASIS.

Citando o mesmo documento, agora como o que consi-

ISO e usar o documento está violando o copyright da ISO... bonito, né?

Mais um ‘mal exemplo’ está na nossa querida ABNT, que também só disponibiliza a versão em português do ODF para comercialização (simbólicos R\$ 354,10, e o mais irônico é que a ABNT não participou do trabalho de tradução do documento, realizado por algumas empresas, coordenado e revisado por mim).

• A propriedade intelectual - ou seja, patentes eventualmente presentes - do padrão (ou de partes dele) está disponível irrevogavelmente através de uma base livre de royalties.

Este é para mim um ponto fundamental e que explica o motivo de tantas inovações surgirem com base em padrões abertos. Royalties podem ser um bom negócio para quem desenvolve uma tecnologia ou padrão, mas são uma barreira, muitas vezes intransponível, para quem deseja implementar aquele padrão.

Imaginem se o ODF fosse licenciado através da cobrança de royalties. Quem pagaria para utilizá-lo dentro do OpenOffice (ou KOffice, Abiword e tantos outros softwares livres que suportam o padrão) ? E o protocolo TCP/IP ? Será que teríamos evoluído tão rápido com a Internet se cada empresa que pretendesse utilizar o protocolo tivesse que pagar alguns milhões de dólares de licenciamento ?

Gosto sempre de destacar que ser livre de royalties não é garantia de que o padrão tem licenciamento livre, mas este pode ser um assunto para uma outra coluna, caso vocês se interessem por ele.

• Não há restrições sobre a reutilização do padrão.

Este é nosso velho conhecido, que também responde pelo nome 'Não reinventarás a roda'.

Quem conhece um pouco mais sobre tecnologia de redes, sabe que toda a Internet funciona com base em uma pilha de protocolos e padrões abertos (TCP/IP e recomendações W3C). A maioria destes padrões não foi desenvolvida do zero, e toma como base grande parte do que já foi especificado em um outro padrão já existente.

malmente os comitês internacionais de desenvolvimento de padrões abertos têm como membros as maiores autoridades no assunto tratado.

De algum tempo para cá, existe uma forte tendência de que esta definição de padrão aberto da União Europeia ganhe um quinto ponto: A existência de ao menos uma implementação de referência

 Quem conhece um pouco mais sobre tecnologia de redes, sabe que toda a Internet funciona com base em uma pilha de protocolos e padrões abertos 

Jomar Silva

O ODF também é assim. Utilizamos XML, Xforms, Xlinks e diversos outros padrões abertos já existentes. Utilizamos ainda conceitos já existentes, como a estrutura utilizada pelas tabelas e planilhas, muito parecidas com as tabelas em HTML.

Um outro ponto importante desta reutilização, é que através dela podemos utilizar sempre o que existe de mais bem estruturado em cada setor de padronização, pois nor-

em código aberto (e eu sou um dos que defende esta tese).

Para que possa ser implementado em código aberto, um padrão deverá estar bem documentado e construído, ter seu documento de especificação disponível, ter suas questões de licenciamento de acordo com a definição apresentada e o mais importante de tudo, ser viável tecnicamente.

Vocês podem me chamar de maluco por colocar a viabilidade técnica em questão, mas



Poster produzido pela equipe de designers da Red Hat

quem acompanhou a trajetória do OpenXML na ISO até agora, sabe que ele é um padrão que ainda não pode ser (e não foi) implementado, dado o número elevado de inconsistências técnicas que ainda possui (e mesmo assim, foi aprovado como norma ISO... legal essa ISO, não é mesmo?).

A existência de uma implementação em Open Source seria para mim como um anabolizante (ou um catalisador) para o desenvolvimento de mais aplicações com suporte ao padrão. Esta aliás é uma discussão muito interessante, pois muita gente gosta de criticar o ODF dizendo que sua interoperabilidade hoje é baseada no fato de que muitas das implementações existentes são baseadas em um mesmo código fonte, do OpenOffice.org.

Esta crítica não é tão bem fundamentada assim, pois existe de fato hoje interoperabilidade entre implementações sem relação alguma com o OpenOffice.org, mas focando na crítica feita, será que ela faz mesmo algum sentido?

Em tempos de crise econômica como vivemos hoje, onde os investimentos em desenvolvimento são cada vez mais escassos, não seria muito mais sensato e inteligente o desenvolvimento de uma base comum de processamento de documentos ODF, que pudesse ser executada em diversas plataformas e utilizado por todo e qualquer projeto de software? Será que não é esse o rumo que as coisas deveriam tomar?

Acho que agora todos vocês já sabem distinguir gato de lebre. Toda vez que alguém lhe disser que um padrão é aberto, faça a ele quatro perguntas básicas:

1 — Quem desenvolveu, e como posso acompanhar e participar do desenvolvimento?

2 — Onde acesso sua documentação completa?

3 — É necessário algum licenciamento ou pagamento para sua implementação ou utilização?

4 — Posso criar o meu pa-

drão usando-o como base?

E se quiser apimentar ainda mais:

5 — Posso implementar este padrão no meu projeto de código aberto?

Se gaguejarem na resposta, ele só é Padrão Aberto no nome: buzzword.

## Maiores informações:

### ODF Alliance:

<http://www.odfalliance.org>

### Site da W3C:

<http://www.w3.org>

### Site da OASIS:

<http://www.oasis-open.org>

### ISO:

<http://www.iso.org>

### ABNT:

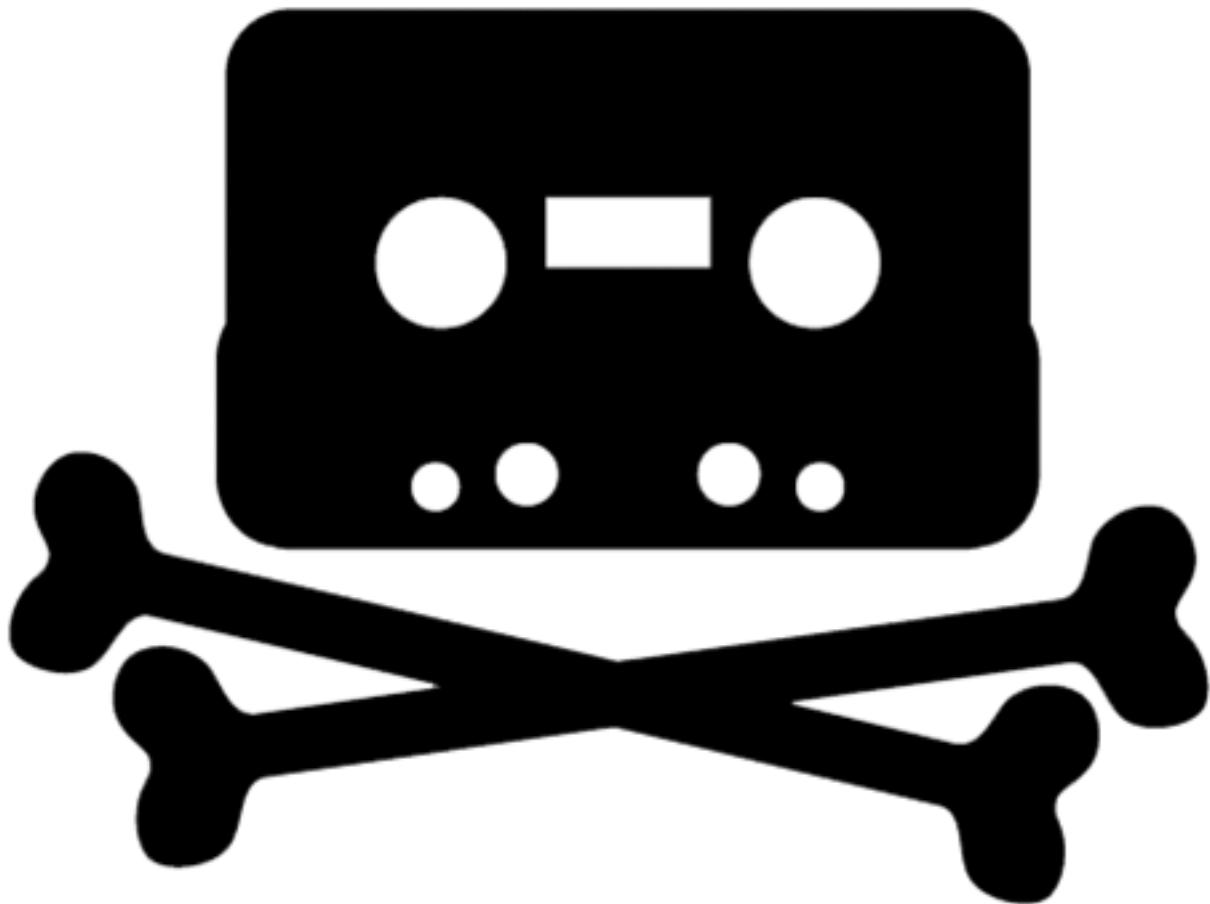
<http://www.abnt.org.br>

### Blog do Jomar:

<http://homembit.com>



**JOMAR SILVA** é engenheiro eletrônico e Diretor Geral da ODF Alliance Latin America. É também coordenador do grupo de trabalho na ABNT responsável pela adoção do ODF como norma brasileira e membro do OASIS ODF TC, o comitê internacional que desenvolve o padrão ODF (Open Document Format).



[www.ben-costello.com](http://www.ben-costello.com)

# Julgamento do The Pirate Bay

Reflexo de um acirramento de opiniões sobre o  
compartilhamento digital

Começou em 16 de fevereiro de 2009 o ‘julgamento espetáculo’ — *spectrial* — do The Pirate Bay. Para quem não sabe, a “baía dos piratas” é um dos maiores trackers Bit-Torrent do mundo, com uma grande e participativa comunidade de usuários, onde se trocam arquivos torrent que apontam para áudios, filmes, documentos, jogos e mais arquivos digitais existentes nos discos rígidos desses membros.

A acusação, feita pela IFPI (Federação Internacional da Indústria Fonográfica), representante das grandes empresas discográficas do mundo, foi de que os responsáveis pelo Pirate Bay promovem... ‘pirataria’. Segundo os acusadores, ‘pirataria’ é o ato de distribuir conteúdo protegido sob copyright sem autorização, causando danos aos artistas e, principalmente, a suas empregadoras, as gravadoras.

Para além do uso do termo ‘pirataria’ sob um novo significado — o que constitui uma típica arma retórica, pois acaba por relacionar uma situação nova, o compartilhamento digital, com uma prática criminosa tipicamente marítima — o julgamento em questão representa um confronto entre uma tecnologia revolucionária do ponto de vista da distribuição cultural e uma indústria anacrônica que tenta de todas as formas manter o seu antigo modelo de negócios. E esse embate pode refletir algo novo

na comunidade que discute Cultura Livre.

Processar empresas e grupos que fazem compartilhamento de materiais desse tipo não é novidade: o Napster encontrou o fim de sua pioneira iniciativa p2p em um processo movido pela RIAA (Associação da Indústria Discográfica da América). eDonkey, Grokster e outras, também. A linha de batalhas judiciais chegou aos próprios usuários desses serviços. Milhões foram levados aos tribunais, entre eles, uma senhora de 71 anos e uma criança de 12 anos de idade.

Mas existe uma diferença fundamental entre o julgamento do The Pirate Bay e os ocorridos anteriormente. Dos primeiros processos até esse, comunidade e a própria sociedade sofreram uma mudança de percepção quanto a questão do download, distribuição e compartilhamento, seja a favor de uma posição de liberação, seja por uma posição de criminalização.

O lobby de grupos como o IFPI, RIAA, MPAA e outros em governos ao redor do mundo em adição a uma interpretação mais conservadora das legislações, objetiva proteger o modelo de negócios dos interesses que representam. Assim, aca-

bam por criminalizar o compartilhamento e, na prática, fomentam a criação de leis de controle sobre a rede. Na França, por exemplo, está em tramitação uma lei que restringirá o acesso do internauta a rede caso ele seja identificado realizando download de material sobre copyright.

Agora, segue a pergunta: como saber se aquele pacote que trafega na rede é material protegido por copyright, um material de distribuição liberada ou um e-mail pessoal? Se você pensou que os pacotes serão violados e espionados, acertou. E mais, como proibir o acesso de um internauta a rede? Será através do IP? Se for, sabemos que não dará certo. Será que todo internauta terá um ‘identificador’ pessoal ao acessar a rede? É a Sociedade do Controle chegando ao ciberespaço.



The Pirate Bay

wikimedia.org

Pela liberação temos a conscientização crescente da sociedade que a distribuição a baixo custo desse material é uma forma de universalizar o acesso à cultura, principalmente das classes mais baixas que não tem acesso ao cinema ou casas de espetáculo, mas que conseguem baixar o material

disputa entre indústria discográfica e emissoras de rádios, ou a mais recente indústria cinematográfica e emissoras de televisão contra a Sony e seu video-cassete.

A saída seria encontrar uma forma de permitir a partilha desse material, procurando assegurar de alguma forma que os artistas recebam pelo seu trabalho. Uma questão difícil de ser resolvida, e que demandará uma mudança de paradigma na maneira como a indústria cultural sobrevive.

Tentativas de construir uma alternativa nestes moldes estão sendo discutidas de forma verdadeira e não apenas no discurso. Bård Vegar Solhjell, Ministro da Educação e Pesquisa e membro do Partido da Esquerda Socialista da Noruega, cogita a possibilidade de levar para a rede um modelo de negócios semelhante aos das rádios, onde os usuários ouvem música gratuitamente e as rádios pagam os direitos dos autores, através de patrocínio. Para o ministro, a indústria deveria compreender a Internet, ao invés de tentar combatê-la, e a legalização dos downloads de material em copyright deve ser defendida pois é uma forma inovadora de distribuição de cultura.

Enquanto uma solução co-

mo esta não chega, seguirão os processos. O veredito sobre o julgamento do Pirate Bay deve sair no dia 17 de abril, e será possível de recurso por ambos os lados. Mas, para além de sair culpado ou inocente, o grupo por trás do site está cumprindo seus objetivos, expressos nas palavras de Anakata, que podem ser vistas no documentário Good Copy, Bad Copy, produzido anos atrás: "vejo o The Pirate Bay como uma organização de desobediência civil, que visa forçar as mudanças na lei de copyright que temos hoje". 

## Proibir o desenvolvimento tecnológico nunca se mostrou eficaz

Filipe Oliveira Saraiva

em uma LAN house ou adquirir um produto similar a um preço menor no comércio não-formal. Temos até a fundação de um Partido Pirata na Europa, representado no Brasil por um movimento com o mesmo alinhamento ideológico de descriminalização do compartilhamento e neutralidade na Internet.

Como nas outras vezes em que grupos de partilha foram processados, este caso do Pirate Bay, se julgado culpado, não acabará com o compartilhamento de material protegido por copyright na rede. Proibir o desenvolvimento tecnológico nunca se mostrou eficaz — guardadas as devidas proporções, vemos apenas a repetição da mesma

### Maiores informações:

#### The Pirate Bay:

<http://www.thepiratebay.org>

#### Documentário Good Copy, Bad Copy

<http://www.goodcopybadcopy.net>

#### Blog Liberdade na Fronteira

<http://www.liberdadenafronteira.blogspot.com>



**FILIPE DE OLIVEIRA SARAIVA** estuda Ciências da Computação na Universidade Federal do Piauí, entusiasta do Linux, da Cultura Livre e das possibilidades de criação coletiva oferecidas pelo mundo conectado. É pesquisador da área de Cibercultura, Pesquisa Operacional e Inteligência Artificial.

# Quando o pinguim vai à escola: o software livre como recurso no ensino e na aprendizagem

Por Sinara Duarte



Piotr Lewandowski- www.sxc.hu

A partir da década de 80, com o advento e democratização das TICs – Tecnologias de Informação e Comunicação na educação começaram a surgir pesquisas acerca dos benefícios da utilização do computador na aprendizagem, destacando-se os estudos de Moram (2008), Valente (1999), dentre outros. Assim, diversos estudos surgem a cada dia sobre as possibilidades educativas do computador na sala de aula.

Todavia, na verdade, o uso da tecnologia digital para ensinar não é uma idéia nova. Muito embora os primeiros computadores somente tenham surgido na década de 40, com fins bélicos, pode-se afirmar

que os fundamentos em que se baseiam a sua estrutura remontam a centenas ou até mesmo milhares de anos atrás com os ábacos, conhecidas como as primeiras máquinas “pensantes”.

Em termos educacionais, Parpet nos anos 60, já vislumbrava os computadores como instrumentos para expandir o aprendizado dos estudantes para além das limitações físicas da sala de aula. Na época, suas teorias pareciam ficção científica. Entre 1967 e 1968, desenvolveu uma linguagem de programação totalmente voltada para a educação, o Logo, no qual a criança poderia desenvolver suas habilidades cognitivas.

Na atualidade, a mídia nos inunda com milhares de produtos destinados a faixa etária de 2 a 15 anos, como softwares e brinquedos eletrônicos que prometem facilitar o processo de ensino-aprendizagem. Nesta idade as crianças gostam muito de jogos, que podem ser facilmente instalados em computadores. Muitos destes jogos são atraentes, chamativos, coloridos e podem ser classificados como educativos, pois incentivam a criatividade, o pensamento lógico, a memória e a capacidade de resolver problemas como o Logo, por exemplo. Outros, porém, apesar de serem desenvolvidos para crianças são bastante perigosos, pois trazem conteúdos

ligados à violência, associando o poder de destruição com a vitória no jogo, banalizando a violência e/ou promovendo a agressividade de forma contundente ou disfarçada.

Daí a necessidade de um olhar mais crítico acerca de alguns softwares que se dizem educativos, pois nem sempre o que está no rótulo representa de fato o seu conteúdo. Para efeito de exemplificação, um determinado software nacional de língua portuguesa, proprietário, um dos mais aclamados e vendidos do país, tem como cenário (pasmem!) um cabaré sideral com extraterrestres em trajes sumários e no melhor estilo francês, com dançarinas de Can-Can que a cada acerto do jogador fazem demonstração de seu “balé” como recompensa. Já a cada erro, a personagem que representa a criança que interage com o computador, recebe um tiro na face, parodiando o clássico Patolino. (Quem não lembra, na infância, do azarado Patolino, o pato que passava o episódio inteiro a fugir das balas de Hortelino, mas sempre levava o pior contra o coelho mais esperto do cinema, perdendo sempre seu bico ao final.) Como diria Patolino: “você é um ser desprezível”. É inaceitável que em pleno século XXI, educadores estimulem seus alunos a utilizar softwares proprietários, tendo opções livres, gratuitas e de qualidade indubitavelmente superior.

Exemplos como este mostram que nem todo programa de computador, embora tenha sido idealizado para este fim, é de fato educativo. A visão mais consensual na comunidade acadêmica é a de que todo software que apresenta uma metodologia que contextualize o processo ensino-aprendizagem, pode ser considerado educativo. Portanto, a classificação software educativo deve ter como critério a finalidade educativa para qual o programa é utilizado.

Desta forma, existem softwares educativos criados para fins educacionais e também aqueles criados para outros fins mas que acabam servindo a este propósito, devendo ser conceituados em referência à sua função, e não à sua natureza.

Assim, o software educacional é todo aquele que é utilizado de forma a contribuir com a aprendizagem. Por exemplo, o Calc, aplicativo da OpenOffice, que foi idealizado para ser uma planilha eletrônica, para uso em escritórios, em atividades que requeiram cálculos e afins, contudo, torna-se um software educativo, se no laboratório de Informática Educativa (LIE), a professora utilizar esse programa para ensinar ciências, criar gráficos, tabelas e planilhas nas aulas de matemática, criar exercícios com o auxílio deste software.

Da mesma forma, o Impress, editor de slides, pode

ser usado para criar apresentações sobre temas diversos como História Antiga ou álbuns digitais no mesmo estilo do Picasa dentre outras funções. Já o Writer, versão livre do Word, pode ser usado para fazer cruzadinhas, caça-palavras, estudo de texto e gramática, produção textual, confecção de jornal escolar, enfim, as possibilidades educativas são quase infinitas. Portanto, estes softwares originalmente não foram criados com fins educativos, mas podem ser considerados educativos, se dentro da proposta pedagógica da escola, tiverem como fim, auxiliar o processo de ensino-aprendizagem no contexto da informática educativa.

A Informática Educativa privilegia a utilização do computador como a ferramenta pedagógica que auxilia no processo de construção do conhecimento. Neste momento, o computador é um meio e não um fim, devendo ser usado considerando o desenvolvimento dos componentes curriculares. Nesse sentido, o computador transforma-se em um poderoso recurso de suporte à aprendizagem, com inúmeras possibilidades pedagógicas, desde que haja uma reformulação no currículo, que se crie novos modelos metodológicos e didáticos, e principalmente que se repense qual o verdadeiro significado da aprendizagem, para que o

computador não se torne mais um adereço travestido de modernidade.

O Software Livre oferece uma gama de alternativas que podem ser utilizados na educação. Na verdade, é cada vez mais crescente o número de tecnologias livres que são criadas e/ou podem ser adaptadas para educação. Em âmbito nacional cresce o interesse de estudos acerca dos benefícios do software livre na educação, destacando-se o LATES – Laboratório de Tecnologia Educacional e Software Livre mantido pela Universidade Estadual do Ceará.

Da mesma forma, o Ministério da Educação, tem investido em softwares livres, acreditando no seu potencial educativo, a exemplo da distribuição nacional Linux Educacional, sistema operacional sugerido para laboratórios de informática educativas das escolas públicas nacionais.

De fato, têm crescido o interesse das comunidades de mantenedores em trazer o pingüim para sala de aula. Um exemplo disso são os projetos Edubuntu, Kelix, Pandorga, Debian-Edu, Kde-edu, versões customizadas da GNU/Linux de cunho eminentemente educativo. Assim, nosso objetivo,

a partir das próximas edições, será destacar alguns destes projetos e softwares livres, discutindo-os com mais profundidade dentro do contexto educativo nacional.

As ferramentas computacionais, especialmente o software livre, podem ser um recurso rico em possibilidades que contribuem com a melhoria do nível de aprendizagem, desde que haja uma reformulação no currículo, que se crie novos modelos metodológicos, que se repense qual o significado da aprendizagem. Uma aprendizagem onde haja espaço para que se promova a construção do conhecimento. Conhecimento, não como algo que se recebe, mas concebido como relação, ou produto da relação entre o sujeito e seu conhecimento. No qual esse sujeito descobre, constrói e modifica, de forma criativa seu próprio conhecimento. Essa é a filosofia do software livre!

A única forma de produzir uma mudança efetiva na educação é através da revolução social e acreditar que o esforço coletivo ou individual de um professor pode sim fazer uma grande diferença. Construir de forma colaborativa o conhecimento é uma revolução silenciosa, no qual não existem

perdedores. Todos os setores da sociedade se beneficiam. Enfim, o grande desafio da atualidade consiste em trazer essa nova realidade para dentro da sala de aula, o que implica em mudar, de maneira significativa, o processo educacional como um todo. 

## Maiores informações:

**Projeto Software Livre Educacional:**

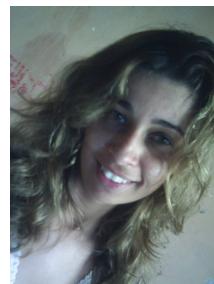
<http://sleducacional.org>

**Blog Software Livre na Educação:**

<http://softwarelivrenaeducacao.wordpress.com>

**Linux Educacional:**

<http://www.linuxeducacional.com>



**SINARA DUARTE** é professora da rede municipal de Fortaleza, pedagoga, especialista em Informática Educativa e Mídias em Educação, com ênfase no Software livre. Colaboradora do Projeto Software Livre Educacional e mantenedora do Blog Software Livre na Educação.



Festival Latino-americano de Instalação de Software Livre  
<http://flisol.aslgo.org.br/2009>

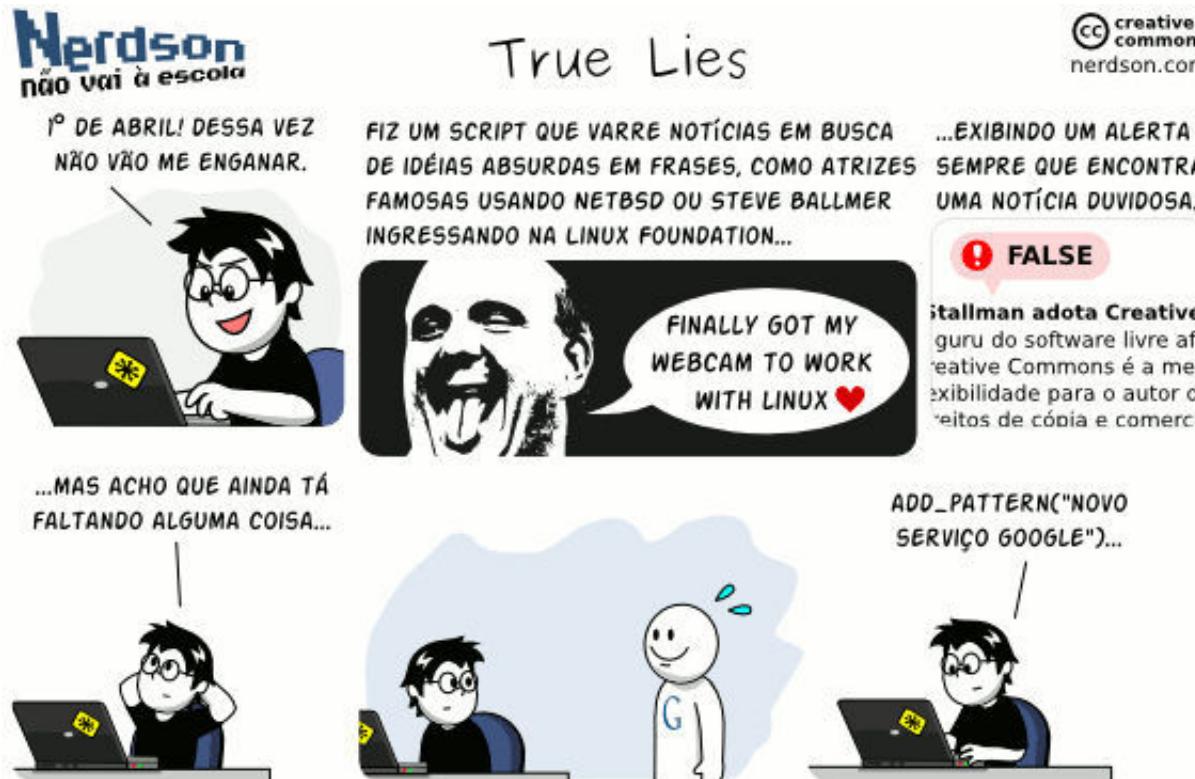
# QUADRINHOS

Por Rodrigo Leão e Karlisson de Macêdo Bezerra

## LINO E WINO



## NERDSON NÃO VAI À ESCOLA





DocumentFreedomDay

# DFD'09: 25 de Março Document Freedom Day

Por João Fernando e Lázaro Reinã

O Dia da Liberdade dos Documentos (DFD) é um dia global para Libertaçāo dos Documentos.

O Dia da Liberdade dos Documentos (DFD) é um dia global para Libertaçāo dos Documentos. É um dia de atividades de base ao redor do mundo com o objetivo de promover e desenvolver a consciência da relevância dos Formatos de Documento Livres em particular e dos Padrões Abertos em geral.

O Document Freedom Day - DFD - é apoiado por um grande grupo de organizações e indivíduos, incluindo, mas não limitado a Ars Aperta, COSS, Fundações Software Livre Europa e América Latina, ODF Alliance, OpenForum Euro-

pa, OSL, IMatix, Red Hat, Opentia, Sun Microsystems, IBM, Esoma, The Open Learning Centre, Estandares Abiertos.

No estado do Espírito Santo, a Iniciativa Espírito Livre e o Grupo de Usuários BrOffice.org estiveram envolvidos na organização de duas atividades para comemorar este dia.

## 25/03 - Escola Estadual Irmā Maria Horta

Na quarta-feira, 25/03/09, ocorreu o Documents Freedom Day, evento de dimensões



João Fernando (GUBrO-ES / Iniciativa Espírito Livre) na Escola Irmã Maria Horta

mundiais que tem como objetivo divulgar e conscientizar a sociedade quanto ao uso de documentos abertos.

Este ano o evento realizado em Vitória-ES teve duas vertentes, uma delas no dia 25/03 onde as atividades foram realizadas na Escola Estadual de Ensino Fundamental e Médio Irmã Maria Horta. Neste dia foram apresentadas duas palestras direcionadas aos alunos da escola, cujos temas foram voltados para a conscientização da importância do uso de padrões abertos.

A primeira palestra foi apresentada por João Fernando Costa Junior, líder do GUBrO-ES (Grupo de Usuários de BrOffice.org no ES), que pales- trou sobre "Conhecendo o BrOffice.org", apresentando as

vantagens de se utilizar o BrOffice.org e fazendo um paralelo entre os padrões abertos e os modelos proprietários. De maneira bem descontraída os alunos puderam absorver a apresentação de todas essas vantagens e benefícios. A palestra teve uma repercussão grande entre os alunos que permaneceram compenetrados no assunto até o fim da segunda palestra.

A segunda palestra foi apresentada por Romulo Sampaio Pires, estudante do curso de Sistemas de Informação, que pales- trou sobre "Documentos Abertos" completando a palestra anterior, ele falou da importância do uso de documentos abertos além de dar algumas informações sobre Software Livre em geral.

Ao fim das atividades na escola Irmã Maria Horta, os alunos pediram vários CD's, tanto do BrOffice.org quanto do Ubuntu GNU/Linux, mostrando assim que o objetivo do evento foi atingido ali.

### 28/03 - IFES Unidade Vitoria

Já no dia 28 de março, a Iniciativa Espírito Livre e o Grupo de Usuários BrOffice.org do Espírito Santo estiveram presentes organizando o DFD'09 [Dia da Liberdade dos Documentos] no auditório do IFES - Unidade Vitória (anteriormente chamado de CEFET-Vitória). O evento foi aberto ao público em geral e contou com a presença de alunos de cursos técnicos da Grande Vitória, universidades, prefeitura, entre outros.

Os trabalhos começaram às 08 da manhã, com a chegada dos participantes e o recebimento dos alimentos referente a inscrição. Em seguida, João Fernando, da Iniciativa Espírito Livre/GUBrO-ES, abriu oficialmente o DFD'09 neste dia, sugerindo discussões e propostas a partir do próprio evento em relação aos formatos abertos de documento. Hélio Ferreira, em seguida, inicia o ciclo de palestras às 09 da manhã com a palestra "A Importância Social dos Padrões Abertos, que como o próprio título sugere, mostrou a importância na sociedade dos padrões e formatos abertos,

amplamente difundidos por ferramentas livres.

A seguir, Gustavo Pacheco, da ONG BrOffice.org, que reside em Porto Alegre/RS, e que estava de passagem pelo estado do Espírito Santo, enriqueceu ainda mais o evento com a palestra BrOffice.org e Padrões Abertos, demonstrando aos presentes as diversas vantagens do software e do formato ODF.

Gustavo, apresentou ainda casos de sucesso do passado e presente, mostrando mais uma vez que é crescente o uso de tal ferramenta e a adoção dos padrões abertos de documento. Rubens Júnior fecha o ciclo de palestras com a palestra "Java e o padrão aberto", apresentando a linguagem e as vantagens de seu uso.

Em todas as palestras, houve uma boa participação dos presentes, em sua grande maioria estudantes e jovens, fazendo perguntas e mostrando interesse no assunto ali apresentado.

O término do evento contou com o sorteio de brindes, entre eles mídias de Linux, chaveiros e bolsas de estudo em cursos de Linux e Java na Point Treinamentos. O sorteio aconteceu de forma bem descontraída e animada, contando com a participação de todos nas brincadeiras e piadas habilmente preparadas pelo comedante-desenvolvedor Rubens Jr. O projeto iNDEX também



Gustavo Pacheco (BrOffice.org / ASL) no IFES Vitória

foi novamente mencionado assim como no LinuXmas 2008. Em breve haverá novidades a respeito.



**JOÃO FERNANDO COSTA JÚNIOR** é o líder do GUBrO-ES e responsável pela Iniciativa Espírito Livre / Revista Espírito Livre.

### Doação de alimentos

A entrega dos alimentos arrecadados durante o evento está prevista para acontecer entre a segunda e a terceira semana de abril, no Projeto Social Amor e Vida, localizado no Bairro Jardim América, em Cariacica/ES. 



**LÁZARO REINÃ** é usuário Linux, estudante C/C++, Lua, CSS, PHP. Integrante do EESL, ministra palestras e mini-cursos em diversos eventos de Software Livre.

The banner features the logos for 'ODF ALLIANCE' and 'BrOffice.org NBR ISO 26300'. It includes the slogan 'Liberte seus documentos' and 'DocumentFreedomDay'. It also provides links to 'http://www.documentfreedom.org' and 'http://dfd.espiritolivre.org'. There is a stylized blue leaf graphic on the right side.



# AGENDA

## ABRIL

**Evento: 2º CONSOLIP - Congresso de Software Livre do Interior Paulista**

Data: 08 a 11/04/2009

Local: Araras/SP

**Evento: 9ª Rio Wireless - Internet Móvel: A Nova Fronteira**

Data: 15/04/2009

Local: Rio de Janeiro/RJ

**Evento: Workshop Futuro da Internet**

Data: 15/04 e 16/04/09

Local: Campinas/SP

**Evento: Palestra Desenvolvimento de aplicativos com Adobe Flex**

Data: 15/04/09

Local: Porto Alegre/RS

**Evento: Palestra Carreira e Certificações em Banco de Dados Oracle**

Data: 15/04/09

Local: São Carlos/SP

**Evento: C&Tec - Mostra de Ciência e Tecnologia**

Data: 22/04 a 28/04/2009

Local: Campinas/SP

**Evento: Tchelinux 2009 Santo Ângelo**

Data: 25/04/2009

Local: Santo Ângelo/RS

**Evento: 14º EDTED - Encontro de Design Tecnologia Digital**

Data: 25/04/2009

Local: São Paulo/SP

**Evento: 1º Encontro do PHP MG**

Data: 25/04/2009

Local: Belo Horizonte/MG

**Evento: FLISOL - Festival Latino-Americano de Instalação de Software Livre**

Data: 25/04/2009

Local: Diversas cidades da América Latina

**Evento: Palestra Virtualização de Infra-**

**estrutura com Linux**

Data: 28/04/2009

Local: Piracicaba/SP

**Evento: Brazilian Road Show 2009 - Java EE Open Source**

Data: 29/04/2009

Local: Uberlândia/MG

## MAIO

**Evento: I FreeFESP**

Data: 04 a 06/05/2009

Local: Curitiba/PR

**Evento: Brazilian Road Show 2009 - Java EE Open Source**

Data: 05/05/2009

Local: Cuiabá/MT

**Evento: II EeBrO**

Data: 07 e 08/05/2009

Local: Osasco/SP

**Evento: Brazilian Road Show 2009 - Java EE Open Source**

Data: 20/05/2009

Local: Porto Alegre/RS

**Evento: Brazilian Road Show 2009 - Java EE Open Source**

Data: 27/05/2009

Local: Rio de Janeiro/RJ

**Evento: Free Software Bahia 2009**

Data: 28 e 29/05/2009

Local: Salvador/BA

**Evento: III ENSL e IV Festival**

Data: 29 e 30/05/2009

Local: Salvador/BA



# fisl10

## 10º Fórum Internacional Software Livre

A tecnologia que liberta

Edição Especial

24 a 27 de Junho de 2009  
Centro de Eventos PUCRS  
Porto Alegre/RS - Brasil

Maiores informações em:

[www.fisl.org.br](http://www.fisl.org.br)



Promoção



Organização



PROJETO  
SOFTWARE  
LIVRE - RS

PSL-RN  
Projeto Software Livre do  
Rio Grande do Norte

Realização



Transmissão

