

REVISTA  
*espírito*  
*livre*

LIBERDADE E  
INFORMAÇÃO

<http://revista.espiritolivre.org> | #013 | Abril 2010

**ENTREVISTA**

Entrevista com  
Richard Stallman e Brian  
Gough  
do Projeto GNU

**ENTREVISTA**

Entrevista com  
Luciano Ramalho,  
desenvolvedor Python



# PROJETO GNU

Free Software Foundation e muito mais!

**TECNOLOGIA**

O fim das suítes de  
escritório

**TUTORIAL**

Sensor de presença no  
Linux

**GRÁFICOS**

Criando um fantasma no  
GIMP

EDIÇÃO DE ANIVERSÁRIO - 1 ANO



## Atribuição-Us o 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 licenciente.



**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).

# 1 ano de vitórias e conquistas!

E cá estamos na edição de número 13, a edição de aniversário da Revista Espírito Livre. 12 meses se passaram em meio a alegrias e tristezas, dificuldades e tantos outros acidentes de percurso. Mas resistimos e mais fortes do que nunca nos firmamos junto a nossos leitores, que a cada mês esperam por uma nova edição. Nestes 12 meses tivemos muitas presenças ilustres em nossas páginas, nacionais e internacionais, algumas mais populares, outras nem tanto, mas todas de alguma forma, contribuindo para a democratização da informação através de matérias, relatos, tutoriais, entrevistas e o que mais couber em nossas páginas.

Nesta edição trazemos em nossa capa, um grande e imponente GNU, mascote de um projeto que ao longo dos anos vem se fortalecendo através de novos colaboradores e nossos sub-projetos. O Projeto GNU e a Free Software Foundation fazem parte do cotidiano de muitas comunidades e seus representantes, em especial a figura de Richard Stallman, nosso entrevistado juntamente com Brian Gough, que fazem do projeto GNU e a FSF, parte de nossas vidas. Stallman é amado por muitos e odiado por outros, mas sempre está ali, presente e ativo em sua posição de evangelizador e um ícone da comunidade de software livre mundial. Stallman e Brian Gough, também do Projeto GNU, concederam uma entrevista bem interessante a nossa equipe, brilhantemente representada pelo colunista Alexandre Oliva, ativista conhecido por aqui em nosso Brasil, e no restante da América Latina também. Os entrevistados esclarecem diversas dúvidas e questões que sempre povoam o imaginário de muitos membros das comunidades de software livre.

Krix Apolinário, juntamente com Alexandre Oliva, Cesar Taurion, Cárliston Galdino e Roberto Salomon trazem cada um em sua coluna, do seu jeito, excelentes contribuições. Não somente os colunistas, mas também todos os outros colaboradores, cada um a seu modo, contribuiu com uma parcela para que tivéssemos uma edição digna de edição de aniversário. A capa, é arte do Cadunico, nosso artista. Walter Capanema e Luis Henrique Silveira apresentam dois artigos apresentando aspectos jurídicos de duas situações bastante polêmicas. Capanema trata da pirataria enquanto Luis Henrique fala da nova licitação do Programa Professor Digital, amplamente coberto pelas mídias e assunto do momento. Além da entrevista ligada ao tema de capa, conversamos Juliana Kryszczun conversou com Luciano Ramalho, desenvolvedor em Python. Jomar Silva, sem papas na língua, declara a morte das suítes de escritório, mas com um enfoque bastante consciente e fundamentado. Patrick Amorim fala sobre a Tecnologia PLC, que provê banda larga através da rede elétrica. E que tal controlar a proteção de tela do seu computador através de bluetooth? Marcelo Moreira Mello mostra o caminho das pedras. Filipo Tardim também apresenta como fazer uma remasterização do Ubuntu, do zero. Enquanto Wagner Emmanoel faz um review sobre a nova edição do Ubuntu que deve ser liberada essa semana. Igor Morgado também traz um artigo bem interessante sobre gerenciamento de redes enquanto Francilvio Alff fala sobre gestão do conhecimento e ferramentas wiki. Eliane Domingos e Igor Morgado, ambos da Gnutech, meus agradecimentos, que também se estendem a nossos parceiros que estão sempre nos ajudando.

Carlos Eduardo (o nosso artista) também mostra em um tutorial bem interessante como criar um fantasma utilizando o GIMP. Assombroso! E outro Carlos Eduardo, que não é artista mas um escritor de mão cheia, juntamente com a própria Revista Espírito Livre apresenta seu livro intitulado Ubuntu - Guia do Iniciante, que será lançado nesta semana, juntamente com o lançamento oficial do Ubuntu 10.04.

Todos os outros colegas que contribuíram com materiais mas que não listados aqui não foram esquecidos, aliás serão lembrados sempre por mim e por nossos leitores que a cada mês fazem o download das edições com sede de conhecimento. Meus sinceros agradecimentos a todos vocês. Vocês ajudaram a construir uma edição recorde em número de páginas e em conhecimento.

E agora é a hora de soprar a velinha! Eu já fiz meu pedido. Você já fez o seu?!



**João Fernando Costa Júnior**  
Editor



## EXPEDIENTE

Diretor Geral  
João Fernando Costa Júnior

Editor  
João Fernando Costa Júnior

Revisão  
Eliane Domingos

Tradução  
Paulo de Souza Lima  
Francilvio Roberto Alff

Arte e Diagramação  
João Fernando Costa Júnior  
Igor Morgado

Capa e contra-capas  
Carlos Eduardo Mattos da Cruz

Contribuíram nesta edição  
Alessandro Silva  
Alexandre Oliva  
André Farias Oliveira  
André Gondim  
Antônio Misaka  
Brian Gough  
Cárliston Galdino  
Carlos Eduardo do Val  
Carlos Eduardo Mattos da Cruz  
Cezar Taurion  
Cindy Dalfovo  
Fernando Brito  
Fernando Medeiros  
Filipo Tardim  
Francilvio Roberto Alff  
Gludson Junior  
Igor Morgado  
Jandson Damasceno Rabelo  
João Fernando Costa Júnior  
Jomar Silva  
Juliana Kryszczun  
Krix Apolinário  
Luciano Ramalho  
Luis Henrique Silveira  
Marcelo Moreira Mello  
Márcio Pivoto  
Patrick Amorim  
Paulo de Souza Lima  
Rafael Cirolini  
Richard Stallman  
Roberto Salomon  
Wagner Emmanoel  
Walter Capanema  
Wesley Samp

Contato  
revista@espiritolivres.org

O conteúdo assinado e as imagens que o integram, são de inteira responsabilidade de seus respectivos autores, não representando necessariamente a opinião da Revista Espírito Livre e de seus responsáveis. Todos os direitos sobre as imagens são reservados a seus respectivos proprietários.

# SUMÁRIO

## CAPA

- 34** **A licença "Seja Legal"**  
Que tal?
- 38** **Uma questão de nome**  
Gnu or not Gnu?
- 41** **Tradução também é GNU**  
The books on the table...
- 43** **Compartilhar...**  
O que eu ganho com isso?

## COLUNAS

- 13** **Mitologia Grega III:**  
Minotauro
- 17** **Warning Zone**  
Episódio 7 - Concorrência
- 20** **Métodos e regras**  
Ao gerenciar projetos de Open Source
- 25** **Gadgets femininos**  
Como está o mercado?
- 27** **Velinhas!**  
Parabéns para nós...

## JURIS

- 45** **Justiça seja feita!**  
Programa Professor Digital...
- 48** **Pirataria:**  
Como enfrentá-la

## ENTREVISTA

- 51** **Luciano Ramanho**  
Desenvolvedor em Python

**Entrevista exclusiva  
com Richard Stallman  
e Brian Gough,  
do Projeto GNU**

**PÁG. 29**



**Entrevista exclusiva  
com desenvolvedor  
em Python, Luciano  
Ramalho**

**PÁG. 51**



**124 AGENDA**



**06 NOTÍCIAS**

---

## TECNOLOGIA

---

- 54** O fim das suítes de escritório  
Será... só imaginação?
- 59** PLC  
Dados pela rede elétrica

---

## TUTORIAL

---

- 63** Sensor de presença no Linux  
Ativando sua proteção de tela
- 67** Criando um LiveCD do Ubuntu  
A partir do zero! Vai encarar?!
- 74** Google Reader no Ubuntu  
Fácil fácil...

---

## SYSADMIN

---

- 77** GIT:  
Fazendo controle de versões

---

## REVIEW

---

- 82** Ubuntu 10.04:  
Conheça as novidades

---

## REDES

---

- 89** O zen e a arte...  
... do gerenciamento de redes

---

## GESTÃO

---

- 96** Knowledge is Power!  
E é mesmo...

---

## DESENVOLVIMENTO

---

- 100** Wordpress 3.0 beta  
Liberado para download

---

## BANCO DE DADOS

---

- 102** Firebird  
Conheça o pássaro de fogo

---

## LANÇAMENTO

---

- 106** Ubuntu - Guia do Iniciante  
Autor: Carlos Eduardo do Val

---

## GRÁFICOS

---

- 107** GIMP Sobrenatural  
Criando um fantasma... Ai que medo!
- 111** Edição de Vídeo  
Com Cinellera...

---

## EDUCAÇÃO

---

- 115** TCC sobre Software Livre  
Vamos ajudar?!

---

## EVENTOS

---

- 117** DFD 2010 - Vitória/ES  
Relato de evento
- 120** IV ENSOL - João Pessoa/PB  
E você, vai participar?!

---

## QUADRINHOS

---

- 123** Suporte\_  
Os levados da Breca

---

## ENTRE ASPAS

---

- 124** Citação de Richard Stallman



10 LEITOR



12 PROMOÇÕES

# NOTÍCIAS

Por João Fernando Costa Júnior

## OSMPlayer, um player multimídia em HTML5



A chegada do HTML5 é para muitos o começo do fim do reinado do Flash como um framework multimídia na web. Analisando o projeto Open Media Player parece ser verdade. OSM é um projeto Open Source, com o objetivo de criar um player

multimídia baseado em HTML5 e usando a biblioteca jQuery. Algumas de suas características são: 100% Open Source e 100% grátis; Usa o framework JavaScript jQuery; Conteúdo baseado em HTML5; Suporte para áudio e vídeo em formatos populares; Capacidade de mudar o tema (com base na ThemeRoller); Integração com o Vimeo e Youtube e suporte para playlist. Mais informações [aqui](#).

## Lançado Beta do Red Hat 6.0



A Red Hat, lançou a público sua versão “open beta” do seu produto RHEL 6.0. Um dos pontos chave desta versão é a capacidade de migração de máquinas antigas RHEL 5 serem migradas do Xen para o KVM (que atualmente é mantido e é um produto da Red Hat) e assim, os usuários podem escolher se migram ou não de plataforma de virtualização. Funcionalidades do 6.0 terão “backports” para a série 5 do produto, mas lembrando que algumas funcionalidades podem não ser totalmente compatíveis com o kernel novo que será introduzido, sendo descartado o backport de tais funcionalidades, por quebra de compatibilidade. Junto com a notícia, os “tarados de plantão dos benchmarks” do

Phoronix já fizeram uma comparação entre o Red Hat 6.0 beta, CentOS 5.4, Fedora 12 e Ubuntu 10.04 LTS que pode ser conferida [aqui](#).

## Netbooks com Chrome OS custarão US\$400



Eric Schmidt, CEO do Google, declarou que os netbooks com Chrome OS chegarão ao mercado no começo de 2011, custando menos de 400 dólares. Recentemente, durante a Atmosphere Cloud Computing, Schmidt admitiu que

a empresa não tem muito controle sobre os preços finais dos produtos, entretanto ressaltou que os valores são determinados pelos custos dos componentes dos netbooks – incluindo aí o software. Como o Chrome OS e Android são gratuitos, não há taxa associada a eles que possa encarecer o produto final. Schimidt espera que sejam mantidos os preços atuais para netbooks, que variam de US\$300 a US\$400.

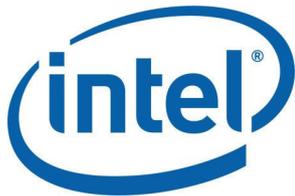
## Google publicará codec VP8 como software livre



O Google em breve estará liberando seu codec de vídeo, chamado VP8, como software livre. A companhia deve fazer o anúncio oficial na conferência para desenvolvedores Google I/O no próximo mês. Com

este lançamento as equipes de desenvolvimento do Mozilla Firefox e Google Chrome devem anunciar o suporte a vídeo no HTML5 usando o novo codec.

## Intel prepara versão do Meego para laptops e desktops tradicionais



Além de netbooks e smartphones, o sistema operacional Meego, uma distribuição GNU/Linux, baseada na combinação dos sistemas operacionais Moblin, da Intel e do Maemo, da Nokia, vai rodar em notebooks e desktops tradicionais com processadores Intel Atom e alguns modelos da família Core. A informação é de Doug Fisher, vice-presidente do grupo de software e serviços da Intel, em entrevista recente a jornalistas do grupo IDG. Segundo Fisher, a Intel planeja lançar uma edição de Meego para os chamados modelos de entrada, máquinas mais baratas, de menor poder de processamento. Mas não revelou modelos, nem possíveis datas para o lançamento. O maior esforço de desenvolvimento em torno Meego está focado em netbooks, smartphones (com processadores ARM, mas também IntelMoorestown) e dispositivos como TVs e set-top boxes. A edição Meego para esses aparelhos de consumo provavelmente será lançada em outubro. Já para netbooks, o lançamento está previsto para o início do segundo semestre deste ano, segundo Fisher.

## Google contratará desenvolvedores para manter Android sincronizado com kernel.org



Chris DiBona, que é o responsável pela área de software livre do Google, confirmou no "Linux Collaboration Summit" que o Google precisa de fazer melhor, contribuindo com patches ao Linux kernel e anunciou que abriram um processo de seleção para encontrar dois candidatos para fazer este trabalho. A comunidade do Linux kernel decidiu no início do ano eliminar código do Android alegando que não se obtinha do Google suficientes patches, nem colaboração.

## Oracle começa a cobrar por ODF plugin



A Oracle, que adquiriu a Sun recentemente, bem como todo seu portfólio de produtos, começou a cobrar US\$90 por usuário pelo Microsoft Office ODF plugin - exigindo mínimo de 100 licenças. Apesar do plugin nunca ter sido open source, sempre esteve disponível para download gratuito. A quem diga que é uma estratégia para que os usuários utilizem sua suíte de escritório, de código aberto, o OpenOffice.org, visto que este trabalha com o formato já por padrão. Entretanto, alguns temem tal medida e a consideram precipitada ou simplesmente o resultado do que pode vir a acontecer com o seus outros produtos recém-adquiridos.

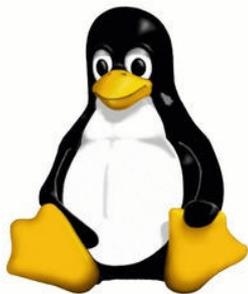
## Wikipedia amplia plataforma e ganha espaço para vídeos



WIKIPEDIA

A enciclopédia virtual Wikipedia vai ganhar espaço para vídeos na plataforma, segundo informou em comunicado oficial na quarta-feira (21). "Agora, muitos poucos artigos da Wikipedia têm vídeos. É hora de mudar isso! Começando agora, você pode estar entre as primeiras pessoas a experimentar com as possibilidades de vídeo colaborativo. Suas contribuições poderão modelar o futuro da Wikipedia, fazendo [dela] uma fonte rica e dinâmica de conteúdo educacional de alta qualidade", diz o texto. O comunicado diz que, "quando você posta um vídeo na Wikipedia, você também ajuda a promover o vídeo aberto". O site diz ainda que vai usar 100% de vídeos gratuitos e de código aberto, sob tecnologia HTML5, o que é um avanço. A Wikipedia instrui sobre como postar um vídeo no seguinte endereço <http://videoonwikipedia.org/howto.html>.

### TeamViewer: acesso remoto no desktop de Linux para Linux



Até pouco tempo se alguém procurasse algum programa voltado para Linux destinado ao suporte remoto, sem ter que redirecionar alguma porta (como no caso do VNC) para assim obter acesso e de baixa qualidade, obviamente o usuário ou mesmo técnico

da área não encontraria absolutamente nada. Programas como o TeamViewer, LogMeIn dentre outros, não são novidades para a plataforma Windows ou até mesmo MAC. Neste caso, para o Linux, sim. Isso é uma grande novidade pois, até onde se sabe, se este seria um dos únicos a desempenhar esta tarefa, com suporte ao Linux. O TeamViewer, que ainda está em sua versão beta, inicialmente possui pacotes disponíveis para Red Hat, Fedora, Suse, Mandriva, Ubuntu e Debian, contendo versões para 32 e 64 bits. Também está disponível um pacote genérico (tar.gz), que pode ser utilizado em qualquer distribuição nas arquiteturas 32 e 64bits. Vale lembrar porém que, para uso não comercial a utilização do TeamViewer é gratuita. Para usá-los comercialmente, entre em contato com o desenvolvedor no site oficial do produto.

### Adobe desiste dos planos para levar Flash ao iPhone e iPad



A Adobe deixará de lado a continuidade dos seus planos para levar o Flash para o iPhone e o iPad, ambos fabricados pela Apple. De acordo com o site da revista "Wired", a Adobe afirmou na noite da última terça-feira (20) que

abandonou o investimento em uma ferramenta de software que permitiria aos desenvolvedores a aplicar o Flash nos aparelhos da Apple. "O prin-

cipal objetivo do Flash sempre foi habilitar a plataforma e desenvolvimento para múltiplos navegadores. Isto é exatamente o oposto do que a Apple quer. Eles querem amarrar os desenvolvedores sob a sua plataforma, e restringir suas opções para tornar difícil para os desenvolvedores mirarem outras plataformas", escreveu o gerente de produtos da Adobe, Mike Chambers. Em fevereiro, Steve Jobs, o conhecido executivo-chefe da Apple, disse que a Adobe era "preguiçosa" porque "tem todo o potencial para fazer coisas interessantes, e se recusa a fazer isso". A tecnologia Flash é problemática para a Apple, segundo Jobs, porque trava o computador Macintosh. Jobs disse ainda que a Apple não a apoia porque é falha. E fez a previsão de que "ninguém mais vai usar Flash" e que "o mundo está se direcionando para o HTML5".

### Apache.org sofre ataque



O Time de Infraestrutura do projeto Apache relatou um ataque direto e direcionado contra o servidor que hospeda seu software de ras-

treamento de bugs. "Se você for um usuário do JIRA, Bugzilla ou Confluence, uma cópia em hash de sua senha foi comprometida. JIRA e Confluence ambos usam um hash SHA-512, mas sem um 'salt' aleatório. Acreditamos que o risco de haverem senhas simples baseadas em ataques de dicionário é um tanto alta e a maioria dos usuários deveria mudar suas senhas. O Bugzilla usa SHA-256, incluindo um 'salt' aleatório. O risco para a maioria dos usuários é de baixo a moderado, já que senhas baseadas em dicionário não são eficazes, mas ainda assim é recomendável alterá-los. Adicionalmente, se você logou na instância Apache JIRA entre 6 de abril e 9 de abril, você deveria considerar sua senha como comprometida, já que os atacantes alteraram o formulário de login para logar as senhas."

### Televisores Panasonic com FreeBSD



A documentação oficial indica que algumas TVs Panasonic usam FreeBSD para o seu firmware de alguns de seus televisores de plasma da série VIERA G20, G25 e VT). Segue um trecho dos termos de licenciamento dos produtos: "The Software (defined below) contains a number of individual copyrighted open source software programs, such as FreeBSD. Please refer to "Software Licence" menu on Product for applicable license terms." Interessante não é mesmo? Vale lembrar que televisores de outras marcas como Sony e Samsung trazem GNU/Linux em seus firmwares.

### Pesquisadores conseguem espionar celulares

Dois pesquisadores de segurança, Nick DePetrillo e Don Bailey, descobriram uma forma de explorar as brechas no sistema de telecomunicação móvel para espionar pessoas. Tudo de forma legal, alegam eles. De acordo com os pesquisadores, a falha é encontrada quando eles reuniram um número-alvo de telefone celular a partir de uma base de dados pública que liga nomes aos números para fins de identificação do chamada, o também chamado "bina". Eles usam um software PBX open-source para disfarçar o identificador de chamadas e, em seguida, o telefone automaticamente chama a si mesmo, forçando o sistema a fazer uma pesquisa do nome. Mais informações em <http://ur1.ca/wwoi>.

### Criador de Doom 3 fará palestra no Brasil



No dia 10 de maio, a Cultura Inglesa promoverá duas palestras sobre a construção de histórias para videogames. A primeira, que

faz parte do 14º Cultura Inglesa Festival, acontecerá às 9h com a professora-doutora Renata Gomes, que fará uma apresentação com o tema "Videogame: imagem, narrativa". No mesmo dia, às 19h, será a vez do roteirista Matt Costello, que falará sobre história, gameplay e os desafios de criação para os jogos eletrônicos. Ele é um dos nomes por trás do game Doom 3, de 2005, e do clássico The 7th Guest, lançado em 1993. Inscrições podem ser feitas pelo telefone 3095-4466. Mais informações pelo site [www.culturainglesasp.com.br/festival](http://www.culturainglesasp.com.br/festival).

### Google bloqueado em 25 dos cem países nos quais atua

O Google afirmou, em um post publicado em blog oficial, no dia 19/04, que dos cem países para os quais oferece serviços, no mínimo 25 bloqueiam de certa forma os produtos fornecidos pela empresa. "Nós enxergamos esses esforços no controle de muitas formas. A China é o exemplo mais polarizado, mas não é a única. Os produtos do Google --de buscas e blog até YouTube e Google Docs-- têm sido bloqueados em 25 de cem países onde nós oferecemos nossos produtos", diz o Google. Segundo o site [TechCrunch](http://TechCrunch), há por volta de 40 países que censuram algum tipo de conteúdo na internet, atualmente.

### Sharp lança tablet movido a Linux

A Sharp anunciou mais um produto na sua linha de ultraportáteis, batizada de Netwalker. O Netwalker PC-T1 é uma versão sem teclado do "mini netbook" Netwalker PC-Z1 lançado em agosto passado. Segundo o site [SlashGear](http://SlashGear) o Netwalker PC-T1 tem um processador ARM Freescale i.MX515 de 800 MHz, 8 GB de espaço interno (expansível com cartões microSD), interface de rede WiFi 802.11 b/g, Bluetooth e o conjunto tradicional de portas USB.

# EMAILS, SUGESTÕES E COMENTÁRIOS



Ayhan YILDIZ - sxc.hu

Um ano se passou. Muita gente nos acompanha desde a primeira edição, e a estes nossos sinceros agradecimentos. Muitos, ao longos dos meses, ficaram sabendo da publicação e a partir daí começaram a nos seguir e nunca mais pararam. Gente de todo o Brasil que participa de nossas promoções, que nos encontra nos eventos, que mandam emails recheados de elogios. Mesmo completando apenas um aninho, gostaríamos que essa troca de informações continuasse, pois ela é que nos leva ainda mais longe!

Então não fique com vergonha: diga-nos o que achou da última edição ou das últimas matérias! Algo não ficou legal? Alguma matéria lhe ajudou muito? Ficou satisfeito por ter encontrado o que procurava? Então manifeste-se e mostre a nós e aos demais leitores o quão importante é ter o "espírito livre". Abaixo listamos alguns comentários que recebemos nos últimos dias:

Achei muito boa, principalmente pelo conteúdo e finalização, são muito bons. E fico muito contente que existem pessoas trabalhando para nos dar esse tipo de conhecimento.

**Alayne de Oliveira Borges - Marituba/PA**

Seguindo o mundo livre a revista está de parabéns pelas reportagens, material muito bem elaborado. Já venho a algum tempo acompanhando e já está dentre as minhas favoritas.

**Felipe Santos Barbosa - Teresina/PI**

Maravilhosa, ótimos conteúdos, matéria muito bem esclarecidas, uma revista bem objetiva e inteligente, parabéns aos colunistas e os produtores dessa magnífica revista.

**Guilherme Rodrigues Soares - Caraguatuba/SP**

Excelente iniciativa a ser seguida pela Comunidade Open Source.

**Irineu Porfirio da Costa - Varzea Grande/MT**

Uma ótima iniciativa para trazer informação relevante para a Comunidade. Sempre com boas entrevistas e colunas.

**Debora Thomaz Martins - Caldas Novas/GO**

Um das melhores revistas relacionado a comunidade de software livre.

**José Eilton dos Santos Costa - Escada/PE**

Num mundo com diversos caminhos que nos deixam em dúvidas, encontrar um guia confiável e experiente para, além de nos ensinar o caminho a seguir, também vai junto segurando em nossa mão, é um privilégio sem igual. É isso o que acho da Revista Espírito Livre! Parabens a toda equipe!

**Herdiley Alves Coelho - Uruçuí - PI**

Muito instrutiva! Adoro as dicas e reportagens.

**Michel Martins Valtuille - Goiânia/GO**

Acho uma excelente ideia, pois com a veiculação da revista pessoas leigas sobre o assunto e que resistem em saber o que é linux e usar linux, vão poder entender de forma suave, e se apaixonar por software livre e assim criar laços de amizade por todo mundo, disseminando informação. "Isso é ser livre", "isso é software livre."

**Leandro P. Moreira - Ribeirão das Neves/MG**

Um ótima revista sobre Linux e GNU. O melhor é o fato de ser totalmente livre.

**Carlos Roitman Maceno - São Paulo/SP**

Ótima fonte de pesquisa de novas tecnologias para implementação em meu trabalho. Além disso, sempre fico sabendo de novos eventos da área de tecnologia e notícias do mundo da informática.

**Rubia Karla Ramos - João Pessoa/PB**

Uma excelente iniciativa no intuito de pregar a liberdade de criação, acesso e divulgação do conhecimento, seja ele tecnológico, cultural ou artístico. Parabéns a todos.

**Paulo Rômulo Barros - Campina Grande/PB**

Muito boa, onde sua atitude livre e inovadora deveria servir de exemplo para muitas outras revistas.

**Patrick Renilton dos Santos - Maceió/AL**

Uma das melhores revistas sobre tecnologia da atualidade. Com a vantagem de ser completamente acessível e focada no Software Livre.

**Brunno Pereira - Jaboatão dos Guararapes/PE**

A melhor no segmento eletrônico e em conteúdo.

**Bruno Torres Viana - Vitória/ES**

A Espírito Livre, ajuda não só a mim, mas também todos os meus amigos alguns até que não são da área de informática. Ajudo divulgando a revista na minha rede de contatos para que os

mesmos possam ter um conhecimento com fontes confiáveis e de boa base.

**Diogo Alvez da Silva - Belo Horizonte/MG**

Uma ótima iniciativa por parte da comunidade de software livre nacional. Seu conteúdo, artigos e tutoriais são ótimos para quem está começando a trilhar os caminhos do software livre e também para quem, como eu, já desfruto faz tempo.

**Biliane Moreira Dantas da Silva - Maceió/AL**

A Revista Espírito Livre nos dá acesso a um mix de informações, mostrando seriedade e credibilidade em todas as matérias que nos oferece. Disponibiliza ao leitor um grande crescimento intelectual.

**Larissa Araújo de Alencar - João Pessoa/PB**

A Revista Espírito Livre é uma ótima iniciativa, pois ajuda a divulgação do Software Livre de forma abrangente pela web. Acho uma pena não ter tempo para ler toda a revista, pois normalmente só leio algumas passagens que acho mais interessantes.

**Helton de Melo Duarte - Natal/RN**

É uma revista bem legal que abrange várias áreas do mundo livre, acho que devia falar mais de Python =D Eu sinceramente acompanho desde a #1.

**Rodrigo Cesar Lira da Silva - Paulista/PE**

"A consolidação do Software Livre hoje é devido ao poder da colaboração de muitas empresas e pessoas envolvidas". E a Revista Espírito Livre, está fazendo sua colaboração de tão grande importância para a comunidade brasileira. Meus parabéns!

**Sávio de Oliveira Sousa - João Pessoa/PB**

Ótima para manter atualizado sobre a comunidade de Software Livre e ter um bom aprendizado técnico.

**André Miranda - Aparecida de Goiânia/GO**

# PROMOÇÕES

Na edição #012 da Revista Espírito Livre tivemos diversas promoções bem como promoções através de nosso site e canais de relacionamento com os leitores, entre eles a Promoção Relâmpago do IV ENSOL, que acontece em João Pessoa/PB. Quem estava atento conseguiu participar da promoção do IV ENSOL, então se você não foi um dos sorteados, não desanime. Quem sabe na próxima edição seu nome não aparece logo abaixo?! Mais promoções estão a caminho...

Abaixo, segue a lista de ganhadores de cada uma das promoções.



## Ganhadores da Promoção VirtualLink:

1. Michel Martins Valtuille - Goiânia/GO
2. Herdiley Alves Coelho - Uruçuí/PI
3. José Eilton dos Santos Costa - Escada/PE
4. Debora Thomaz Martins - Caldas Novas/GO
5. Leandro Paulo Moreira - Ribeirão das Neves/MG

## Ganhadores da promoção Clube do Hacker:

1. Guilherme Rodrigues Soares - Caraguatatuba/SP
2. Felipe Santos Barbosa - Teresina/PI
3. Alayne de Oliveira Borges - Marituba/PA

## Ganhadores da promoção IV ENSOL:

1. Paulo Rômulo Alves Barros - Campina Grande/PB
2. Fabiana Furtado - João Pessoa/PB
3. Larissa Araújo de Alencar - João Pessoa/PB

**TreinaLinux** 

www.treinalinux.com.br

A TreinaLinux em parceria com a Revista Espírito Livre estará sorteando kits de DVDs entre os leitores. Basta se inscrever neste [link](#) e começar a torcer!

**VirtualLink**   
Soluções e Treinamentos em Linux

A promoção continua! A VirtualLink em parceria com a Revista Espírito Livre estará sorteando kits de cds e dvds entre os leitores. Basta se inscrever neste [link](#) e começar a torcer!

 **DESTAQUE-SE**  
entre para o clube do hacker

Não ganhou? Você ainda tem chance! O Clube do Hacker em parceria com a Revista Espírito Livre sorteará associações para o clube. Inscreva-se no [link](#) e cruze os dedos!



# Mitologia Grega III: Minotauro

Por Alexandre Oliva

wallyg - Flickr.com

Há milênios, a lenda do Minotauro vem sendo contada para ensinar o perigo de trair e enganar os poderosos. Deve ser o método de dominação mais antigo e mais eficaz: demonizar quem vê a saída do labirinto, para que o restante do rebanho não o siga até a liberdade. Após tal lavagem cerebral, quem ousaria questionar o mito de que o Minotauro é um monstro repulsivo e perigoso?

Era uma vez o trono do reino insular de Creta disputado por dois irmãos. Minow\$, o mais inescrupuloso e sedento de poder, pediu que Poseidon, deus dos mares, interviesse por ele. Já coroadado, viu sair do mar um bellissimo bovídeo branco, que deveria sacrificar ao deus. Sacrificou outro animal, mas Poseidon percebeu e puniu o traidor: providenciou para que Pasífae, esposa de Minow\$, se apaixonasse pelo bovídeo, e para que, da união carnal dos dois, meca-

nicamente possível graças ao engenhoso hacker Dédalo, nascesse uma criatura metade humana, metade bovídea. (Leitores curiosos já devem estar se perguntando: seria Richard Minotauro Stallman, o homem-gnu?)

Por capricho do tirano, aquele bastardo cretino, deveria a criatura viver prisioneira, num labirinto que o mesmo Dédalo seria obrigado a construir, debaixo do palácio R\$ de ©nosso\$. A cada ano, Minow\$ jogava sete jovens e sete damas no labirinto, para servir de alimento ao Minotauro. Teseu, um desses jovens, entrou no labirinto com um novelo de lã para marcar o caminho de volta (ideia de ninguém menos que Dédalo), derrotando o suposto monstro e libertando os jovens e damas que ali encontrou.

O verdadeiro monstro, Minow\$, ainda aprisionaria no labirinto o hacker Dédalo e seu filho, Ícaro. Recuperariam sua liberdade com outro hack: colando asas aos seus braços, com cera de abelha. Diz a lenda que Ícaro gostou tanto da liberdade que voou alto demais; tão alto que o Sol derreteu a cera e, sem asas, Ícaro caiu sobre o mar e morreu.

Ora... Bastam algumas perguntas, fruto de análise crítica, para verificar que se trata de um mito, uma lenda para assustar e conter os amantes da liberdade. Como assim, se voar muito al-

to, esquenta, a ponto de derreter a cera das asas? Todo mundo sabe que, quanto mais alto, mais frio e rarefeito fica o ar!

Como assim, Teseu libertou os jovens e damas que encontrou no labirinto? Não eram comida para o Minotauro? Se ele não os devorava, vivia de quê? E os jovens, como sobreviveram? E a cera, penas e madeira para as asas de Dédalo e Ícaro, vieram de onde? Seria Dédalo um hacker poderoso como o MacGiver, de “Profissão: Perigo”, a ponto de tirar madeira da cara de pau de quem inventa estas histórias, juntar com as penas às quais Dédalo e Ícaro foram condenados, fazer uma cera na entrada da área e pronto!, é só colar e sair voando? Tenha DOS!

Além do mais, os bovídeos eram idolatrados em Creta. O palácio R\$ de ©nosso\$ era todo decorado com peças e obras de arte que remetiam ao tema bovídeo e suas divindades. O gado ainda garantia “pão e circo”: a diversão nas touradas acrobáticas, comparáveis a alguns rodeios dos pampas brasileiros; a carne orgânica diretamente dos “pam-

pas”, para alimentar o corpo; para o espírito, os sacrifícios religiosos de bovídeos nos rituais minoanos. Dá até pra imaginar que o gélido vento Minuano, que sopra nos pampas sul-americanos, deva seu nome aos tão importantes ventos dos quais dependiam as navegações minoanas.

“ Há milênios, a lenda do Minotauro vem sendo contada, como exemplo do perigo de trair e enganar os poderosos. Deve ser o método de dominação mais antigo e mais eficaz: demonizar quem vê a saída do labirinto, para que o restante do rebanho não o siga até a liberdade.

Alexandre Oliva

O simbolismo da lenda é claro, e chama atenção a presença, também na simbologia, da dualidade divino-pecuária bovídea. Minow\$, egoísta, traiçoeiro, ganancioso e sedento de poder, queria só para si o divino, belíssimo bovídeo branco (“Ô, Montanha, pó pará!”, intervém Maçaranduba,

“Tá duvidando da real masculinidade? Vou dar... porrada!”), o trono e a coroa que o bovídeo faturou. Fez de tudo para aprisionar aquele que nasceu metade humano, metade bovídeo, bem como os súditos e prisioneiros que, impotentes, não resistiam ao subjugo, ao controle e aos abusos do tirano, reduzindo-se assim a gado humano: também metade humanos, metade bovídeos. E aí de quem ousasse desafiar o tirano: seria feito prisioneiro e atirado no labirinto de Minow\$. E aí daquele que sonhasse voar, símbolo de desejo de liberdade: perderia as asas e a vida no mar de Poseidon! Vida de gado, nada de hackear!

Removendo as alegorias mitológicas, o que resta é uma história que deve ter sido mais ou menos assim: Pasífae teve um filho que não se parecia nem um pouco com Minow\$. A própria lenda aponta para o provável pai, aquele que tinha intimidade suficiente para ajudá-la a engravidar, mas, francamente, de um bovídeo?!? O ser que nasceu pode muito bem ter sido o próprio Ícaro, de mãe e origem desconhecidas. Foi anunciado ao público como monstro e escondido.

Dédalo já havia iniciado o movimento social humanitário SoL, com ideais de solidariedade social, respeito ao próximo e às liberdades essenciais. Hacker que era, encontrou uma forma de cumprir a ordem do tirano, de construir um labirinto nas masmorras do palácio R\$ de ©nosso\$, sem cercear a liberdade dos que seriam injustamente feitos prisioneiros ali.

Construiu-o de tal forma que apenas quem fosse movido pelo egoísmo, pela ganância e pelo impulso de cercear a liberdade de outros teria a sensação de estar preso nas masmorras de ©nosso\$. Inaugurou assim o projeto tecnológico GNU: desenvolvimento colaborativo de plantas e técnicas arquitetônicas para levar esperança e liberdade a hackers, a prisioneiros de tiranos e a vítimas de leis injustas.

Para que ninguém mais se perdesse nos labirintos dos tiranos, desenvolveu um receptor de sinais de satélites que assinalava a localização e o melhor trajeto para chegar à liberdade. Chamou-o Global Positioning Liberator do GNU, ou simplesmente GNU GPL.

mente GNU GPL.

Criou, dentro do labirinto, bem longe da Vista das janelas de Minow\$, uma organização para defesa da liberdade humana: difusão dos ideais do SoL e apoio ao projeto tecnológico GNU. Ensinava o caminho aos prisioneiros que chegavam ao labirinto sem perspectiva de liberdade, convidando-os a participar do projeto e do movimento.

“ Para que ninguém mais se perdesse nos labirintos dos tiranos, desenvolveu um receptor de sinais de satélites que assinalava a localização e o melhor trajeto para chegar à liberdade. Chamou-o Global Positioning Liberator do GNU, ou simplesmente GNU GPL. ”

Alexandre Oliva

Teseu, insatisfeito com a incompatibilidade do Minix (privativo bem ao gosto de Minow\$) com suas preferências arquitetônicas, desenvolvia um componente central que, por acaso, faltava ao GNU. Chegou ao labirinto durante a reunião do movimento em que Dédalo apresentava a segunda versão do GPL. Adotou técnicas, projetos e até a GPL do GNU, mas, ao invés de desafiar a autoridade de Minow\$ e dar crédito ao projeto de libertação, decidiu ceder às pressões, chamando seu projeto Minux e anunciando que vencera o Minotauro, criatura que simbolizava o fruto do trabalho de Dédalo: metade humano, o movimento humanitário SoL, e metade bovídeo, o projeto tecnológico GNU.

Com isso, a participação de Dédalo e Ícaro no movimento chegou ao conhecimento de Minow\$, que por isso os fez prisioneiros. (Por que os aprisionaria por ajudar Teseu a vencer o Minotauro?) Distorcia a realidade para desmerecer o Minotauro e afugentar potenciais colaboradores do movimento e do projeto. Felizmente, por mais que comparasse o GPL a um câncer, a um vírus, não pôde conter o movimento.

Dédalo e Ícaro certamente não conseguiram nem precisaram sair do labirinto voando, mas ganharam asas como todos os usuários do projeto GNU, e puderam sim alcançar o SoL, o Software Livre. O movimento e o projeto voaram longe: chegaram a muita gente, em terras distantes, a filosofia e o conhecimento do Minotauro, de liberdade, de respeito ao próximo, de solidariedade social e de fraternidade.

Já passou da hora de deixar a luz e o calor do SoL derreterem o manto do medo, da incerteza e da dúvida lançados pelos muitos seguidores do tirano Minow\$, com os quais pretendem que aceitemos seu subjugo; que demonizemos ao invés de almejarmos ser hackers livres e autônomos. Querem nos manter gado humano, analfabeto, incapaz de ler ou escrever as linguagens que controlam e pervadem nossa existência para que, como meros usuários, possamos ser mais facilmente programados e usados. Precisamos, devemos e podemos resistir! Não há o que temer: os riscos estão na lenda, nos mitos. Vamos reabilitar, reconhecer e celebrar o Minotauro: o movimento social humanitário do Software Livre e o projeto tecnológico do bovídeo GNU! O Minotauro não morreu, viva o Minotauro!

Copyright 2010 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/svnwiki/blogs/ixo/pub/minotauro> 



**ALEXANDRE OLIVA** é conselheiro da Fundação Software Livre América Latina, mantenedor do Linux-libre, evangelizador do Movimento Software Livre e engenheiro de compiladores na Red Hat Brasil. Graduado na Unicamp em Engenharia de Computação e Mestrado em Ciências da Computação.



**IV** Encontro de Software  
Livre da Paraíba  
**Inscreva-se**



Por Carlisson Galdino

## Episódio 07

### Concorrência

Manhã na cidade de Stringtown. Transeuntes indo e vindo para seus trabalhos e para movimentarem o comércio desta grande metrópole baiana. No pólo de tecnologia as coisas acontecem do mesmo jeito. Mais ou menos, já que boa parte dos programadores tem contrato por horas trabalhadas e chegam quando bem querem...

Milihash é uma das mais fortes empresas mundiais no ramo de placas-mãe. Uma das maiores empresas do pólo tecnológico, mesmo a fábrica propriamente dita ficando fora da cidade. No pólo há apenas a empresa Milihash, que inclui laboratórios de pesquisa, escritórios, departamento de marketing, etc, etc.

Os primeiros funcionários chegam e se deparam com estranhos visitantes...

**Bull:** É isso aí, véi! Perderam! Nós passamos vocês! Agora vocês já eram!

**Funcionário-1:** Quem são vocês? O que são vocês?!

**Tungstênio:** Não interessa, mas pensando bem... Somos a equipe W4! A evolução da principal concorrente de vocês! Bwahahahahaha!!!

**Funcionário-2:** Vocês são da ASUS?

**Tungstênio:** Não.

**Funcionário-1:** Da Gigabyte?

**Montanha:** Também não.

**Bull:** Daqui mesmo de Stringtown!

**Funcionário-2:** Ahhh... A Bitmancer Games?

**Bull:** Não, pô! Na moral! A Sysatom Technology! Somos a Sysatom Technology!

**Funcionário-1 e Funcionário-2:** Ah! ... Legal! A gente pode ir trabalhar agora?

**Montanha:** Claro que não!

**Tungstênio:** Este é um ataque à sede de vocês. Quero que vocês dois entrem e tragam todos os notebooks que houver.

**Funcionário-1:** Notebooks? Mas...

Tungstênio soca a parede do prédio e seu braço a atravessa. Os dois funcionários se olham assustados. Um terceiro funcionário se aproxima.

**Funcionário-3:** O que está havendo por aqui?

**Funcionário-1:** Fomos atacados. Temos que pegar todos os notebooks da empresa.

**Funcionário-2:** É.

**Seamonkey:** É pra hoje?!

**Tungstênio:** Isso! Vão logo que temos pressa!

**Seamonkey:** PQP!

Os três correm para dentro da Milihash para cumprirem a missão ordenada. O grupo de invasores apenas espera do lado de fora.

**Seamonkey:** A idéia era levarmos computadores.

**Tungstênio:** Sim, e notebooks são computadores! Com a diferença que tem bateria!

**Seamonkey:** Muito esperto! Quero ver os dedos de vocês acertarem as teclas de um notebook!

**Montanha:** Isso é um mero detalhe! O importante é ter o equipamento. Não se meta no que não entende.

Seamonkey olha com raiva e se afasta um pouco, olhando os arredores.

**Funcionário-4:** Ei, quem são vocês?

**Funcionária-5:** Ai meu Deus!

**Tungstênio:** Você veio de carro, não veio?

**Funcionário-4:** Sim, mas...

**Tungstênio:** Traga o carro até aqui ou ela morre!

Tungstênio puxa a mulher pelo cabelo e a coloca sob os cuidados de Montanha.

Em pouco tempo o carro está ali.

**Funcionário-4:** Pode levar! Mas deixa a gente em paz!

**Tungstênio:** Vamos deixar...

Os três funcionários chegam, trazendo um total de... Quatro notebooks.

**Tungstênio:** Só isso?!

**Funcionário-2:** É! A maioria das pessoas leva os notebooks para casa e trazem no outro dia.

**Tungstênio:** Tudo bem. Botem dentro do carro.

**Funcionário-4:** Vocês não vão entrar no meu carro, vão?

**Montanha:** Por que?

**Funcionário-4:** Porque não cabem! Vão destruir!

**Montanha:** Não é da sua conta.

**Tungstênio:** Na verdade era para levarmos os notebooks, mas como só foram esses, acho que isso já serve.

Tungstênio arranca a porta do carro.

**Tungstênio:** Mas antes... Vamos, rapazes? Seamonkey, cuide do carro.

Os três vão ao prédio e começam a derrubar as paredes.

**Funcionário-3:** Valei-me! Minha Nossa Senhora dos Bad Blocks!

E os blocos e tijolos vão ao chão diante dos... oito? Não, não. Doze! Doze funcionários que chegaram até o momento.

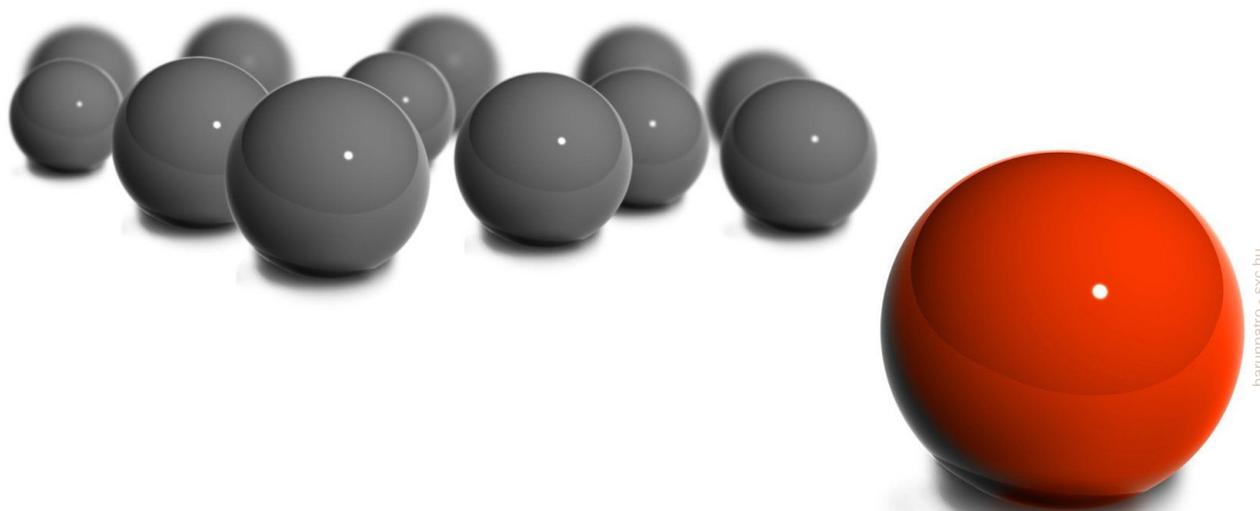
E o grupo W4 vai pela rua levando a porta do carro com os quatro notebooks.

**Bull:** Maneiro isso de derrubar prédio! A gente devia fazer isso mais vezes! Desestressa! 🇧🇷



**CARLISSON GALDINO** é Bacharel em Ciência da Computação e pós-graduado em Produção de Software com Ênfase em Software Livre. Já manteve projetos como laraJS, Enciclopédia 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.

**TreinaLinux**  <sup>®</sup>  
www.treinalinux.com.br



# Gerenciando projetos de Open Source: métodos e regras

Por Cezar Taurion

Um projeto de Open Source não é apenas entusiasmo e trabalho pesado nas horas vagas e fins de semana. O desenvolvimento de um software Open Source não é um trabalho amadorístico, mas demanda métodos de gerenciamento com características e peculiaridades próprias.

Entretanto, ainda não existem, na prática, metodologias Open Source que já tenham sido comprovadas em dezenas de projetos. As iniciativas de maior sucesso como Linux, Apache, Eclipse e outros não usam as mesmas metodologias e nem as mesmas ferramentas de apoio. Entretanto, podemos analisar pontos em comum entre eles e identificar alguns aspectos que

aparentemente os levam a ter sucesso.

Gerenciar um projeto de Open Source é gerenciar pessoas em uma organização virtual, formada por colaboradores voluntários espalhados pelo mundo todo, que trabalham de maneira autônoma e não tem vínculos oficiais entre si, nem com o próprio projeto. Podem sair quando quiserem, sem aviso, simplesmente deixando de colaborar. É um mundo diferente do modelo de gerenciamento típico de uma organização hierárquica, onde as pessoas quase sempre estão no mesmo local e obedecem a regras corporativas e a hierarquias bem definidas.

O grupo de colaborado-

res de um projeto Open Source é diverso em suas competências técnicas (desenvolvedores super-seniores a estudantes), que tomam decisões por si e se engajam por motivos diversos, que vão do ideológico e messiânico ao financeiro, em busca de projeção e novas oportunidades profissionais.

Não existe o contato direto nem reuniões de projeto e, portanto ambigüidades de comunicação (línguas e países diferentes, com seus costumes diferenciados) e muitas vezes a falta de uma estratégia bem definida dos rumos do projeto, além de inexistência de cronogramas e prazos criam toda uma nova dinâmica de gerenciamento de projetos.

Basicamente um projeto de Open Source deve, como qualquer outro projeto, gerenciar os processos necessários a desenvolver as tarefas, bem como gerenciar as próprias tarefas.

Um grande desafio a estes projetos é a construção dos mecanismos de controle e coordenação, uma vez que os membros são autodirigidos e controlam por si mesmo seus ritmos e intensidade de colaboração. Assim, um fator crucial no projeto Open Source é a criação da motivação e confiança entre os membros do projeto. A colaboração voluntária só acontece quando motivada. Conquista-se a confiança dos colaboradores quando o traba-

Um grande desafio a estes projetos é a construção dos mecanismos de controle e coordenação, uma vez que os membros são autodirigidos e controlam por si mesmo seus ritmos e intensidade de colaboração.

Cezar Taurion

lho individual é considerado adequadamente, os conflitos de opinião são julgados e decididos com base em critérios bem definidos e transparentes, e todas as informações sobre o projeto estão disponíveis a todos.

A questão da livre disponibilização das informações é muito importante. Quanto mais compartilhadas forem as informações referentes ao projeto, maiores as chances dos colaboradores contribuírem de forma mais eficaz. Um exemplo é a inspeção de código fonte. Nenhum código deve ter restrições para visualização. Qualquer membro da comunidade pode ter acesso a ele e fazer comentários ou modificações. Se estas forem aceitas ou não é outra ques-

ção, mas o acesso deve ser livre. Um outro exemplo são os processos de tomada de decisão. De maneira geral o mantenedor do projeto e seus auxiliares não devem restringir o acesso às informações e opiniões sobre a aceitação ou não de determinado código, mas sim disponibilizá-las sempre em discussões abertas para a comunidade. É um modelo diferente de projetos convencionais, onde muitas informações são restritas e trafegam apenas entre os gerentes responsáveis, sendo inacessíveis aos demais membros do projeto.

Entretanto, algumas informações podem ser mantidas privadas, como a identificação dos membros da comunidade. Salvo por decisão pessoal, os

# Os projetos de Open Source devem procurar criar políticas de uso de emails (netiquetas) de modo a evitar geração de conflitos desnecessários.

Cezar Taurion

membros identificam-se por e-mails, muitas vezes apenas sob pseudônimos, não cedendo endereços ou telefones para outros contatos.

Os principais meios de comunicação entre os membros do projeto são os eletrônicos, como e-mails, wikis e listas de discussão. São métodos ágeis, mas devido à impossibilidade do contato direto, tornam-se perigosos quando mal utilizados. Em um e-mail não existe feedback imediato, não existe comunicação implícita, de se olhar no rosto, ver gestos e se ouvir a entonação da frase.

Um texto mal escrito ou uma opinião sucinta e direta demais pode gerar conflitos desnecessários. A situação se agrava sob pressão, quando mais e-mails são direcionados e nem sempre respondidos com a calma e clareza necessária. Os líderes do projeto (mantenedor e seus auxiliares) sofrem bastante com esta pres-

são, pois concentram em si um volume imenso de mensagens. Respondê-las de forma inadequada pode criar desavenças inconciliáveis entre os membros, chegando a gerar dissidências no projeto.

Os projetos de Open Source devem procurar criar políticas de uso de e-mails (netiquetas) de modo a evitar geração de conflitos desnecessários. Estas políticas geralmente são informais e aceitas pela comunidade. O site do projeto deve explicitar as regras de conduta que a comunidade deseja que os membros adotem. O site deve também definir as regras de participação e como a organização virtual está estruturada, como e quando acessar os seus líderes, como colocar mensagens, como participar de debates e assim por diante.

O uso de chats também encontra adeptos, embora não permita a participação de toda a comunidade, pelo simples fa-

to das agendas pessoais (e fusos horários) nem sempre coincidirem.

Existe também o desafio da tecnologia de apoio ao projeto. Algumas ferramentas adicionais são importantíssimas, além dos tradicionais e-mails e listas de discussão. Como o projeto é colaborativo e assíncrono, com diversos desenvolvedores trabalhando simultaneamente na mesma solução, é necessário dispor de ferramentas que permitam este tipo de controle.

O gerenciamento das diversas versões (estável, em teste, e instável ou em desenvolvimento) e da versão a ser liberada para uso geral é fundamental em qualquer projeto de Open Source. Não existem regras únicas para esta tarefa e os principais projetos adotam mecanismos diferentes. É necessário definir claramente de quem é a autoridade para liberar a versão para uso e que critérios devem ser obedecidos para que a versão seja liberada publicamente.

De maneira geral, salvo exceções de projetos de grande popularidade, como o Linux, Eclipse ou Apache, os projetos de Open Source são registrados em sites especiais que atuam como espécies de câmaras de compensação (clearinghouses), como o SourceForge ([www.sourceforge.net](http://www.sourceforge.net)), o FreshMeat [www.freshmeat.net](http://www.freshmeat.net) e outros.

O líder do projeto é de fundamental importância. Tem como papéis principais dar a visão do projeto para a comunidade; assegurar-se que o projeto está adequadamente modularizado para ser trabalhado de maneira simultânea; atrair e reter colaboradores; e manter o projeto unido, sem dissidências.

Os líderes de projetos de Open Source conquistam este posto ou por terem iniciado o projeto (geralmente assumem a função de mantenedor) ou por méritos próprios, reconhecidos pelos seus pares. Devem assumir uma postura de coaching, liderando os esforços da comunidade, resolvendo conflitos e tomando decisões sobre os rumos do software. Não existe a figura do chefe ou do patrão, uma vez que não existem vínculos empregatícios entre os membros da comunidade e toda colaboração é voluntária.

Reconhecer os fatores motivacionais e incentivá-los é uma das principais funções dos líderes de projetos de Open Source. Os valores para motivação são diferentes. Por exemplo, entre a maioria do pessoal técnico um fator motivacional é o reconhecimento de sua expertise pelos pares. A meritocracia é o modelo de incentivo. Em áreas de vendas, a motivação principal é geração de dinheiro, com bônus por resultados.

Manter a motivação da comunidade em todo o ciclo de vi-

da do software é questão de sobrevivência do projeto. Desenvolver software em comunidade implica em total colaboração. Entretanto, colaborar não é uma ação incentivada em muitas organizações tradicionais, muito mais focadas nas competições individuais pela conquista de uma nova posição hierárquica ou bônus financeiros.

Nos projetos que envolvam muitos colaboradores, observa-se que uma grande parcela do código vem de apenas um reduzido número de desenvolvedores. Alguns projetos recebem colaboração apenas de estudantes, que desenvolvem código como parte de seus cursos de graduação. Se as colaborações não vierem também de desenvolvedores de nível sênior, a qualidade do código pode ser bastante sofrível. Os desafios gerenciais aumentam sensivelmente, pois os líderes do projeto devem ter muito mais cuidado (e trabalho) na aceitação do código.

Na prática, controlar a qualidade do software é uma tarefa pesada, pois como a comunidade é diversa em sua expertise e proficiência técnica, a variedade de estilos de codificação que chega aos líderes de projetos populares é muito grande, com muita contribuição de qualidade duvidosa.

Este é um outro grande desafio do gestor do projeto: o que aceitar ou recusar das cola-

borações. Muitas colaborações não são aceitas, sejam por existirem outras mais eficientes ou simplesmente por não atenderem aos critérios de codificação do projeto. Os líderes devem ter cuidado na comunicação destas recusas e as decisões devem ser sempre baseadas no voto majoritário da comunidade. Decisões isoladas não fazem parte da cultura colaborativa e são adotadas por poucos projetos.

O modelo de desenvolvimento colaborativo do Open Source já passou por alguns testes práticos. Existem projetos de grande complexidade, com milhões de linhas de código, desenvolvidos com qualidade e rapidez. O ritmo de desenvolvimento incremental de projetos como o kernel do Linux tem se mostrado superior a de muitos projetos de softwares houses.

Existem projetos de alta qualidade, devido ao uso de peer review, com custos diluídos por toda a comunidade. Claro que o que deu certo em Linux, no Eclipse e no Apache não necessariamente dará certo em qualquer outro projeto de software. As circunstâncias peculiares que envolvem estes projetos (timing, motivação, área de domínio, apoio da indústria e outras) não se replicam automaticamente.

Para o sucesso de uma iniciativa de Open Source nos atrevemos a fazer algumas recomendações:

A) O projeto pode começar totalmente do zero ou reaproveitar código já existente em algum repositório como Sourceforge e outros. Começar do zero é sempre mais complicado, pois neste caso não existe nenhuma peça de código que funcione, mesmo precariamente. É sempre melhor começar com algum código, mesmo defeituoso, uma vez que este pode ser depurado e tornar-se rapidamente um software funcional. Mas, se começar do zero, defina claramente o propósito do software, para que as colaborações sejam direcionadas no mesmo sentido.

B) É importante que exista colaboração para a codificação do software. Não recomendamos usar linguagens pouco conhecidas, mesmo que sejam também livres. Sem massa crítica de colaboradores escrevendo código o software simplesmente não sai do lugar.

C) Utilize adequadamente recursos tecnológicos que permitam facilidade de comunicação e controle como mailing lists, wikis, listas de discussão, softwares de controle de versões e assim por diante. O con-

trole de versões, dos códigos fonte e o rastreamento dos patches a serem aplicados são atividades essenciais ao sucesso do projeto.

D) Defina claramente as posições chave da comunidade. Existirão colaboradores pouco ativos e alguns muito ativos. Não esqueça que são voluntários, e talvez não tenham muita flexibilidade para alocar tempo.

E) Defina claramente as regras do jogo, os processos de tomada de decisão e resolução de conflitos, e de aceitação das colaborações. As regras de participação devem ser bastante claras e estarem disponíveis a todos os interessados.

F) Defina as necessidades de documentação e procedimentos de teste. Documentação é importante para usuários finais, que não irão recorrer a códigos fonte para tirar dúvidas de como usar o software. Quanto aos testes, pode-se adotar o clássico mecanismo de peer review ou outro procedimento qualquer. Peer review pode ser adotado se houver massa crítica de colaboradores suficiente para

garantir a qualidade das inspeções de código.

G) Definir política de distribuição de versões. Pode-se usar critérios de três níveis, com a versão estável disponível para uso público; uma versão em teste, aguardando a liberação em um próximo release; e a versão em desenvolvimento, instável e liberada apenas para desenvolvedores.

H) E não esqueça que um Open Source é um projeto que não tem fim. Sempre cabe mais uma modificação ou adição de funções. 🇧🇷

## Para mais informações:

Site Open Source Initiative

<http://www.opensource.org>

Artigo sobre Open Source na Wikipédia

[http://pt.wikipedia.org/wiki/Open\\_source](http://pt.wikipedia.org/wiki/Open_source)



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



# O mercado dos gadgets femininos

Por Krix Apolinário



Mario Alberto Magallanes Trejo - sxc.hu

O interesse feminino por gadgets veio sem mesmo existir algo direcionado a elas, isso se deu pela praticidade trazida por essas belezinhas que fez com que a mulher moderna trocasse sua agenda comprada todo início de ano por smartphones ou até mesmo por laptops onde o gerenciamento de suas atividades se tornariam muito melhores e fazendo com que suas wish lists crescessem ganhando itens além dos já conhecidos por todos como sapatos, roupas, cosméticos e jóias.

De acordo com o Conselho de Assessoria Econômica dos EUA, as mulheres compram cerca de 57% dos eletrônicos voltados ao consumidor e claro que as empresas que os desenvolvem estão de olho mercado e percebendo esse crescimento das vendas para o público feminino começaram personalizar câmeras digitais, celulares, laptops, video gamers e outros de seus produtos, boa parte com na cor rosa, com desenho de flores e tudo de mais meigo possível para atrair ainda mais consumidoras e manter as que já existem.

Vendo esse crescimento no mercado algumas grifes e empresas do mundo da moda começaram a firmar parcerias e podemos citar dentre muitas, a recente parceria da Sanrio [<http://www.sanrio.com/>] com a Sony Ericsson [<http://www.sonyericsson.com/>] no desenvolvimento do celular W395 e também a da Hewlett-Packard [<http://www.hp.com/>] com estilista chinesa Vivienne Tam [<http://www.vivienne-tam.com/>] no desenvolvimento do design do laptop HP Vivienne Tam Edition.

Contudo o que a maioria das mulheres busca não é somente aparência e sim funcionalidade se os dois vierem juntos, se tornam perfeitos, claro, afinal foi por suas funções que elas passaram a adquirir eles. Laptops e os smartphones tem sido, pelo que tenho observado, o alvo de boa parte das mulheres, e tem sido usados bastante, principalmente para uso de redes sociais.

O bom disso tudo é que o mercado está criando gadgets com funções específicas para as mulheres e assim encontramos os mais diversos e com ótimas funções. Fazendo uma busca no tio Google podemos encontrar um monte com as mais variadas funções, e selecionei alguns que achei bem interessantes, mas principalmente úteis, foram:

Fone de ouvido Bluetooth O.R.B. (Orbital Ring Band) desenvolvido pela Hybra Advan-

...o que a maioria das mulheres busca não é somente aparência e sim funcionalidade, e se os dois vierem juntos, se tornam perfeitos, claro, afinal foi por suas funções que elas passaram a adquirir eles.

Krix Apolinário

ce Technology Inc. [<http://www.hybratech.com/>], o interessante dele é que ele não é um fone de ouvido 'tradicional', basta um toque e ele se torna um anel, ou seja, ele é prático, discreto, funcional e lindo.

DuoFertility desenvolvido pela Cambridge Temperature Concepts [<http://www.temperatureconcepts.com/>], para a mulher que está tentando engravidar esse é um gadget perfeito, com ele a mulher pode saber o momento da ovulação com até 6 dias de antecedência e para isso ele mede a temperatura corporal.

Air Touch Foundation criado pela SK-II [<http://www.sk-ii.com/>], quase toda mulher usa maquiagem e realizar requer tempo e muitas não dispõem dele, então nada melhor que um 'aparelho' que possa

aplicar a base de forma uniforme e na quantidade certa com rapidez, e é isso que esse gadget promete.

O mais importante disso tudo é que o mercado tem percebido que as mulheres, tanto quando os homens, tem interesse no mundo da tecnologia, e não é somente para se manter atual e sim pelas facilidade que trazem para as atividades rotineiras, o que ajuda e muito na correria e a multiplicar o tempo que a cada dia que passa se torna mais excasso. 



**KRIX APOLINÁRIO** é graduada em Internet e Redes de Computadores e atua como Administradora de Sistemas Unix/Linux do C.E.S.A.R.

# Velinhas

Por Roberto Salomon



Michael Lorenzo - sxc.hu

A impressão que tenho é que o tempo passa cada vez mais rápido. Ou isso ou comecei a acelerar ladeira-a-baixo. (Não, pensando bem, estes fios brancos na barba são de stress mesmo). O que sei é que nem parece que há apenas pouco mais de um ano, o

João Fernando me mandou um e-mail convidando-me para escrever alguma coisa para uma nova revista eletrônica focada em Software Livre. Acho que nem ele esperava o sucesso e o trabalho que a empreitada gerou.



Não tem jeito.

Depois de um ano e olhar pra trás e ver o que fizemos dá uma ponta de saudade. Alguns textos melhores que os outros, algumas ideias mais atuais, outras que se perderam no contexto.



Roberto Salomon

De edição em edição, a revista se consolidou e os mais de 5 mil downloads mensais comprovam que não se trata de uma moda passageira, a Espírito Livre veio para ficar.

Escrever para uma revista como essa nos dá a liberdade de opinar sobre assuntos dos quais realmente gostamos. Compartilhar nossa opinião com você é uma atividade prazerosa e, algumas vezes, dolorida. O bacana é que concordando ou não nos argumentos, continuamos compartilhando o mesmo espaço e os mesmos ideais. Se por mais nada, só isso justifica a existência de iniciativas como a Espírito Livre.

Foi um ano de aprendizado e de crescimento. Um ano para chegar a um estilo que

me permitisse conversar com os leitores, tentando trazer alguns assuntos mais ou menos sérios de forma mais leve. Sempre lembrando que há que se tomar cuidado ao escrever para uma revista de tecnologia, o risco de ficar chato é muito grande. Acho que, pelo menos até agora, consegui evitar ficar chato demais.

De jogos a aprendizado passando por documentos e comunidades, foi um ano falando um pouquinho de cada coisa, ajudando a ampliar a discussão e lembrando de coisas do fundo da memória. Cheguei até a pensar em cavar umas covas no jardim e ensinar meus filhos a jogar bola de gude. Infelizmente, minha coluna não concordou muito com o projeto e eles ficaram contentes em sa-

ber que um dia existiu um jogo com bolinhas de vidro que era muito popular entre crianças...

Não tem jeito. Depois de um ano e olhar para trás e ver o que fizemos dá uma ponta de saudade. Alguns textos melhores que os outros, algumas ideias mais atuais, outras que se perderam do contexto. Pensando tudo, foi um primeiro ano muito legal. Digo e repito: um primeiro ano. Tenho certeza que muitos outros virão e ainda vou deixar o João Fernando de cabelos brancos (ou minimamente grisalhos) esperando minha coluna atrasada...

Um brinde aos próximos! 🍷

## Para mais informações:

**Blog do Roberto Salomon:**

<http://rfsalomon.blogspot.com>



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



Photo by Eugene Zelenko, copyright (C) 2008 Licensed under a Creative Commons Attribution-Share Alike 3.0 Unported License



Photo by copyright (C) 2008 Licensed under a Creative Commons Attribution-Share Alike 3.0 United States License

# Entrevista com Richard Stallman e Brian Gough, do Projeto GNU

Por Alexandre Oliva e João Fernando Costa Júnior

Nesta edição de aniversário da Revista Espírito Livre, o bate-papo é com Richard Stallman e Brian Gough, ambos do Projeto GNU. A conversa foi conduzida por Alexandre Oliva, colunista da Revista Espírito Livre e membro da Fundação Software Livre América Latina.

**Revista Espírito Livre: Por que o mundo precisava de um sistema operacional, digamos, gnu [em inglês, soa como novo] nos anos 80?**

Richard M. Stallman: O mundo precisava de um siste-

ma operacional em software livre porque não existia nenhum. Portanto, eu decidi escrever um. Tomei, então, decisões técnicas: decidi seguir o desenho do Unix para torná-lo portátil, fazê-lo compatível com o Unix para que os usuários de Unix pudessem migrar para ele facilmente, e dar suporte a máquinas de 32 bits ou superiores, para evitar o trabalho extra de suportar pequenos espaços de endereçamento.

**REL: Que papéis vocês desempenharam, ou desempenham, no projeto GNU e por quanto tempo?**



Figura 1: Richard Stallman, do Projeto GNU e FSF

RMS: Eu lancei o projeto GNU em 1983 e tenho sido seu líder (Chief GNUisance) desde então. Entretanto, recentemente eu envolvi o Comitê de Aconselhamento do GNU em algumas decisões e espero que ele vá tomando conta das tarefas de liderança do GNU no futuro.

Brian J. Gough: Minha função principal na codificação é como mantenedor da biblioteca numérica do GNU, a GSL, à qual comecei a contribuir como desenvolvedor em 1996.

Mais recentemente, fui organizador de vários "Encontros de Hackers GNU" que temos ti-

do na Europa nos últimos anos e neste ano nos EUA, na Conferência LibrePlanet da FSF. Esses encontros são uma oportunidade para os que contribuem para o GNU possam se encontrar e discutir seu trabalho em pessoa -- assim como olhar para frente e ver como poderemos responder às novas ameaças à liberdade.

No ano passado eu me tornei membro do Comitê de Aconselhamento do GNU que promove teleconferências regulares para ajudar a coordenar alguns dos assuntos do dia-a-dia do projeto GNU, tais como organizar esses encontros, por isso tenho dispendido um bocado de tempo nisso, hoje em dia.

**REL: A GNU GPL é considerada como uma das contribuições mais importantes do projeto GNU, mas GNU engloba muito mais: um sistema operacional completamente Livre. O que o GNU anda aprontando hoje em dia, e do que os desenvolvedores**

**do projeto GNU podem se orgulhar nos últimos anos?**

BJG: Tecnicamente, o projeto GNU é focado hoje na melhoria do software existente, como o GCC e o Emacs, para mantê-los atualizados com os novos desenvolvimentos, e naquelas áreas onde precisamos de programas totalmente novos -- os "Projetos de Alta Prioridade" (<http://www.fsf.org/campaigns/priority.html>) como o Gnash, o projeto do GNU para substituição do Flash; GNU PDF, que tem por objetivo ser uma implementação completa do padrão PDF tanto para leitura, quanto para escrita. GNUpstep está atingindo um alto nível de compatibilidade com o framework Openstep (Cocoa) e ajudará as pessoas a escapar da plataforma privativa MacOS.

Também estamos encontrando novas áreas onde o software livre pode ser melhorado -- por exemplo, recentemente o projeto GNU lançou uma nova iniciativa de acessibilidade (<http://www.gnu.org/accessibility/accessibility.html>) para assegurar que pessoas com deficiência tenham a mesma liberdade que outros usuários do software livre. Atualmente, muitos programas livres não são acessíveis, e queremos que todo programa GNU seja acessível e todo desenvolvedor tenha a acessibilidade em mente quando fizer um programa.

Olhando para alguns anos atrás, o trabalho do GNU

Classpath e das equipes do GCJ são exemplos notáveis do valor do trabalho pela liberdade e da substituição de software não livre que as pessoas aceitam por estar disponível sem custo. Quando o Java foi finalmente liberado como software livre pela Sun em 2006, ele estava sob a mesma licença do GNU Classpath e agora temos um ambiente Java completamente livre nos sistemas GNU/Linux (com o nome de IcedTea). O GNU Classpath foi um dos projetos de alta prioridade por esse motivo, por causa do problema de programas livres em Java terem de depender de implementações não livres de Java, às quais RMS chamou a atenção lá em 2004.

Você mencionou a GPLv2 e, claro, houve uma grande atualização da GNU GPL (General Public License, ou Licença Pública Geral) para a GPL versão 3 em 2007, seguindo um processo de consulta pública mundial envolvendo vários projetos de software livre, empresas e especialistas em legislação. A atualização tornou a licença mais fácil de ser usada pelos desenvolvedores e removeu incompatibilidades desnecessárias com outras licenças de software livre semelhantes. Ela também deu aos projetos de software livre novas proteções de patentes de software e tentativas dos fabricantes de contornar as liberdades das licenças utili-

 Tecnicamente, o projeto GNU é focado hoje na melhoria do software existente, como o GCC e o Emacs, para mantê-los atualizados com os novos desenvolvimentos, e naquelas áreas onde precisamos de programas totalmente novos... 

Brian Gough

zando dispositivos bloqueados e DRM (Digital Restrictions Management, ou Gerência Digital de Restrições). Naquela época, algumas pessoas viram pouca necessidade de proteções adicionais, mas agora vemos a nova geração de dispositivos de computação privativos que são completamente bloqueados e usam a DRM para restringir seus usuários. Portanto eu acho que aquelas proteções da GPLv3 provarão ter sido visionárias e agora são mais importantes do que nunca. Eu encorajo todos que desenvolvem software livre a se atualizarem para a GPLv3, caso ainda não o tenham feito.

Olhando para o futuro, nós queremos assegurar que todos tenham liberdade ao usar a Internet. RMS falou recentemente a respeito dos perigos

do "Software como Serviço" na Conferência LibrePlanet da FSF <http://www.gnu.org/philosophy/who-does-that-server-really-serve.html>. Se utilizamos serviços privativos na Web para executar tarefas que poderíamos fazer em nossos próprios computadores com software livre, abrimos mão de nossa liberdade -- isso é algo que devemos evitar.

Para tarefas como processamento de textos, podemos simplesmente utilizar um software livre em nossos computadores. Para tarefas que necessitem de servidores, precisaremos de alternativas. Uma maneira é ter software livre suficiente para que as pessoas possam ter seus próprios servidores em casa ou na sua organização.

“ Recomendamos apenas distribuições que têm uma política firme de não se encaminhar para o software não livre, e explicamos a razão ética para isso. ”

Richard Stallman

Há alguns projetos GNU novos que começam a oferecer tais softwares, como o GNU FM para compartilhamento de música e o GNU Social para redes sociais -- eles estão procurando por mais voluntários. A licença para eles é a GNU AGPL (Licença Pública Geral Affero) -- ela é similar à GPL mas, se alguém roda um programa sob a AGPL num servidor Web e permite que outros o utilizem, eles devem permitir que outros baixem o código fonte. O microblog identi.ca (agora conhecido como Status.net) é outro exemplo de um serviço popular sob a AGPL que você pode baixar e rodar no seu próprio servidor.

Na verdade, o projeto GNU e a FSF têm oferecido serviços livres utilizando a AGPL há um bom tempo através do serviço de hospedagem de projetos Savannah, que está sob a AGPL. Há atualmente mais de 3.200 projetos de software livre utilizando o Savannah e cerca de 50.000 usuários. O

Savannah utiliza apenas software livre e há um link de download em todas as páginas, através do qual qualquer um pode baixar o código fonte completo do site. Também existem oportunidades para voluntários ajudarem no desenvolvimento do Savannah e trabalhar nas solicitações de suporte.

**REL: Apesar disso tudo, várias pessoas sugerem que o projeto não é mais relevante, porque reconhecem o GNU apenas pelas ferramentas de desenvolvimento e pela GNU GPL.**

RMS: Quando as pessoas estão familiarizadas apenas com uma pequena parte daquilo que fazemos e pensam que não é muito, o problema está na ignorância delas, não nas nossas realizações. Não importa quanto bem façamos, não nos ajudará a ganhar apoio se as pessoas o atribuírem a outros.

**REL: Claro, mas deixar essa percepção errada se espalhar prejudica o GNU e o Movimento Software Livre. O que você recomenda para tentar corrigi-la?**

RMS: Além de explicar esse ponto para as pessoas como parte da explicação a respeito do software livre, devemos oferecer nosso tempo, esforço e nomes para atividades que dão crédito ao GNU. Há várias delas que podem utilizar nossa ajuda, portanto não faltarão coisas úteis para fazer.

**REL: O Linux e a maior parte das distribuições GNU/Linux incluem Software não-Livre, o que significa que eles não entendem, ou não compartilham das metas e da filosofia do Software Livre. O que o GNU e a FSF fazem a respeito dessa questão?**

RMS: Recomendamos apenas distribuições que têm uma política firme de não encaminhar o usuário para o software não livre, e explicamos a razão ética para isso.

BJG: No sítio do GNU na Internet existe uma lista de diretrizes que uma distribuição deve seguir para ser chamada de livre e encorajamos as pessoas a utilizar as distribuições que as seguem (as diretrizes estão em <http://www.gnu.org/distros/> assim como uma lista das distri-



Figura 2: Brian Gough  
(buições).

As diretrizes são bastante simples -- o ponto principal é que deve haver uma política clara de somente distribuir software livre, e que ela seja aplicada a todo pacote de maneira consistente, sem exceções para "casos especiais".

Algumas distribuições têm uma política de utilizar apenas software livre, mas não a aplicam totalmente porque ignoram as "bolhas" de software não livre em alguns drivers de dispositivos no kernel Linux. Isso é uma vergonha, porque seria fácil para eles dar a seus usuários uma distribuição 100% livre usando a versão Li-

nux-livre do kernel que não as contém.

**REL: Como alguém se torna um desenvolvedor do GNU?**

BJG: Há muitas maneiras de contribuir para o GNU e ser um desenvolvedor é apenas uma delas. Escrever documentação, contribuir com traduções, testar e auxiliar no Savannah.gnu.org, o sítio de hospedagem do GNU, também são importantes. Existe uma página na web que explica as diferentes formas de se envolver com o projeto em

<http://www.gnu.org/help/help.html>.

Para quem for muito dedicado, existe uma lista dos projetos de alta prioridade em <http://www.fsf.org/campaigns/priority.html>. A lista tem sido reduzida gradualmente -- recentemente, dois brasileiros, Rodrigo Rodrigues da Silva e Felipe Sanches, têm feito bons progressos em um item destacado há muito tempo, uma biblioteca para converter arquivos salvos no formato privativo do AutoCad para formatos livres. Isso é essencial para permitir que as pessoas migrem para softwares CAD livres, devido à

grande disseminação do uso de arquivos AutoCad nessa área.

**REL: Agradecemos a vocês dois pelo tempo e pela dedicação ao GNU e à Liberdade de Software. Algumas palavras finais para nossos leitores?**

BJG: Estamos organizando encontros regionais dos contribuidores do GNU em todo o mundo, e ouvimos falar que Rodrigo e Felipe estarão organizando um na conferência FISL (de 21 a 24 de julho) em Porto Alegre. Portanto, gostaria de incentivar a qualquer um que esteja interessado em contribuir para o GNU a juntar-se a eles neste inverno -- os detalhes do evento serão divulgados na nossa página de eventos em <http://www.gnu.org/ghm/>.

RMS: Milhares de pessoas têm trabalhado para tornar possível aos usuários de hoje utilizarem um computador e ter controle sobre o que ele faz. Em 1983 isso era quase impossível, porque primeiro você teria de escrever um sistema operacional livre. Nós o escrevemos, e atingimos um patamar no qual rejeitar o software não livre e o SaaS é perfeitamente possível com algum esforço. Entretanto, gostaríamos de tornar isso fácil e indolor o tempo todo, e ainda tem chão até chegar lá. Agradeço por nos ajudar a fazê-lo. 🙏



Mike Johnson - sxc.hu

# A LICENÇA "SEJA LEGAL"

Por Cindy Dalfovo

Eu gosto de desenhar – o que não quer dizer que eu SAIBA desenhar, mas que eu gosto de pensar que estou aprendendo. Como tal, um dos meus sites favoritos é o DeviantArt[1], tanto pelos desenhos maravilhosos para me inspirar e eventualmente usar como base, como para encontrar tutoriais e desenhos preto-e-branco nos quais possa treinar colorização.

É comum que os usuários se comuniquem e referenciem o trabalho dos outros, seja mencionando de qual usuário

eles conseguiram aquela base, aquele cenário, aquele pincel...

Ou seja, quem coloca seus trabalhos no DeviantArt geralmente tem um certo espírito de comunidade, de compartilhar coisas legais, e por isso as pessoas costumam permitir que outras pessoas usem seus trabalhos como base para construir coisas diferentes.

Além disso, os usuários podem indicar os melhores trabalhos, que costumam ser expostos com destaque no site.

Temos aí, implicitamente, permissão para visualizar as

imagens, compartilhá-las com quem quisermos, para fazer obras derivadas e colocá-las no site.

Parece familiar?

Sim, é bastante familiar e se chama bom senso – algo que você aprende desde o jardim de infância: não diga que é seu algo que você não fez, mas compartilhe coisas interessantes que você encontrou por aí, use-as como inspiração.

Anos mais tarde, é necessário criar uma licença para dizer às pessoas "Seja legal", "Não roube o trabalho dos outros", "Compartilhe o que é bom".

Ou ainda, foram criadas várias licenças cujo principal intuito é dizer coisas tão simples. Uma das mais famosas é, justamente, a licença GNU GPL (e LGPL, e AGPL), que se estabelece sobre quatro liberdades: o direito de usar, de ler e modificar o código fonte de um programa, e de redistribuir tanto a versão modificada quanto a versão original.

Mas por que criar essas licenças foi tão necessário? E por que ainda existem tantas pessoas que tem medo de usar essas licenças em seus aplicativos? Por que ainda existe tanto software proprietário quando, em muitos casos, uma escolha por um software livre faria muito mais sentido?

Medo. De que roubem seu aplicativo, de que mudem



Por que ainda existe tanto software proprietário quando, em muitos casos, uma escolha por um software livre faria muito mais sentido? Medo. De que roubem seu aplicativo, de que mudem e deturpem o que você fez, de que simplesmente vendam seu programa e você fique a ver navios.

Cindy Dalfovo



e deturpem o que você fez, de que simplesmente vendam seu programa e você fique a ver navios. De que mesmo que você apenas diga "seja legal", alguém irá e não será legal.

E esses receios fazem sentido, sob um certo ponto de vista. Ninguém quer pensar que gastou horas em um projeto para que um "espertinho" venha e roube o seu trabalho. Mas a grande questão é que pessoas anti-éticas não ligam para licenças. Não ligam nem mesmo se o código fonte está disponível – dependendo da complexidade do software, elas irão simplesmente contratar alguém para fazer engenha-

ria reversa e seguir em frente. Pessoas sem escrúpulos sempre irão existir, e você sempre pode ter o azar de ser a vítima da vez de uma dessas pessoas. Licenças não são sobre essas pessoas – para essas pessoas existem leis, fiscalização e processos (e um lugar especial no inferno).

Licenças são para pessoas como eu e você, que gostamos de software, gostamos de compartilhar, gostamos de aumentar a comunidade como um todo. Que conseguimos perceber que nossos programas podem servir para construir coisas ainda maiores, se nós permitirmos isso. Elas ser-

“ ...licenças são um acordo prévio entre desenvolvedor e usuário, e devem expressar o que significa "Seja legal" para o desenvolvedor. Para muitos é algo como "sim, use como quiser, modifique se for necessário...”

Cindy Dalfovo

vem, antes de mais nada, para dizer com o que os desenvolvedores do programa se sentem confortáveis – alguns vão aderir a uma licença GNU, outros irão permitir algo ligeiramente mais restritivo, ou talvez mais aberto como uma licença MIT, outros irão simplesmente lançar-se na licença "Do What The Fuck You Want To"[2].

Novamente, licenças são um acordo prévio entre desenvolvedor e usuário, e devem expressar o que significa "Seja legal" para o desenvolvedor. Para muitos é algo como "sim, use como quiser, modifique se for necessário, pode usar em um produto comercial, mas se for o caso, pense em me pagar um café um dia desses, ok?".

Eu sei o que muitos pensam ao ler textos assim sobre li-

cenças livres: esse discurso é muito bonito, muito tocante, mas não paga o meu aluguel. E há, aí, outro problema com conceitos: nenhuma licença livre diz que o desenvolvedor não deve ganhar nada pelo seu trabalho, muito pelo contrário. Afinal de contas, mesmo o mais idealista dos desenvolvedores precisa comer. O que ocorre é uma diferença na percepção de como isso deve ocorrer, e o que o usuário deve pagar. Um exemplo bastante atual mostra diversas empresas oferecendo serviços online, ainda que disponibilizando seu código fonte.

Um exemplo dos mais famosos é o Wordpress[3], que disponibiliza sua framework para criação de blogs (e de outros tipos de sites) com código

aberto, mas que também disponibiliza um domínio para quem quer apenas criar um blog sem se preocupar com domínios e servidores. E como eles ganham dinheiro? Entre prováveis outras formas, eles conseguem dinheiro com serviços "premium": melhor hospedagem para blogs muito populares, mais espaço em disco, além de mostrar anúncios nos blogs. Isso permitiu que se formasse uma verdadeira comunidade ao redor do Wordpress, com dezenas, centenas de temas, plugins, usuários... ao ponto de que certos temas e plugins foram, eventualmente, incorporados ao núcleo do Wordpress. Ou seja, manter seu código aberto não apenas se mostrou um modelo comercialmente viável e que permitiu que muitas pessoas utilizassem seu aplicativo, mas também permitiu que boas adições fossem feitas ao código original.

Oras, e que desenvolvedor não sonharia com isso? Em ter uma aplicação utilizada por milhares de pessoas e que ainda o sustentasse no final do mês?

Variações desse modelo tem sido muito utilizadas, com diversas aplicações oferecendo versões "Community", gratuitas e de código aberto, e "Enterprise"[4], que é essencialmente a mesma, mas paga e com suporte dos desenvolvedores para treinamentos e eventuais modificações para

“ Licenças livres não são sobre o dinheiro, e não são sobre pessoas sem escrúpulos. Licenças livres são sobre permitir que o maior número possível de pessoas possa usufruir daquele software que você gastou horas para fazer... ”

Cindy Dalfovo

necessidades específicas. Novamente, é uma situação em que tanto desenvolvedor quanto usuário saem ganhando, e muito.

Licenças livres não são sobre o dinheiro, e não são sobre pessoas sem escrúpulos. Licenças livres são sobre permitir que o maior número possível de pessoas possa usufruir daquele software que você gastou horas para fazer, e permitir que outras pessoas possam construir coisas ainda maiores, que você possa olhar e pensar "uau, eu faço parte disso". É, acho que no fundo é isso: software livre é sobre querer fazer parte de algo melhor e maior.

Esta é, provavelmente, a ideia mais forte de qualquer licença livre, inclusive da GPL:

vamos fazer parte de algo melhor e maior. Além disso, a GPL diz: "vamos formar uma comunidade, se você gostar dessa comunidade, participe, senão vá embora", como uma maneira de proteger seu software de deturpações. E, mesmo que eu pessoalmente não pretenda usar a licença GPL em minhas aplicações, eu sou obrigada a respeitar os esforços da Free Software Foundation para construir essa comunidade livre ao redor de fortes ideais.

E, se alguém me perguntar qual a minha licença favorita... seria algo "Do What The Fuck You Want, Just Give Me Credit – And If You Ever Make Money With Some of This, Consider Buying Me Some Coffee". É, acho que isso resume bem. 🙌

### Para mais informações:

[1] <http://www.deviantart.com>

[2] <http://sam.zoy.org/wtfpl/>

[3] <http://www.wordpress.com> e <http://www.wordpress.org>

[4] Como por exemplo o Liferay, em <http://www.liferay.com/>



**CINDY DALFOVO** é estudante de Engenharia de Controle e Automação Industrial, especialista na arte de procrastinar projetos. Atualmente procrastina projetos de jogos, sites, aplicativos e de vez em quando escreve no <http://www.diskchocolate.com/blog> sobre essas coisas. (e-mail: [cindy.dalfovo@gmail.com](mailto:cindy.dalfovo@gmail.com))



# UMA QUESTÃO DE NOME

Por Cárliston Galdino

## O que é GNU?

Tudo começou com projeto de Richard Stallman. Um projeto que visava a criar um Sistema Operacional que fosse totalmente livre, que permitisse a um usuário desenvolver atividades em um computador sem que tivesse que se submeter a licenças nocivas para a sociedade. Licenças que cerceiam, que negam o conhecimento e nos dizem que ajudar nosso vizinho não é uma coisa boa. Para isso surgiu o projeto GNU.

Como o habitat natural dos hackers naquele tempo eram sistemas Unix, o projeto

de Sistema Operacional totalmente livre o adotou como guia. Então a ideia passou a ser a de recriar o Unix. Mas uma vez recriado, não seria mais Unix, seria “um outro Unix”, que seria livre. Então Stallman resolveu fazer uma piada nerd, que já era feita naquele tempo. Criou um chamado “acrônimo recursivo”, que nada mais é do que uma sigla que se repete dentro da própria definição. Alguém cria um software chamado Sure, outro cria um “INÉS Não é Sure”, outro cria um “BUSAM é Um Sure Altamente Melhora-

do”... Esses dois seriam “acrônimos recursivos”. No nosso caso, o Sistema Operacional se tornou o “GNU Não é Unix”.

Mas não é só essa a brincadeira do nome GNU: há mais curiosidades a seu respeito e razões para a escolha desse nome. Uma delas é a pronúncia em inglês, que se aproxima de “New”. Outras razões se referem ao animal Gnu.

Gnu é um quadrúpede estranho que vive na África. Uma mistura de cavalo com mini-búfalo ou qualquer coisa parecida. Enfim, é um animal esquisito, o que tornou a escolha do nome ainda mais interessante. Como se não bastasse tudo isso, Gnus se agrupam em manadas enormes, das maiores do reino animal, o que é bem conveniente para reforçar o conceito de Comunidade, que é tão importante em projetos de Software Livre.

### O GNU e o Linux

O Sistema Operacional GNU estava quase pronto, mas a equipe terminou cometendo um erro ao tentar ser vanguardista demais. O kernel que escolheram desenvolver – o Hurd – seria baseado em microkernel. Imagine várias pequenas partes sendo executadas independentemente, trocando mensagens entre si. A ideia é muito boa, mas o problema estava justamente nessas mensagens. A equipe



Gnu é um quadrúpede estranho que vive na África. Uma mistura de cavalo com mini-búfalo ou qualquer coisa parecida. Enfim, é um animal esquisito, o que tornou a escolha do nome ainda mais interessante.

Cárlisson Galdino



só veio notar quanto era difícil depurar essas mensagens quando já era bem tarde.

Tarde porque apareceu um outro kernel de Sistema Operacional Unix antes que o Hurd ficasse minimamente utilizável: o Linux. Isso terminou sendo muito bom, pois se não fosse o Linux, o GNU provavelmente demoraria ainda uns anos para se tornar utilizável. Mas houve um efeito colateral no fortalecimento do Linux: o nome.

Notem que o Projeto GNU foi idealizado por Richard Stallman, mas contou com uma equipe, tornando-se um projeto fundamental da Free Software Foundation (fundação também criada por Stallman). O nome GNU se refere ao projeto e toda a equipe que

nele trabalhou, resultando em um Sistema Operacional quase completo. Então nos chega um kernel como quem não quer nada e assume todos os méritos. E o que era o Sistema Operacional GNU rodando em kernel Linux passa a ser chamado simplesmente de Linux.

Parece uma disputa de ego, não é? Mas a situação não é tão simples. O Projeto GNU está inteiramente fundamentado nas bases do Movimento Social do Software Livre, na visão de que o software ético é o software livre e que software privativo não é uma boa para ninguém, exceto financeiramente para seus próprios criadores. Do outro lado, o projeto Linux remete à imagem de seu criador e líder: Li-

---

“ GNU é mais que um projeto, é um estado de espírito. Pense bem quando chamar o sistema somente de Linux, tenha cuidado para não estar deixando de lado tudo o que motivou essa importante luta. ”

Cárlisson Galdino

motivou essa importante luta. Falam que GNU/Linux é longo e que Linux é uma forma de resumir, mas não se pode deixar o principal fora do resumo... 

### Para mais informações:

Site Projeto GNU  
<http://www.gnu.org>

Site Free Software Foundation  
<http://www.fsf.org>

nus Torvalds. E o que Linus Torvalds pensa sobre tudo isso? Bem, para Linus, o importante é se divertir.

Resumindo a ópera: um usuário utiliza o sistema e se admira com ele. Então procura saber mais sobre ele. Dizem que o sistema que ele está utilizando é “o Linux”, então ele busca sobre Software Livre e Linux. O que descobre? Nada sobre as causas sociais, e toda a luta da Free Software Foundation termina comprometida por conta de uma palavra...

mo “Licença Pública GNU”. Gente, não! Não existe uma “Licença GNU”! Não podemos nos referir desta forma. Simplesmente porque o projeto GNU oferece várias licenças. O “G” da GPL é de “General” e o GNU você coloca antes, sem abreviar: GNU GPL. Além desta ainda temos a GNU LGPL, a GNU FDL, dentre algumas outras. GNU General Public License quer dizer: Licença Pública Geral do projeto GNU, daí a necessidade de sermos específicos.

### A “Licença GNU”

Além do Sistema Operacional, o Projeto GNU terminou criando muitas outras coisas. Desde o GNOME até... licenças de uso. Já vi recentemente um site que falava da GPL co-

### Conclusão

GNU é mais que um projeto, é um estado de espírito. Pense bem quando chamar o sistema somente de Linux, tenha cuidado para não estar deixando de lado tudo o que



**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, Enciclopédia 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.

# Tradução também é GNU

Por André Gondim

sanja.genero - sxc.hu

Você sabe quais são as quatro liberdades do software livre?

Segundo a Wikipedia, são:

A liberdade para executar o programa, para qualquer propósito (liberdade nº 0);

A liberdade de estudar como o programa funciona, e adaptá-lo para as suas necessidades (liberdade nº 1). Acesso ao código-fonte é um pré-requisito para esta liberdade;

A liberdade de redistribuir, inclusive vender, cópias de modo que você possa ajudar ao seu próximo (liberdade nº 2);

A liberdade de modificar o programa, e liberar estas modificações, de modo que toda a comunidade se beneficie (liberdade nº 3). Acesso ao código-fonte é um pré-requisito para esta liber-

dade;

O que isso significa para um grupo de tradução, como o do Ubuntu ao qual faço parte e lidero? Vou buscar fazer uma explanação geral abaixo.

Primeiro um desenvolvedor faz um programa em sua língua e o libera na forma em que ele desejar, no caso estamos falando em software livre há diversas licenças, dentre as mais usadas estão a GPL V.2 e V3 e, BSD e outras milhares existentes, qualquer pessoas pode criar sua licença, mas aconselho escolher dentre as várias existentes.

Continuando, depois de criado o programa e o mesmo sendo livre, é possível alterar o seu código. Há uma forma de fazer tradução, quando o programador já pensa nisso. A criação de

um arquivo modelo que tem a extensão POT (PO Template). Através desse arquivo é possível gerar arquivos PO, nestes arquivos são possíveis fazer as traduções para todos os idiomas, por exemplo, um grupo da Alemanha quer tradução um programa, baixa o arquivo POT, gera um arquivo PO que terão as frases/palavras (também chamadas de strings) para serem usadas em seu idioma, ao terminar irá enviar para o desenvolvedor que terá a capacidade de gerar o programa na língua desejada. Um exemplo do pacote Brasero que é traduzido pelo time de tradução do GNOME e importado para o Ubuntu, veja abaixo um trecho:

```
#: ../src/brasero-search.c:1044
msgid "Number of results displayed"
msgstr "Número de resultados exibidos"

# Não é continuação de frase alguma
#: ../libbrasero-burn/burn-plugin.c:678
#, c-format
msgid "\"%s\" could not be found in the path"
msgstr "\"%s\" não foi localizado no caminho"

#: ../libbrasero-burn/burn-plugin.c:682
#, c-format
msgid "\"%s\" GStreamer plugin could not be found"
msgstr "Não foi possível localizar o plug-in \"%s\" do Gstreamer"
```

No caso o item original é o que está na linha msgid e o que foi traduzido está na linha msgstr.

A tradução de um programa ajuda na inclusão social e na sua maior adoção por mais usuários em mais países. No momento em que qualquer pessoa de um determinado idioma tem acesso a um programa em seu idioma o mesmo fica mais fácil de uso e de ser mais adotado. No software livre sempre há voluntários dispostos a ajudar na tradução e são sempre bem-vindos. Pois se você deseja ajudar, não é programador, tem conhecimento de inglês básico ou interme-

diário, essa é uma ótima opção.

Vários grupos têm suas portas abertas para voluntários. Alguns deles podem ser vistos em:

GNOME - [http://l10n.gnome.org/teams/pt\\_BR](http://l10n.gnome.org/teams/pt_BR)

KDE - [http://l10n.kde.org/team-infos.php?team-code=pt\\_BR](http://l10n.kde.org/team-infos.php?team-code=pt_BR)

LDP - BR - <http://ldp-br.conectiva.com.br/>

Ubuntu - <http://wiki.ubuntu-br.org/TimeDeTraducao>

Fedora - <http://fedoraproject.org/wiki/L10N/Teams/BrazilianPortuguese>

XFCE - [http://i18n.xfce.org/wiki/team\\_pt\\_br](http://i18n.xfce.org/wiki/team_pt_br)

Debian - <http://debian-br.alioth.debian.org/>

Gentoo - [http://www.gentoo.org/doc/pt\\_br/overview.xml](http://www.gentoo.org/doc/pt_br/overview.xml)

Junte-se ao grupo que melhor você se identificar, conheça, participe, se envolva!

Em breve escreverei mais detalhadamente como colaborar com as traduções no Ubuntu, ao qual lidero o time. 



**ANDRÉ GONDIM** faz parte da comunidade Ubuntu Brasil. Iniciou pela parte de tradução onde hoje é líder desde o FISL 10. Já contribuiu com documentação, suporte (onde vez por outra ainda contribui seja com post, seja na lista de usuários do Ubuntu), Ubuntu Member desde 2007. Eleito membro do Conselho Ubuntu Brasil em Agosto de 2009.



B.S.K. - sxc.hu

# O que ganho ao compartilhar?

Por Juliana Kryszczun

Esta pergunta se encaixa na terceira liberdade do Manifesto do Projeto GNU do Richard Stallman: "A liberdade de melhorar o programa e de tornar as modificações públicas de modo que a comunidade inteira se beneficie da melhoria".

Pode-se afirmar que ao criar ou modificar um código de um software de código aberto e compartilhar para a comunidade se ganha satisfação pessoal. Mas, e aí? Isso realmente basta? Qual a real motivação para compartilhar gratuitamente

te um código de própria autoria e por que as quatro liberdades criadas por Richard conquistaram e conquistam tantos adeptos?

O conhecimento adquirido com o desenvolvimento de software e a visibilidade são pontos bastante impactantes no compartilhamento do código de forma gratuita. A oportunidade de se envolver com grandes projetos de Software Livre mantém o interesse do desenvolvedor em compartilhar seu conhecimento. Para que isso funcione o desenvol-



Pode-se afirmar que ao criar ou modificar um código de um software de código aberto e compartilhar para a comunidade se ganha satisfação pessoal. Mas, e aí? Isso realmente basta?

Juliana Kryszczun



vedor tem que se manter sempre motivado, acreditar no projeto e vê-lo crescer. Isso nem sempre acontece e por este motivo tantos projetos "morrem" no meio do caminho.

Significado da palavra "compartilhar": participar de alguma coisa, partilhar algo com alguém. No Manifesto está escrito que programadores podem se tratar como amigos, compartilhando aquilo que sabem e o que acabaram de descobrir. Não se faz necessário assinar termos de sigilos e tirar dúvidas com um grupo restrito de pessoas.

Há outras formas de compartilhar num projeto de Software Livre além de código. Num projeto grande tem equipes de gerenciamento, análise, teste, etc. Basta ter interesse e envolvimento para compartilhar co-

nhecimento em um projeto de Software Livre.

Isso parece filosófico demais, um "mundo perfeito". Mas se pararmos para pensar, é assim mesmo que o Software Livre é feito. Pessoas contribuem e não recebem financeiramente por isso. E tem inúmeras pessoas que fazem isso! Mesmo no meio do capitalismo que vivemos hoje, as liberdades criadas por Stallman há anos atrás estão a vivas.

Compartilhar conhecimento faz bem, se fosse o contrário não teríamos a quantidade de Softwares Livres com código aberto que temos hoje. Talvez teríamos, mas a qualidade e usabilidade seriam consideravelmente mais baixas. 🐦

## Para saber mais:

Site Projeto GNU

<http://www.gnu.org>

Site FSF

<http://www.fsf.org>

Artigo na Wikipédia sobre Software Livre

[http://pt.wikipedia.org/wiki/Software\\_Livre](http://pt.wikipedia.org/wiki/Software_Livre)



### JULIANA

**KRYSZCZUN** é bacharel em Sistemas de Informação, pós-graduada em Engenharia de Software com Ênfase em Software Livre, certificada em teste pela ISTQB. Atualmente trabalha como analista de teste. Gosta e admira a forma como o software livre é feito. Depois que conheceu GNU/Linux nunca mais largou e incentiva pessoas a experimentarem também. Nas horas vagas aprende Python e escuta música.



# Justiça seja feita! Programa Professor Digital tem nova licitação

Por Luis Henrique Silveira

Apesar do Tribunal de Justiça aceitar a tese de ilegitimidade da ASL, em representar os interesses dos seus associados, a Justiça foi feita pela realidade dos fatos. O Governo do Estado reconheceu o seu erro no encaminhamento na licitação do Programa Professor Digital e, nesta última semana, a Secretaria de Educação (SEC) cancelou o edital de concorrência e lançou outro, atendendo a solicitação das entidades e da comunidade de software livre.

Por este novo edital a abertura das propostas das empresas será no próximo dia 28/4. Aos professores serão oferecidos equipamentos com o sistema operacional Windows que são softwares proprietário

(Código fechado e pago), ou com Linux que são softwares livres (Código aberto e gratuito).

Todo o processo feito anteriormente não tem mais validade. Os professores que se inscreveram para adquirir os equipamentos terão que refazer o pedido. Quem não tinha se candidatado terá uma nova chance. A vantagem é que agora a inscrição ocorrerá diretamente nas agências do Bannisul, onde os contratos serão assinados.

No início do processo a Associação Software Livre e a ONG BrOffice.org entraram com pedido administrativo, para suspensão da licitação e mudança do edital, que foi negado. Não obtendo êxito pela via administrativa entraram

com um mandado de segurança, em dezembro de 2009, contra o edital do pregão eletrônico 589, da Central de Compras do Governo do Estado/Cecom.

Os argumentos eram a possível violação da lei das licitações 8666/93, o direcionamento da compra para o sistema operacional da Microsoft e a venda casada com o equipamento. O edital também não atendia a Lei aprovada na Assembléia Legislativa do Estado que justamente determinava a venda em separado do software e do hardware e a não especificação de uma marca, no caso a Microsoft.

O Tribunal de Justiça do RS acolheu o mandado de segurança e em caráter liminar suspendeu todos os procedimentos referentes ao Programa, em dezembro de 2009. Se o Governo tivesse atendido a solicitação feita pela ASL e BrOffice.org, nesta época, os professores já estariam com os seus tão sonhados notebooks. Agora terão que esperar no mínimo mais 90 dias.

O Programa do Governo que serviria para limpar a imagem da Governadora junto ao magistério virou mais uma dor de cabeça. O atrativo do programa eram as boas condições de pagamento, em 36x com uma parcela compatível com os parques salários, pagos aos professores. Os preços não são tão módicos assim, pois se encontra computador, com a

“ No início do processo a Associação Software Livre e a ONG BrOffice.org entraram com pedido administrativo, para suspensão da licitação e mudança do edital, que foi negado...”

Luis Henrique Silveira

mesma configuração, até mais barato.

Mais de 100 mil professores acessaram a página do programa e mais de 30 mil se inscreveram. Destes, 16 mil já estavam habilitados para receberem os notebooks no início de março. A suspensão do Programa gerou muita polêmica e reclamação deste público que estava ansioso para adquirir o tão sonhado equipamento.

O debate nem entrou no aspecto pedagógico, ele saiu do âmbito jurídico legal, para o emocional, o próprio Governo tentou utilizar este argumento junto ao Supremo Tribunal de Justiça de Brasília, quando entrou com um recurso, que foi negado. Com a suspensão do Programa os professores perderam a compostura e o assunto

virou polêmica na mídia e na internet. Parecia que tinham tirado o doce de uma criança.

Muitos veículos de comunicação do interior do Estado, rádios e jornais ao noticiarem, percebiam a grande repercussão. Jornalistas comentavam que nunca tiveram tanto “feedback” de pessoas indignadas, com a suspensão. A forma como a mídia cobriu o episódio cabe um capítulo a parte. As manchetes colocavam os professores contra a ASL e a ONG BrOffice.org por terem “barrado a aquisição dos notebooks”. O jornalista do site Baguete fez um artigo de opinião dizendo que a ASL “deu um tiro no pé”, porque agora os professores não queriam nunca mais ouvir falar de software li-

“ Na internet então, blogs e sites deram ampla repercussão. Análises profundas, técnicas e filosóficas com relação ao uso do software livre ou proprietário.”

Luis Henrique Silveira

vre. Na sexta-feira (16/4) noticiou que o “Governo venceu a briga do Professor Digital”.

Na internet então, blogs e sites deram ampla repercussão. Análise profundas, técnicas e filosóficas com relação ao uso de software livre ou proprietário. Por outro lado depoimentos e recados chingavam a ASL de “onguizinha” para baixo, endeusando o Windows e a Microsoft e criticando o Linux. Cada post em sites ou blogs gerava dezenas de comentários e teses.

O capítulo desta última semana foi decisivo, com a suspensão da liminar, na sexta-feira (16/4), e a publicação do edital refazendo a licitação. Justiça seja feita, o

governo do Estado refez o edital conforme a lei, a ASL e ONG BrOffice.org garantiram o princípio da moralidade do serviço público e os interesses dos seus associados, no propósito de contribuir para que a sociedade não seja dependente tecnológica de empresas multinacionais e utilizem alternativas livres na área de software. E, o principal, os professores terão seus sonhados notebooks e poderão optar pelo sistema operacional que melhor lhe convier.

A decisão dos desembargadores que extinguiram o mandado de segurança por entenderem que Associação Software Livre não tinha legitimidade para pedir a suspen-

são da compra dos notebooks acabou não surtindo efeito. Porém com este episódio talvez o governo do Estado tenha aprendido alguma coisa sobre leis de licitação e os Professores sobre hardwares e softwares, o que é certo ou errado, o que é legal e moral e o que é ilegal. Outro aspecto importante foi perceber o nível que está a educação do Estado, a partir da reação, postura e falta de educação e bom senso de alguns professores.



### Para mais informações:

**Site Associação Software Livre**  
<http://www.asl.org.br>

**Site BrOffice.org**  
<http://www.broffice.org>



**LUIS HENRIQUE SILVEIRA** é Jornalista e Mestre em Comunicação. Assessor de Imprensa da Associação Software Livre.Org (ASL) e Associação dos Servidores da UFRGS E UFCSPA (ASSUFRGS).



REVISTA  
**espírito  
livre**

LIBERDADE E INFORMAÇÃO

<http://www.revista.espiritolivre.org/>



# Pirataria: como enfrentá-la?

Por Walter Aranha Capanema

Olga Gutovska - sxc.hu

Quando se começa a estudar Direito Penal na faculdade, uma das primeiras coisas que o professor nos ensina (ou, ao menos, nos deveria ensinar) é que não são apenas os malfeitores e psicopatas que cometem crimes. As pessoas de bem cometem, diariamente, alguns pequenos delitos, quase que restritos aos crimes contra a honra (injúria, calúnia e difamação).

A pirataria, antes da Internet, era uma tarefa de persistentes usuários, que se conectavam a BBS<sup>1</sup>, e baixavam programas de computador em seus valentes modems de 28.800 bps. Não era uma tarefa fácil: era preciso ter um programa de comunicação, saber configurá-lo e, ainda, ter a paciência de conseguir acessar a linha telefônica da tal BBS. Nesses tempos remotos, basicamente se pirateavam jogos e aplicativos. MP3? Nem existia ainda.

Com o surgimento e a disseminação da Internet e de sistemas operacionais amigáveis, a pirataria se difundiu de forma assustadora. Não são apenas os hard-users de outrora que pirateiam. Senhores respeitáveis, professores, advogados, médicos e donas de casa compartilham toda a sorte de material protegido.

E não se pirateiam apenas jogos e aplicativos, agora: são livros, audiobooks, músicas, filmes, revistas e HQs, séries de TV e tudo aquilo que puder ser digitalizado.

A pirataria se tornou, então, um crime praticado por todos e, tendo em vista a jurisprudência dos nossos Tribunais, a condenação, no Brasil<sup>2</sup>, basicamente se restringe à empresas que copiam ilegalmente sistemas operacionais, software de escritório e outros de elevado valor.

Mas como deter essa força, se até o Presidente da MPAA<sup>3</sup> – entidade americana que defende os interesses dos estúdios de cinema -, afirmou, no excelente documentário “Good Copy, Bad Copy”<sup>4</sup>, que a pirataria jamais poderá ser detida? Como lidar com um comportamento que é socialmente aceito pela maioria das pessoas, afinal, ninguém sofre preconceito por baixar músicas, filmes e programas, aliás, muito pelo contrário?

“ Com o surgimento e a disseminação da Internet e de sistemas operacionais amigáveis, a pirataria se difundiu de forma assustadora. Não são apenas os hard-users de outrora que pirateiam. ”

Walter Capanema

Existe, no Direito Penal, o princípio da adequação social, segundo o qual o legislador, ao criar uma lei criminal, deverá se restringir aos fatos que tenham relevância social, excluindo, segundo a doutrina de Rogério Greco, “as condutas consideradas socialmente adequadas e aceitas pela sociedade”<sup>5</sup>.

Mas, alerta muito bem o autor, o fato de uma conduta ser praticada por todos não é suficiente para que ela deixe de ser crime, para que a sua norma seja revogada e desapareça do mundo jurídico.

Veja o caso do crime de adultério (art. 240 do Código Penal). Esse crime surgiu com o próprio Código, na década de 1940, e nesse tempo, muita gente foi presa. Com as ideologias do amor livre e da liberdade sexual dos anos 60, se admitiu o casamento aberto, e daí, cada vez mais esse crime era praticado, e a lei ainda tinha a sua vigência. Esse artigo só foi revogado em 2005, pela Lei Federal 11.106.

Para que a pirataria seja socialmente aceita, seria necessário revogar os dispositivos legais que tratam da violação de direito autoral do art. 184, CP (obra literária, científica ou artística), bem como o dispositivo que versa sobre os direitos do autor de programa de computador (art. 12 da Lei 9.609/98).

Surge, então, uma questão que sempre é esquecida pelos defensores da pirataria: a lei penal, quando prevê um crime, faz isso para proteger um determinado bem.

Assim, por exemplo, quando o Código Penal proíbe o homicídio (art. 121, CP), está, inequivocamente, protegendo a vida humana. E nos supramencionados art. 184 e 12, o direito autoral. A revogação dessas duas normas, embora se atendes-se a uma conduta socialmente

aceita, estaria deixando desprotegido o direito do autor.

Defendo que a pirataria deve continuar a ser criminalizada sim. Proteger o direito autoral é proteger a inteligência humana e suas manifestações artísticas. Todavia, entendo que esse combate não deve ser feito apenas pela lei penal.

A própria ciência econômica e as técnicas de mercado podem ser utilizadas para coibir a pirataria. Deve-se prover incentivos e facilidades para o pirata-infrator “trocar de lado” e adquirir o produto de forma legal.

Essa assertiva parece inocente e quase impossível de ser realizada. Mas, na prática, já se tem um exemplo fantástico: a App Store<sup>6</sup>, da Apple, com programas e jogos para seus dispositivos móveis e telefones (iPad, iPod Touch e iPhone). O software é oferecido por preços irrisórios, em comparados aos demais, e há grande facilidade em adquiri-los e instalá-los nos gadgets. Além disso, é uma inteligente plataforma de marketing para seus produtos.

Elimina-se o trabalho de procurar, baixar e

“ Defendo que a pirataria deve continuar a ser criminalizada sim. Proteger o direito autoral é proteger a inteligência humana e suas manifestações artísticas.”

Walter Capanema

instalar um programa pirata e, ainda, há a compra legal por um preço justo e com suporte técnico.

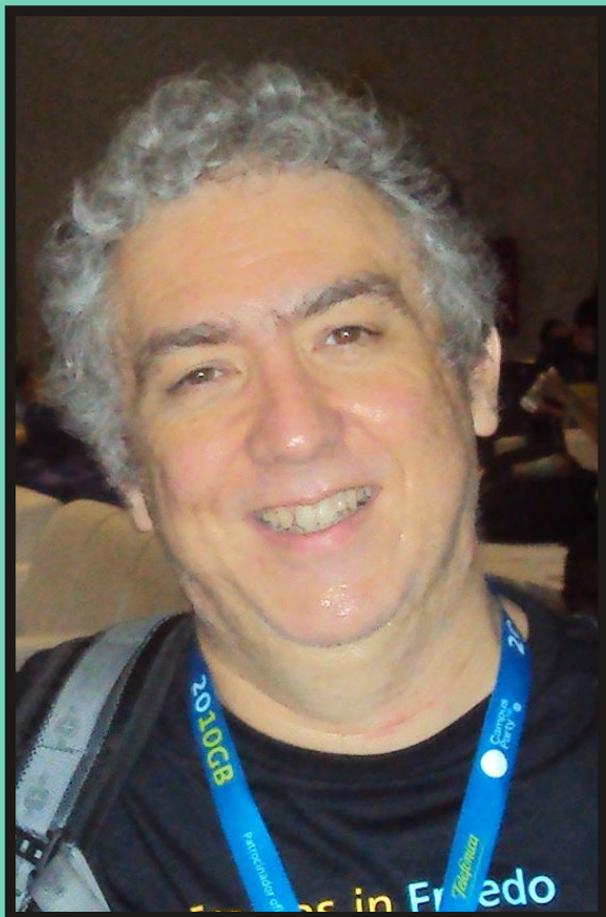
Portanto, a pirataria, por mais corriqueira que seja, deve continuar a ser criminalizada, para proteger o direito do autor. Todavia, o mercado deve buscar, através de preços justos e facilidades, a incentivar o pirata a largar sua vida de crimes. 

### Referências:

1. Acrônimo de Bulletin Board System – eram sistemas de computadores que permitiam a conexão remota, para a troca de mensagens entre usuários e o download e o upload de arquivos. Algumas, mais profissionais, cobravam uma taxa mensal de uso.
2. Nos EUA, já se tem notícia de alguns casos de condenação de pessoas físicas pelo download ilegal. Contudo, os casos são insignificante comparados a quantidade de programas piratas que existem em sites norte-americanos.
3. Motion Picture Association of America – Associação das Empresas de Cinema dos EUA, em uma tradução livre.
4. Esse documentário pode ser legalmente baixado utilizando o protocolo Bittorrent. O endereço fornecido pelo produtor do filme, informado no site oficial (<http://www.goodcopybadcopy.net>), é [http://thepiratebay.org/tor/3700777/Good\\_Copy\\_Bad\\_Copy\\_-\\_XviD](http://thepiratebay.org/tor/3700777/Good_Copy_Bad_Copy_-_XviD).
5. GRECO, Rogério. Curso de Direito Penal: Parte Geral. 5. ed. Niterói: Impetus, 2005.
6. Disponível em <http://www.apple.com/iphone/appstore>. Acesso em 15.04.2010.



**WALTER CAPANEMA** é professor da Escola da Magistratura do Estado do Rio de Janeiro – EMERJ (Brasil). Formado pela Universidade Santa Úrsula - USU. Advogado no Estado do Rio de Janeiro. Email: [waltercapanema@globo.com](mailto:waltercapanema@globo.com)



# Entrevista com o desenvolvedor Luciano Ramalho

Por Juliana Kryszczun

“ Python é uma linguagem de uso geral, útil para fazer desde pequenos scripts até grandes sistemas como ERPs e CMSs.

Luciano Ramalho constrói sistemas para Web desde 1994. Foi fundador da Hiperlógica, onde desenvolveu projetos para UOL, AOL Brasil e Americanas.com, entre outros. Foi co-fundador e primeiro presidente da Associação Python Brasil. É analista de sistemas na BIREME/OPAS/OMS, o Centro Latino-Americano e do Caribe de Informações em Saúde, que opera diversas bibliotecas digitais a partir do campus da Unifesp/Escola Paulista de Medicina. A entrevista a seguir foi realizada durante a Campus Party em São Paulo, em Janeiro de 2010.

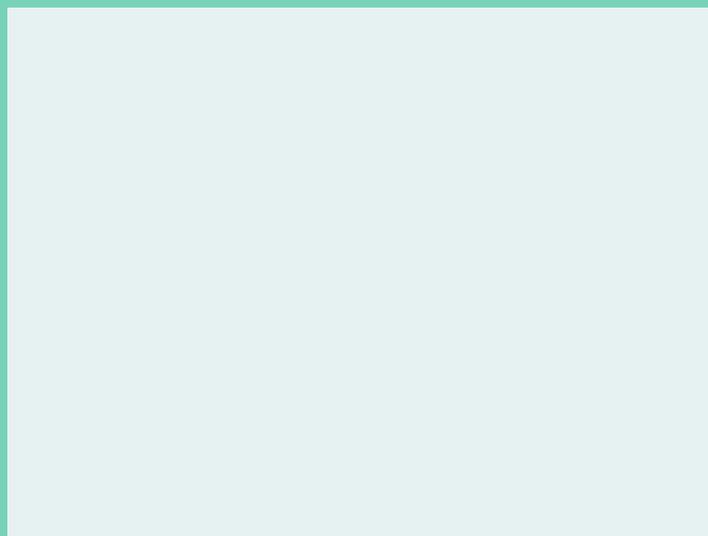




Figura 1 - A nossa correspondente Juliana Kryszczun juntamente com Luciano Ramalho na Campus Party 2010

**Revista Espírito Livre: Nos fale um pouco sobre o Python e a sua importância para o desenvolvimento de software atualmente.**

Luciano Ramalho: Python é uma linguagem de uso geral, útil para fazer desde pequenos scripts até grandes sistemas como ERPs e CMSs. Provavelmente o maior usuário de Python no mundo é o Google, que usa a linguagem para administrar seus milhões de servidores, e também para desenvolver produtos como o YouTube e o GoogleCode. Além disso, Python vem pré-instalada na maioria das distros Linux e no MacOS X, tem versões especiais para as plataformas Java (Jython) e .Net (IronPython) e roda em muitos celulares Nokia, iPhone e Android.

**REL: Python e Software Livre, grandes parceiros?**

LR: Python hoje é parte obrigatória da maioria das distribuições Linux, pois muitas ferramentas essenciais são escritas em Python. Em especial, as distros Ubuntu e Redhat/Fedora utilizam Python como sua principal linguagem de scripting.

**REL: Na sua palestra na Campus Party sobre 10 motivos para usar Python, você abordou sobre a computação gráfica e o Python, fale um pouco sobre o assunto. Aproveitando, nos fale qual foi o papel do Python e da computação gráfica no filme Avatar.**

LR: A produtora de efeitos Industrial Light & Magic, responsável pela série Star Wars, Avatar e muitos outros mega-filmes, utiliza Python há anos, conforme depoimento publicado no Python.org. Mais recentemente, outras produtoras muito importantes como a Weta (Senhor dos Anéis, King Kong, Avatar etc) revelaram que usam a linguagem, e isso

teve um efeito dominó em toda a indústria de computação gráfica. Exemplo disso é que a Autodesk integrou Python em seus produtos Maya e 3ds Max, que já tinham linguagens de script próprias. No ano passado um diretor de computação gráfica me contou que Python virou o padrão de fato na indústria, e que profissionais com conhecimento de Python e 3D têm emprego garantido. Muitos aplicativos gráficos importantes, do Nuke ao Blender, usam Python como linguagem nativa.

**REL: Como você desenvolve em Python desde 1999, como vê o Python daqui uns 10 anos?**

LR: Python tem evoluído de forma muito consistente e sem atropelos desde que eu a conheci. Agora estamos passando pela transição mais importante dos últimos anos, indo para a versão 3. Mas a massa de bibliotecas para a versão 2 é tão extensa que esta versão continuará em uso por muitos anos ainda, então a tendência é que elas evoluam juntas, e aos poucos série 2.x vai ficando cada vez mais parecida com a série 3.x. Isso já pode ser notado na versão 2.6

“ Python hoje é parte obrigatória da maioria das distribuições Linux, pois muitas ferramentas essenciais são escritas em Python. ”

Luciano Ramalho

onde muitas novidades da versão 3 se fazem presentes.

**REL: Qual a importância, no âmbito nacional, a Campus Party, uma feira de tecnologia, para o Python e o Software Livre?**

LR: A Campus Party é essencialmente um evento cultural e de entretenimento, e como tal é muito importante no cenário nacional. Mas sua importância é bem menor que a do FISL, onde além do aspecto cultural e político, existe um conteúdo técnico muito mais amplo e profundo.

**REL: Você já conhecia a Revista Espírito Livre. Como enxerga esta iniciativa de disponibilizar uma revista digital neste segmento?**

LR: Acho muito importante existirem veículos jornalísticos que compreendam e apoiem os

ideais da cultura livre, e por isso a Revista Espírito Livre tem meu total apoio!

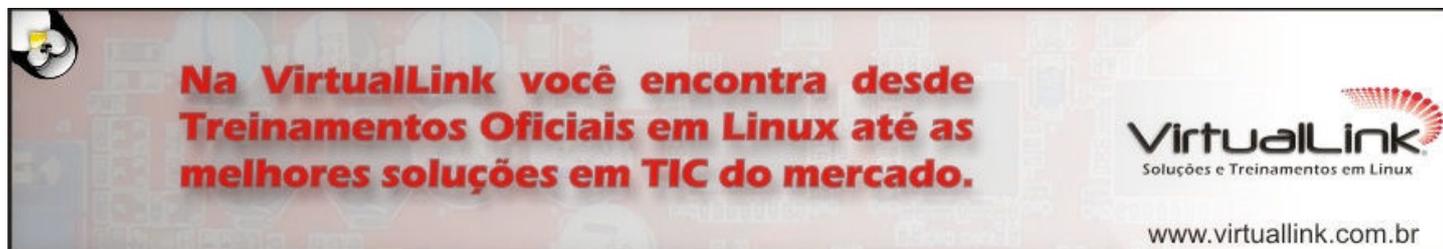
**REL: Deixe um recado 'pythonico' para os leitores.**

LR: Se você programa porque gosta, você precisa conhecer Python, nossos encontros e nossa lista de discussão (veja em <http://python.org.br>). Nossa comunidade tem um nível técnico muito bom. A maioria dos programadores Python não aprendeu a linguagem porque foi obrigado, mas porque gosta de aprender novas linguagens. E além de exercitar seu cérebro com ideias novas, você ainda tem uma boa chance de alavancar sua carreira, porque a demanda por bons programadores Python está muito aquecida, não apenas na computação gráfica, mas também no desenvolvimento Web graças ao sucesso de projetos como o Django e o Google App Engine. 🇧🇷

### Para mais informações:

Site oficial Python:  
<http://www.python.org>

Site Python em Português:  
<http://python.org.br>



**Na VirtualLink você encontra desde Treinamentos Oficiais em Linux até as melhores soluções em TIC do mercado.**

**VirtualLink**  
Soluções e Treinamentos em Linux

[www.virtuallink.com.br](http://www.virtuallink.com.br)



# O FIM DAS SUÍTES DE ESCRITÓRIO

Por Jomar Silva

Particpei nos últimos dias do IV Encontro Nacional do BrOffice.org, e disse uma coisa durante minha apresentação que gerou uma grande polêmica, e para explicar como cheguei a esta conclusão, escrevo este artigo.

Durante a minha apresentação, eu disse que as suítes de escritório, como conhecemos, estão com seus dias contados.

Ao contrário do que muita gente pensa, as suítes de escritório como conhecemos, não foram desenvolvidas com ênfase a aumentar nossa produtividade. Na verdade elas têm passado por diversas melhorias nos últimos anos visando o aumento da produtividade, mas na verdade elas pecam na sua concepção inicial, e para entender isso, precisaremos voltar algumas décadas no tempo.

Antes de falar das suítes de escritório, vale a pena recordar alguns fatos relevantes sobre o próprio computador. Alguém aí sabe me dizer o motivo pelo qual utilizamos (ou utilizávamos) o monitor do computador em cima da CPU? Alguém sabe o motivo pelo qual o monitor está sempre acima do teclado? Falando em teclado, alguém sabe a origem da distribuição de teclas que usamos hoje, o famoso QWERTY?

Se alguém se animou e respondeu: Porque é melhor assim, está completamente engano...

O monitor sempre ficou em cima da CPU, e portanto acima do teclado por um único motivo: Era assim que funcionava a máquina de escrever!

Quanto ao layout do teclado, a máquina de escrever é novamente a resposta. Quem teve oportunidade de usar uma máquina de escrever na vida deve se lembrar do problema que ocorria quando se digitava com muita rapidez: Os tipos se enroscavam.

Já que este era um problema sem solução quando as máquinas foram criadas, a solução dada foi bem simples: Para evitar que as pessoas digitem com rapidez, vamos distribuir as teclas no teclado de forma que as letras mais utilizadas fiquem mais espalhadas possível, tornando a digitação mais lenta!

Sendo assim, você usa um computador hoje que tenta imitar uma máquina de escrever, com pior teclado possível para a digitação... será que é melhor assim?

Quando os computadores pessoais começaram a ser introduzidos nos escritórios, para que eles fossem realmente úteis e desejáveis, foi necessário incorporar ao PC uma série de funcionalidades que permitissem que ele substituísse coisas utilizadas até então.

Foi assim que uma empresa desenvolveu um software para editoração eletrônica (sim, este era o nome dado ao que conhecemos hoje como editor de texto). Inicialmente o software era apenas um editor de texto puro, mas com o tempo este cara foi ganhando cada vez mais funcionalidades até que fi-

nalmente ele eliminava seu concorrente direto, a máquina de escrever.

Todo o modelo de utilização de um editor de texto foi baseado em uma máquina de escrever com funções de formatação (tanto do texto como das fontes), superando assim seu concorrente direto.

Como disse um amigo meu do ODF TC, ao ver a nova barra de formatação de uma suíte de escritório: Será que precisamos mesmo de tudo isso? Estamos escrevendo um texto ou confeitando um bolo?

Quantas pessoas você conhece que antes de iniciar um texto qualquer, perde quase meia hora escolhendo a fonte, formatando o parágrafo, espaçamento, alinhamento e outras

“ “ Todo o modelo de utilização de um editor de texto foi baseado em uma máquina de escrever com funções de formatação (tanto do texto como das fontes), superando assim seu concorrente direto. ” ”

Jomar Silva

“ Em dado momento da história, os usuários de planilhas eletrônicas notaram que as planilhas eletrônicas lhes permitia desenvolver pequenos sistemas, e por isso a importação e exportação de dados passou a ser importante. ”

Jomar Silva

coisas no seu “bolo” de texto? Será que isso é mesmo produtivo? Será que utilizar estilos pré definidos para isso não é mais inteligente? Será que todos os usuários sabem usar de verdade os estilos do seu editor de texto?

Os profissionais que trabalhavam com números, além da máquina de escrever, utilizavam uma folha para cálculos (quase um papel quadriculado) e suas potentes calculadoras no seu dia a dia. Foi então que uma desenvolveu uma versão automatizada da folha de cálculo, com uma grande vantagem: Ela poderia fazer os cálculos automaticamente! Nascia ali a planilha eletrônica.

Em dado momento da história, os usuários de planilhas eletrônicas notaram que as planilhas eletrônicas lhes permitia

desenvolver pequenos sistemas, e por isso a importação e exportação de dados passou a ser importante. A forma mais inteligente e simples de utilizar dados de planilhas para cálculos mais elaborados por sistemas especializados era através do acesso direto ao arquivo, mas para isso era necessário que o fabricante explicasse como o arquivo estava gravado, informando o seu formato... mas se o fabricante fizesse isso, seu concorrente poderia desenvolver uma aplicação que lesse e manipulasse os arquivos gerados pela aplicação dele, tornando esta aplicação dispensável para seus clientes, e isso era algo inadmissível!

Foi aí que os fabricantes deram uma solução genial: criaram as macros!

O resto da história todos nós conhecemos: usuários passaram a desenvolver micro sistemas dentro da sua “planilha eletrônica”, se tornando reféns do software. Pior do que este aprisionamento é o fato de que usuários normalmente não possuem conhecimentos de programação e por isso acabam fazendo verdadeiras peripécias computacionais para desenvolver seus próprios programas. Além disso, acabam cometendo um erro de segurança extremamente grave ao armazenar dados e inteligência de negócio em um único arquivo: basta o roubo de um arquivo esse para que o concorrente descubra os preços praticados pela empresa, seus custos e sua metodologia de cálculo e formação de preço, que normalmente diz muito sobre as parcerias que a empresa possui, seus fornecedores e contratos de fornecimento.

Já que a máquina de escrever e a folha de cálculo estavam digitalizadas, uma terceira empresa resolveu então criar uma versão digital do bom e velho flip chart. Nascia o software para apresentação, que inicialmente era utilizado para se produzir material impresso em papel, que depois foi substituído pela transparência (alguém aí lembra dos retro projetores), que por sua vez foi substituída pelo data show (que era uma transparência digital, como uma tela LCD que

podia ser colocada no retro projetor) para finalmente chegarmos aos projetores modernos e suas conexões diretas com os computadores.

A evolução destas ferramentas de apresentação acabaram incorporando a ela diversas funções e capacidades, e não é raro hoje em dia encontrarmos apresentações que contém de trilha sonora a vídeos embarcados, coisas que o velho flip chart jamais poderia fazer.

Alguns anos mais tarde, empresas se fundiram e foram compradas, até que uma delas resolveu colocar os três produtos em uma única caixa e nasceu assim a suíte de escritórios, que em tese “integra” os produtos. Na prática, esta integração nunca foi de fato realizada, e apesar de ser um argumento convincente de vendas, tecnicamente é muito precária.

As suítes de escritório passaram então a ser tão importantes nos computadores quanto o sistema operacional e rapidamente todos já não conseguiram viver mais sem uma suíte de escritório.

Passaram-se décadas, e hoje em dia continuamos utilizando praticamente as mesmas ferramentas de 20 ou 30 anos atrás, mas será que nossas necessidades hoje são as mesmas daquela época? Será que as suítes de escritório são mesmo produtivas hoje em di-



As suítes de escritório foram criadas em uma época onde a atividade de elaboração de documentos de escritório era individual. A realidade hoje em muitas empresas é que diversas pessoas contribuam para um mesmo documento...



Jomar Silva

as? Eu acho que não.

As suítes de escritório foram criadas em uma época onde a atividade de elaboração de documentos de escritório era individual. A realidade hoje em muitas empresas é que diversas pessoas contribuam para um mesmo documento, e que este seja revisado por um outro grupo de pessoas. O resultado disso são as intermináveis versões de documentos, anotações com fonte colorida e utilização de diversos outros recursos que na verdade só estão lá para que não tenhamos uma boa desculpa para não utilizar mais as suítes de escritório para trabalhos colaborativos. Quem já trabalhou com um wiki sabe bem do que estou falando, e entende

ainda a dinâmica diferenciada que os wikis possuem, muito mais alinhados às nossas necessidades atuais. Quem já usou uma suíte de escritórios on-line (como o Google Docs) sabe também os benefícios que elas nos propiciam.

Temos ainda um caso de uso muito comum hoje em dia, onde uma planilha contém os dados que serão utilizados para a elaboração de um relatório de texto, que por sua vez será a base de uma apresentação. Será que é mesmo útil e produtivo ficar trabalhando a mesma informação várias vezes, sempre na tríade planilha – relatório – apresentação? Será que temos mesmo tanto tempo assim pra perder com isso? Eu acredito que não.

“ Mais legal do que ficar imaginando como a COISA funcionará, é saber que a COISA pode nascer aqui no Brasil, uma vez que hoje estamos em pé de igualdade com os demais países para o desenvolvimento de software...”

Jomar Silva



Nos dias de hoje, já temos capacidade computacional para desenvolver um sistema que armazene as informações e que as apresente de acordo com a necessidade ou vontade do freguês. Não sei como chamar este tipo de software e por isso tenho me referido a ele carinhosamente como COISA.

A COISA vai funcionar assim: coloco nela as minhas informações, crio uma estrutura de relatório (que será também a estrutura da apresentação) e digo para a COISA qual trecho de dados será utilizado em cada parte da estrutura do relatório. Aí a coisa prepara um ambiente para que eu comece a inserir meus dados e elaborar meu texto, fazendo cada coisa apenas uma vez e com a

possibilidade de fazer tudo isso de forma colaborativa. Quando eu terminar de rechear a COISA, posso pedir a ela que me apresente então toda aquela informação como planilha, texto ou apresentação e voilá: ela se encarrega de tratar os dados e me entregar em poucos segundos o documento do jeito que eu preciso. Isso não faz mais sentido?

Quando imaginamos que a COISA vai funcionar na nuvem, as coisas começam a ficar mais interessantes. Já imaginaram ter um desktop corporativo que tenha como pré requisito apenas ter um navegador Internet? Será que este não é o melhor cenário para as empresas no mundo todo? Será que a produtividade não será maior? Será que a

gestão da aplicação não será mais eficiente com ela centralizada?

Mais legal do que ficar imaginando como a COISA funcionará, é saber que a COISA pode nascer aqui no Brasil, uma vez que hoje estamos em pé de igualdade com os demais países para o desenvolvimento de software, e o nivelador indiscutível foi (e é) o Software Livre.

Não preciso nem dizer que ao exportar e importar arquivos a COISA irá usar o ODF, e nem alertar que o padrão ODF como se encontra hoje já está mais do que preparado para que a COISA seja desenvolvida.

Espero que depois de ler este texto todo, você comece a quebrar a cabeça e se motive para desenvolver a COISA, e que entenda o que me faz pensar na morte das suítes de escritório (ainda que não concorde comigo). 🙋



**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).

# PLC: TRANSMISSÃO DE DADOS PELA REDE ELÉTRICA

Por Patrick Amorim



Há muito tempo ficamos pensando quando chegamos a um escritório ou em nossa própria casa, onde poderíamos colocar um computador para ficar mais fácil ou discreto de esconder ou embutir os cabos da rede, se teria que furar a parede ou colocar canaletas, comprar um access point para colocar apenas um computador só para não interferir no layout desejado daquele ambiente, mesmo assim se preocupando com a frequência de micro-ondas ou telefones sem fio.

Agora com a regulamentação da Anatel para as condições do uso da tecnologia PLC, através da resolução 527 que permite a utilização de radiofrequências em banda larga pela rede elétrica, junto com a radiocomunicação que utiliza a radiofrequência de faixa entre 1.705 kHz e 50 MHz, junto com o regulamento de uso, ficou mais fácil a aquisição para se utilizar esta nova

tecnologia em casas e escritórios que pode ser vista em (<http://www.aneel.gov.br/cedoc/ren2009375.pdf>).

A rede elétrica no Brasil está presente em 95% das residências, segundo divulgado através de pesquisas da Companhia Paranaense de Energia – Copel, ela uma das empresas que já estão testando a utilização da transmissão de dados pela rede elétrica, isso significa que não seria preciso modificar nenhum layout ou estrutura já existente naquele ambiente, apenas adequar a tecnologia naquele local desejado, plugando o adaptador na tomada, assim fazendo que todo o sistema elétrico também seja uma rede de dados.

A tecnologia PLC (Power Line Communication) funciona de forma a transformar qualquer tomada elétrica em um ponto de dados, sendo 100% plug and play, para ser ligado, por exemplo, um computador, uma TV com o sinal digital, com uma velocidade muito alta e sem perdas de dados, a taxa pode variar entre 3 Mbps e 33 Mbps, com a média de 20 Mbps, valor muito além das outras tecnologias oferecidas a residências ou escritórios atualmente, a frequência de uma rede elétrica é de 50 e 60 Hz já a PLC é de 1 a 30 Mhz, assim ficando livre de qualquer interferência.

Existem dois tipos de redes PLC, onde são as seguintes:

Indoor é a comunicação de uma rede de baixa tensão, onde se usa a rede elétrica de

Conexão	Velocidade	Download
PSTN	56kbps	3,84KB/s
ADSL	512kbps	53,6KB/s
PLC	7.5mbps	938KB/s

Tabela 1: Comparativo de velocidade



Figura 1: Exemplo de uma rede PLC

dentro daquele ambiente e o transforma em uma “lan”.

O outdoor é o contrário, ela utiliza a rede pública dos postes da rua, podendo ser classificada da seguinte maneira:

- **Segmento de Média Tensão:** É o trecho entre a subestação da operadora de energia elétrica e o transformador que vai até o usuário final.
- **Last mile:** É o trecho entre o transformador "de baixa tensão" do poste, para a residência.
- **Last inch:** São trechos da rede elétrica localizada dentro da residência.

A segurança em uma rede PLC é utilizada uma criptografia que é a DES 56-bits, desenvolvida pela IBM na década de 70, onde se utiliza usuário e senha, ou se utilizando de uma VLAN onde os dados ficam em uma rede separada assim tendo um melhor controle de acesso a rede.

O padrão Homeplug se trata de uma aliança onde estão participando várias empresas com a finalidade de padronizar o acesso a redes do-

miciliares, trabalhando na camada 2 do modelo OSI, que significa que ele trabalha na enlace, podendo ser agregada a uma rede com padrão TCP/IP já existente. No padrão Homeplug 1.0 é usado uma variante bastante conhecida, que é a CSMA/CA (Carrier Sense Multiple Access with Collision Avoidance), com este padrão se pode chegar a taxas de 45 Mbps, onde agora existe também em estudo o padrão HomePlug 2.0 onde nele se pode chegar a taxa de 200 Mbps.

### Exemplos de Testes em uma Rede PLC

Alguns testes já foram feitos em algumas concessionárias de energia elétrica, mais poucas disponibilizam os seus resultados, um exemplo da hidroelétrica de Xanxerê, que disponibilizou dados sobre as velocidades de transferência, conforme pode ser visto na tabela 1. Foi feita a comparação de três links diferentes, onde o primeiro é o de um acesso discado de 56Kbps, o segundo de uma conexão ADSL de 512Kbps e o terceiro de uma rede PLC de 7.5Mbps.

### Equipamentos PLC

#### Modem PLC

O modem tem como funcionalidade de realizar a interface, entre os equipamentos dos usuários e a rede elétrica, ele vai transformar o sinal do equipamento de telecomunicações em um sinal modulado e logo depois é transportado pela rede elétrica, o modem tem a capacidade de separar voz e dados, existem diversos tipos de modems, como para o uso para acesso a internet, para internet e telefonia e apenas para voz.

#### Isolador de Ruídos

Este equipamento deve ser sempre utiliza-

do em uma conexão do modem PLC, quando existirem no mesmo circuito outros equipamentos eletrônicos.

#### Caixa de Distribuição

É utilizada uma caixa de distribuição para a finalidade de se facilitar a distribuição do sinal PLC em painéis, este tipo de aparelho é bastante utilizados em prédios, estes equipamentos vem acoplados com filtros de surto, onde estes equipamentos filtram os ruídos provocados por outros equipamentos na rede elétrica.

#### Repetidores PLC

Um repetidor tem a função de recuperar e re-injetar o sinal PLC nos equipamentos de volta a rede elétrica doméstica, ele é normalmente instalado nas salas de medidores dos prédios, são bastante utilizados como um nó na banda, para aumentar a largura da banda em segmentos críticos.

### Conclusões Finais

Depois de muito ler sobre esse assunto a conclusão que posso tirar desta nova tecnologia, é que ela não veio para apenas ser mais uma possibilidade, e nem muito menos substituir a tecnologia wi-fi, já que agora se pode colocar um access point em qualquer local da casa, basta conectar o modem adsl da velox por exemplo em qualquer tomada com o modem plc, é bastante interessante a interação entre essa tecnologia, colocar um prédio todo conectado a internet apenas ligando o equipamento na caixa de distribuição do prédio e outro no transformador, o mais importante que eu achei dessa tecnologia é que pode ser levada a internet a pessoas menos desfavorecidas, chegando assim a tão espe-

rada inclusão digital em favelas e pequenas cidades, como já existe em testes, em algumas cidades, essa possibilidade de algumas ruas com a tecnologia PLC já está em funcionamento. 🇧🇷



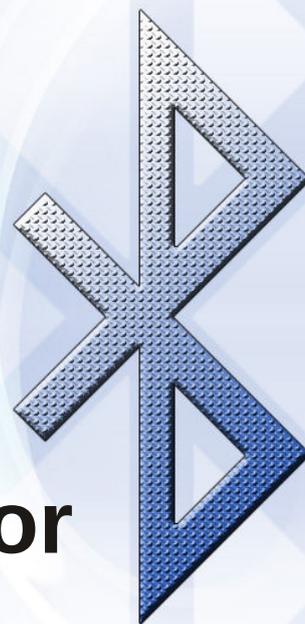
**PATRICK AMORIM** é técnico em Eletrotécnica pelo CEFET-AL, graduando em sistemas de informações pela faculdade de Alagoas, bolsista do instituto de tecnologia em informática e informação do estado de Alagoas, na parte de redes de computadores, entusiasta do Linux e participando do grupo de usuários Linux de Alagoas desde 2007.

### Referências

- Ferreira, Marcos Vinícius. Artigo sobre: PLC – Power Line Communication. Rio de Janeiro. Disponível em: <http://www.radiocb.com/main/images/pdf/PLC.pdf>. Acessado em Novembro de 2008.
- <http://www.wiplug.com.br/empresa.htm>. Acessado em Novembro de 2008.
- <http://www.homeplug.org/home>. Acessado em Janeiro de 2010.
- <http://www.baixaki.com.br/info/3068-power-line-communication-a-rede-na-tomada.htm>. Acessado em Janeiro de 2010.
- <http://www.baixaki.com.br/info/3172-como-anda-o-avanco-do-power-line-communication-no-brasil.htm>. Acessado em Janeiro de 2010.
- [http://www.midiacom.uff.br/~debora/fsmm/trab-2005-2/apres\\_PLC.pdf](http://www.midiacom.uff.br/~debora/fsmm/trab-2005-2/apres_PLC.pdf). Acessado em Janeiro de 2010.
- <http://www.baixaki.com.br/info/1379-banda-larga-na-tomada-.htm>. Acessado em Janeiro de 2010.



# Sensor de Presença no Linux: Ativando / desativando a proteção de tela do seu computador



Por Marcelo Moreira de Mello

Já imaginou que interessante você se aproximar de seu computador e a proteção de tela ser automaticamente desbloqueada? E enquanto você se distancia a proteção de tela ser ativada? É exatamente isso que estamos falando: O seu computador detectar a sua presença! Além de aumentar a segurança de seu computador, é uma boa maneira de impressionar os seus amigos nerds!! :)

Parece mágica entretanto, é claro que precisamos de um mecanismo para enviar algum sinal para o computador indicando que você já esta por perto. Para implementarmos o nosso sensor de presença, precisaremos de um computador e um dispositivo móvel com capacidade Bluetooth<sup>1</sup>. Nesse texto, iremos utilizar um celular como dispositivo móvel para a comunicação via Bluetooth.

Escolhemos o aplicativo BlueProximity<sup>2</sup> por ser intuitivo, simples e de fácil instalação, já que

é basicamente um script escrito em Python. Estamos utilizando o Fedora 12 (64bits) na confecção deste artigo, porém você poderia estar utilizando Ubuntu, Debian, Slackware ou qualquer outro sabor de linux.

Iremos executar a instalação do aplicativo no caso do Fedora utilizando a ferramenta yum<sup>3</sup>. O blueproximity já esta empacotado no formato RPM no repositório oficial do Fedora.

```
--> Processing Dependency: python-configobj for package: blueproximity-1.2.5-5.fc12.noarch
--> Running transaction check
--> Package pybluez.x86_64 0:0.16-2.fc12 set to be updated
--> Package python-configobj.noarch 0:4.6.0-2.fc12 set to be updated
--> Finished Dependency Resolution

Dependencies Resolved

=====
Package             Arch           Version           Repository        Size
=====
Installing:
blueproximity       noarch         1.2.5-5.fc12     fedora            275 k
Installing for dependencies:
pybluez             x86_64         0.16-2.fc12     fedora            81 k
python-configobj    noarch         4.6.0-2.fc12     fedora            168 k
=====
Transaction Summary
=====
Install      3 Package(s)
Upgrade     0 Package(s)

Total download size: 524 k
Is this ok [y/N]: _
```

Figura 1 - Ciclo de evoluções

Precisamos autorizar a comunicação entre o celular e o computador, para isso os dispositivos precisam estar pareados. No Fedora essa configuração é muito simples. O aplicativo bluetooth-wizard irá nos ajudar para parear os dispositivos:

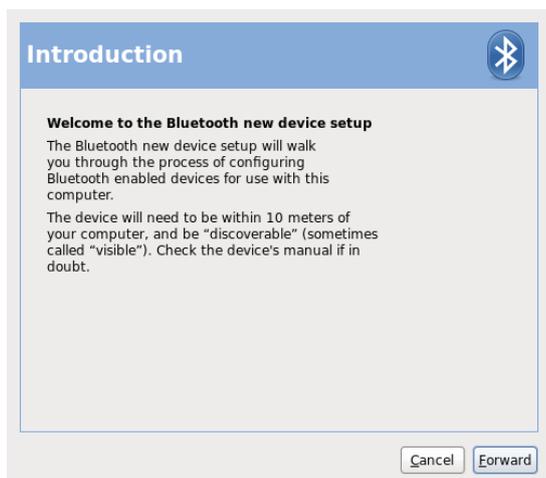


Figura 2 - Inicialização



Figura 2 - Inicialização



Figura 1 - Ciclo de evoluções

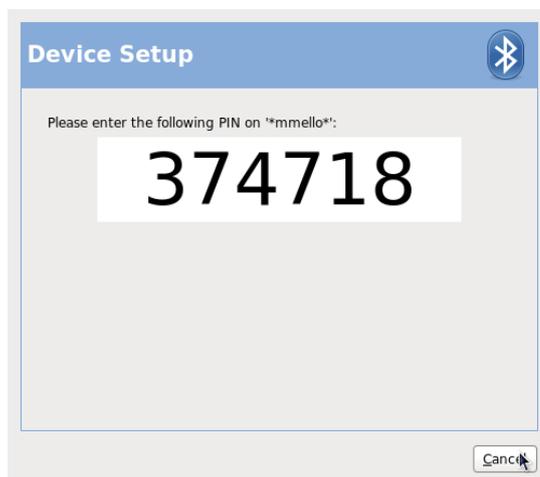


Figura 1 - Ciclo de evoluções

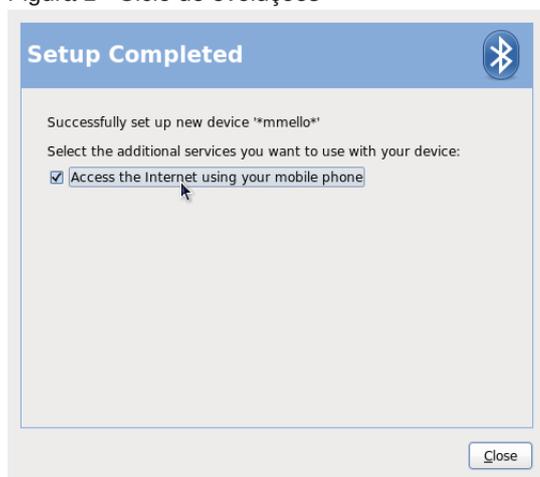


Figura 1 - Ciclo de evoluções

Uma vez instalado o aplicativo e assumindo também que os passos acima na configuração do adaptador bluetooth do computador e também do dispositivo móvel bluetooth já estão funcionando podemos iniciar o blueproximity executando o comando blueproximity na linha de comando ou pela interface gráfica como segue abaixo:

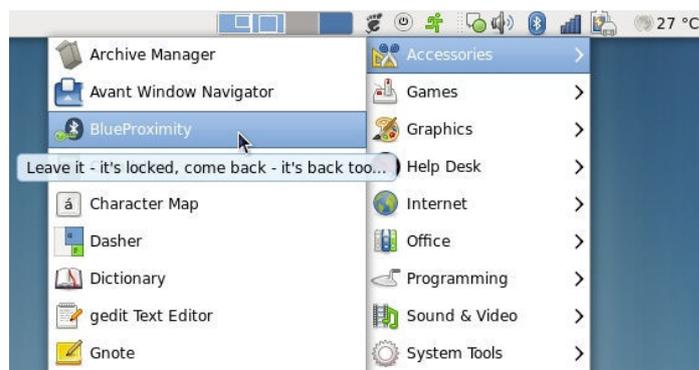


Figura 1 - Ciclo de evoluções

Iniciado o aplicativo na barra de tarefas do Gnome, o ícone do BlueProximity será exibido. Nossa configuração será toda gráfica a partir desse momento.

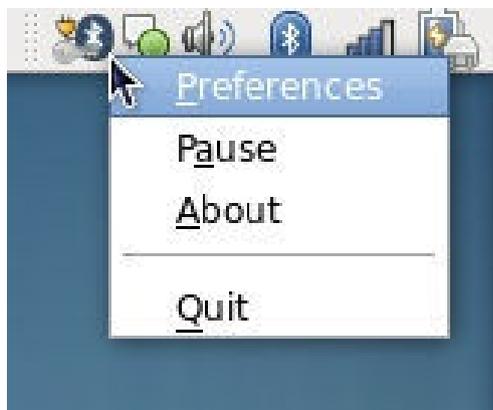


Figura 1 - Ciclo de evoluções

Precisamos agora configurar o dispositivo móvel para ser utilizado com blueproximity.

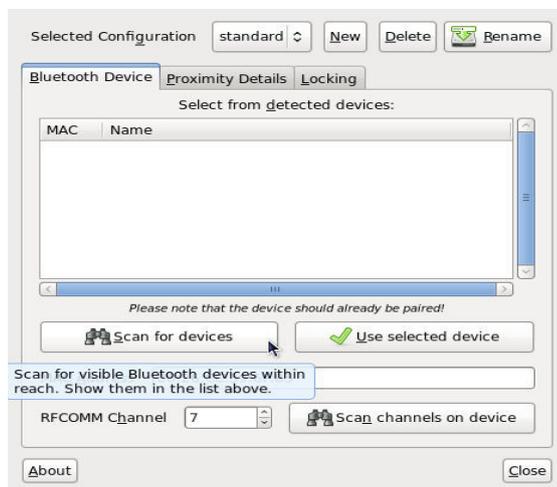


Figura 1 - Ciclo de evoluções

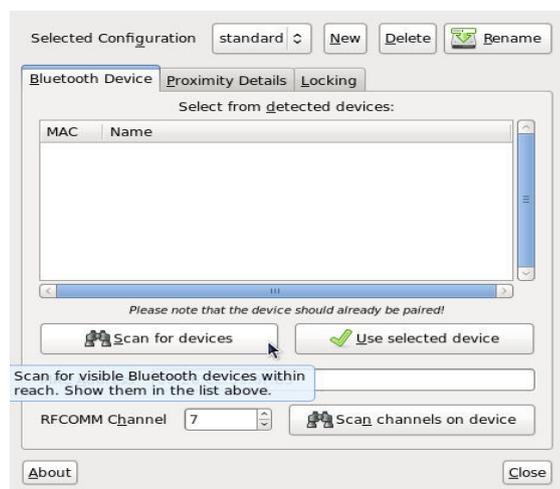


Figura 1 - Ciclo de evoluções

Selecione o dispositivo móvel bluetooth e clique no botão Use selected device (verifique se o MAC Address esta correto com o dispositivo móvel). Terminado essa etapa, o ícone do blueproximity deve estar verde indicando que o dispositivo esta próximo ao computador.



Figura 1 - Ciclo de evoluções

Ainda na tela de configuração do blueproximity, clique na aba Proximity Details para definir a proximidade do seu adaptador móvel bluetooth do computador para efetuar o bloqueio e desbloqueio da proteção de tela. O parâmetro de configuração Duration(s) determina quantos segundos o adaptador móvel deve permanecer na distância configurada para a ação (Lock/Unlock) ser executada. O campo Measured atm irá exibir a distância (em tempo real) entre o adaptador móvel e o computador.

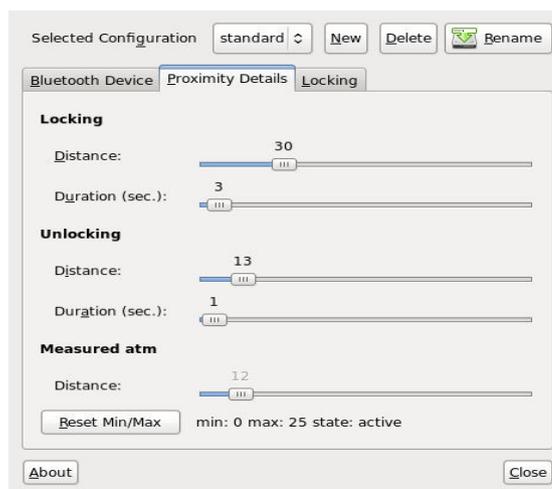


Figura 1 - Ciclo de evoluções

Ainda temos uma terceira aba que permite a configuração dos comandos que serão executados para bloquear e liberar a proteção de tela. O campo Proximity Command é um recurso que será executado a cada intervalo em segundos (Command Interval). Essa configuração permite que a tela não seja bloqueada enquanto você estiver por perto, por mais que você tenha ficado 30 minutos sem tocar no teclado, lembre-se, para o computador agora você esta por perto!

E por último ainda na terceira aba de configuração temos as seções de log onde podemos gerar um arquivo no diretório pessoal do usuário ou criar uma facility no syslog e direcionar o log para lá.

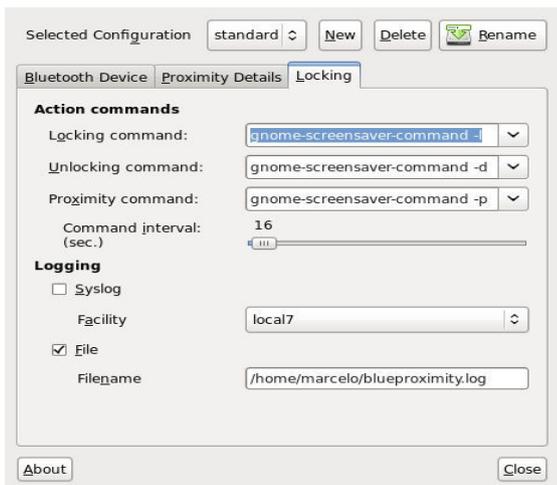


Figura 1 - Ciclo de evoluções



Figura 1 - Ciclo de evoluções

Pronto!!! Nosso aplicativo de controle de presença já esta configurado e pronto para uso!! A começar de agora, a desculpa do acesso à internet, a partir de sua máquina por tê-la esquecido logada ao ir pegar um café na copa não será mais aceita!! :) 

## Para mais informações:

Site Oficial Ubuntu:

<http://www.ubuntu.com>

Informações sobre o Alpha 3 do Ubuntu 10.04:

<http://www.ubuntu.com/testing/lucid/alpha3>

Artigo na Wikipédia sobre o Ubuntu:

<http://pt.wikipedia.org/wiki/Ubuntu>



**MARCELO MOREIRA MELLO** nasceu em São Paulo mas tem coração gaúcho. Trabalha com linux desde 2001 e atualmente atua como instrutor e consultor na Red Hat Brasil. Tendo foco especialmente em tecnologias open-source possui várias certificações, tais como: RHCE, RHCI, RHCX, RHCDS, RHCA, RHCVA. Como segunda paixão, gosta de atuar como Chef de Cozinha :).



# COMO CRIAR UM LIVECD DO UBUNTU "DO ZERO"

Por Filipo Tardim

O Ubuntu<sup>1</sup> é a distribuição Linux mais famosa do mundo, e ganha novos adeptos a cada release semestral.

O LiveCD, que pode ser baixado ou enviado gratuitamente para sua residência através do serviço [shiptit](#), possui o básico de um sistema desktop: suíte de escritório, editor de imagens, navegador web, player multimídia, gravador de CD, além de um grande suporte aos hardwares, que o faz reconhecer automaticamente a placa de som, de vídeo, rede e instalar uma impressora apenas conectando o cabo USB no PC.

No entanto, o sistema também possui suporte a alguns dispositivos que a maioria dos usuários pode nunca vir a usar, como Bluetooth, dispositivos PalmOS, dentre outros. Por outro lado, alguns recursos muito usados não podem vir por padrão, por serem proprietários, como o adobe flashplayer, os codecs para mp3, etc.

Alguns usuários têm preferência por outros programas em alternativa aos que vêm no CD.

Eu, por exemplo, utilizo o Deluge como cliente torrent, no lugar do Transmission, e instalo o Thunderbird no lugar do Evolution.

Além disso, quantos já não instalaram e, logo depois, tiveram de baixar quase um novo CD só de atualizações?

Uma das grandes características do Linux é dar liberdade ao usuário para personalizar o sistema como bem entender. Com o Ubuntu não é diferente. Qualquer pessoa pode recriar o LiveCD, colocando os programas que desejar.

Existe uma série de ferramentas disponíveis para 'remasterizar' o Ubuntu. Uma das mais famosas e mais fáceis de usar é o [Remastersys](#). Este programa cria um LiveCD bootável com o sistema que está instalado no PC. Outros dois programas muito bons são o [Reconstructor](#) e o [Ubuntu Customization Kit \(UCK\)](#). A diferença destes para o Remastersys é que você não precisa ter o sistema instalado, pois ambos utilizam a própria imagem ISO baixada da Internet.

Neste artigo vou mostrar como criar um LiveCD do Ubuntu de uma forma totalmente manual, sem utilizar qualquer programa (com exceção do famoso terminal) e sem a necessidade de ter o CD do Ubuntu nas mãos, ou em ISO. É um processo trabalhoso mas muito compensatório. Dentre os objetivos que posso listar sobre a opção de remasterizar manualmente, os mais importantes são:

- Adquirir maior conhecimento sobre o Ubuntu e sobre o Linux em geral;
- Ter maior domínio sobre o sistema, controlando quais programas estarão disponíveis;
- Gerar uma instalação atualizada, garantindo agilidade em futuras instalações;

Vamos lá então?

Antes de começarmos, uma observação importante. Segundo o [guia oficial](#), é possível gerar um sistema anterior mas não um superior, ao menos que baixe a versão do pacote “[debootstrap](#)” específica para o sistema a ser gerado. Em outras palavras, se você está no 9.04 Jaunty, poderá gerar qualquer versão anterior, mas não poderá gerar uma ISO do 9.10 Karmic, a menos que baixe o debootstrap para ele.

Também não recomendo usar o Karmic para gerar releases anteriores, devido às mudanças que ocorreram no sistema, e por usar o squashfs 4.0 (programa responsável para compactar o sistema que será usado no LiveCD) e o Grub2. Até o 9.04, era usado o Grub e o squashfs 3.2. Nos testes que fiz, ao gerar a ISO no 9.10, o LiveCD apresentou kernel panic.

Vou, então, gerar um LiveCD do Ubuntu 9.04 Jaunty Jackalope i386, pois esse exemplo servirá para os lançamentos mais antigos, sem deixar de destacar as diferenças do Karmic.

Cabe lembrar que a maioria dos passos aqui descritos também servem para gerar um CD personalizado do Debian, e a arquitetura i386 pode ser gerada de dentro de um sistema amd64 sem problemas. Eu mesmo estou no Jaunty 64 bits.

### Primeiros Passos – baixando os pacotes do sistema base

Vou adotar como diretório raiz para o sistema a ser gerado, a pasta /home/remaster. Geralmente a /home fica numa partição separada e possui bastante espaço livre (vai precisar).

Outra observação: o “\$” simboliza o usuário comum e o “#”, o usuário root. Eles não fazem parte dos comandos. Apesar de todos os comandos, com exceção do primeiro, serem feitos como root, mantereí o sinal algumas vezes para saber que se trata de uma nova entrada, visto que alguns ultrapassarão o limite da margem. Toda vez que vir uma faixa preta com letras brancas, trata-se de um comando do terminal. Para facilitar, selecione tudo, copie, cole num editor de texto (gedit, kate, etc), remova os \$ e #, e uma as linhas quebradas. Por fim, copie e cole no terminal tudo de uma vez.

Com o terminal aberto, faça os seguintes comandos:

```
$ sudo su
# apt-get install debootstrap syslinux squashfs-tools
genisoimage sbm casper
```

O debootstrap é o pacote responsável de baixar o sistema base inicial, e como já foi dito, deve ser baixado e instalado manualmente caso queria gerar uma versão superior à que está instalada na máquina. Os outros pacotes são responsáveis para a compactação do sistema e criação de um CD de boot.

Dando prosseguimento, faça agora os seguintes comandos:

```
mkdir -p /home/remaster/isos
mkdir -p /home/remaster/custom/cd
mkdir -p /home/remaster/custom/squashfs
cd /home/remaster/custom
debootstrap --arch=i386 jaunty squashfs
```

E o sistema será baixado e instalado na pasta /home/remaster/custom/squashfs.

## Conectando o sistema na Internet e instalando programas essenciais

Vamos copiar os seguintes arquivos para o sistema recém-instalado:

```
cp /etc/resolv.conf /etc/hosts squashfs/etc/
```

Se você está editando a mesma versão que está instalada no PC, pode usar os mesmos repositórios:

```
cp /etc/apt/sources.list squashfs/etc/apt/  
cp /etc/apt/trusted.gpg squashfs/etc/apt/
```

Se você não está usando a mesma versão, edite o arquivo `squashfs/etc/apt/sources.list` e habilite os repositórios que quiser. Os oficiais são<sup>2</sup>:

```
deb http://br.archive.ubuntu.com/ubuntu/ jaunty main  
restricted universe multiverse  
deb-src http://br.archive.ubuntu.com/ubuntu/ jaunty main  
restricted universe multiverse  
  
deb http://br.archive.ubuntu.com/ubuntu/ jaunty-updates  
main restricted universe multiverse  
deb-src http://br.archive.ubuntu.com/ubuntu/ jaunty-  
updates main restricted universe multiverse  
  
# deb http://br.archive.ubuntu.com/ubuntu/ jaunty-  
backports main restricted universe multiverse  
# deb-src http://br.archive.ubuntu.com/ubuntu/ jaunty-  
backports main restricted universe multiverse  
  
deb http://archive.canonical.com/ubuntu jaunty partner  
deb-src http://archive.canonical.com/ubuntu jaunty  
partner  
  
deb http://security.ubuntu.com/ubuntu jaunty-security  
main restricted universe multiverse  
deb-src http://security.ubuntu.com/ubuntu jaunty-security  
main restricted universe multiverse
```

Mude o nome 'jaunty' para a versão que estiver trabalhando.

Agora vamos entrar na distro baixada:

```
mount --bind /dev/ squashfs/dev  
chroot squashfs
```

Seu terminal irá mudar e a partir daqui todos os comandos serão feitos de dentro do sistema a ser editado. Mas antes de começar a instalar programas, precisamos montar os seguintes sistemas de arquivos:

```
mount -t proc none /proc  
mount -t sysfs none /sys  
mount -t devpts none /dev/pts  
export HOME=/root  
export LC_ALL=C
```

Agora, atualize a lista de pacotes com o comando "apt-get update".

Se você adicionou algum repositório de terceiros e após o update recebeu alguma mensagem de erro como essa: "NO\_PUBKEY 07DC563D1F41B907", dê o seguinte comando:

```
# apt-key adv --keyserver keyserver.ubuntu.com --recv-  
keys 07DC563D1F41B907
```

Substitua o código alfanumérico pelo informado no terminal (apenas os 8 últimos dígitos são necessários).

O Karmic possui um bug ao realizar atividades como chroot. Para amenizar as mensagens de erro, faça os seguintes comandos:

```
# apt-get install --yes dbus  
# dbus-uuidgen > /var/lib/dbus/machine-id  
# dpkg-divert --local --rename --add /sbin/initctl  
# ln -s /bin/true /sbin/initctl
```

Agora, com os repositórios atualizados, vamos instalar os seguintes pacotes:

```
# apt-get install ubuntu-standard casper lupin-casper  
discover1 laptop-detect os-prober linux-generic linux-  
headers-generic
```

Dos pacotes acima, três são opcionais. O `lupin-casper` serve para dar suporte a uma instalação dentro do Windows (Wubi); o `laptop-detect` verifica se o LiveCD está rodando em um

laptop ou desktop; o linux-headers-generic deixa o sistema mais robusto, mas é essencial para compilar alguns módulos, sobretudo os das placas de vídeo, e por isso vem por padrão no CD do Ubuntu.

### Instalando o Xorg e o instalador gráfico

```
apt-get install xorg ubiquity-frontend-gtk
```

Obs: se você for instalar o ambiente KDE, instale o pacote "ubiquity-frontend-kde" no lugar do gtk.

A partir desse ponto, você pode instalar o programa que quiser. Atente-se apenas para a quantidade, pois caso instale muitos programas, o resultado final ultrapassará o limite do CD (700 MB) e você terá de usar uma mídia de DVD, o que também não é problema nenhum, a menos que você não tenha um leitor de DVD (lembrando que a ISO também pode ser transferida para um pendrive).

Agora você já tem o sistema pronto para rodar e instalar a partir de um LiveCD. Vamos agora para a criação do CD de boot, mas antes de sair do ambiente, precisamos limpar o cache e desmontar os sistemas de arquivos:

```
apt-get clean
rm -rf tmp/*
rm -r /etc/resolv.conf
rm -r /etc/hosts
umount -lf /proc
umount /sys
umount /dev/pts
exit
umount -lf squashfs/dev
rm squashfs/root/.bash_history
```

Se está utilizando o Karmic, e fez os comandos destacados anteriormente, execute estes antes do apt-get clean:

```
# rm /var/lib/dbus/machine-id
# rm /sbin/initctl
# dpkg-divert --rename --remove /sbin/initctl
```

### Criando o CD de boot

Se você tem o CD da versão que está editando, facilitaria sua vida ao copiar todo seu conteúdo, com exceção da pasta casper, para a pasta /home/remaster/custom/cd. Caso tenha a .iso no HD, faça os seguintes comandos a partir da pasta onde o arquivo está:

```
# mount -o loop ubuntu-9.04-i386.iso /mnt/
# rsync --exclude=/casper/filesystem.squashfs -a /mnt/
/home/remaster/custom/cd
```

Mas como o presente tutorial aborda uma criação "do zero", vamos criar os principais arquivos para que o CD funcione. Primeiramente, os diretórios:

```
# mkdir -p cd/{casper,isolinux,install,preseed,.disk}
```

#### - casper

É dentro dessa pasta que fica o sistema que acabamos de criar, de forma compactada (filesystem.squashfs), bem como os arquivos responsáveis pelo boot e a lista de pacotes instalados. É portanto a pasta principal. Vamos primeiro criar os arquivos de boot:

```
# cp squashfs/boot/vmlinuz-2.6.**-**-generic
cd/casper/vmlinuz
# casper-new-uuid
/home/remaster/custom/squashfs/boot/initrd.img-2.6.**-
**-generic /home/remaster/custom/cd/casper/
/home/remaster/custom/cd/.disk/
```

Obs: como pode notar pelos #, só existem dois comandos nas linhas acima.

No Karmic, o último comando não funciona. Além disso, no novo sistema o arquivo initrd.gz precisa ser recompactado para o formato .lz. Proceda da seguinte forma:

```
# cp squashfs/boot/initrd.img-2.6.**-**-generic
  cd/casper/initrd.gz
# cd cd/casper
# mkdir initrdtmp
# cd initrdtmp
# gzip -dc ../initrd.gz | cpio -imvd --no-absolute-filenames
# find . | cpio --quiet --dereference -o -H newc | lzma -7 >
  ../initrd.lz
# cd ..
# rm -r initrdtmp initrd.gz
# cd ../..
```

Vamos agora criar o manifest:

```
# chroot squashfs dpkg-query -W --
  showformat='${Package} ${Version}\n' | tee
  cd/casper/filesystem.manifest
# cp -v cd/casper/filesystem.manifest
  cd/casper/filesystem.manifest-desktop
# REMOVE='ubiquity casper live-initramfs user-setup
  discover1 xresprobe os-prober libdebian-installer4'
# for i in $REMOVE
# do
# sed -i "/${i}/d" cd/casper/filesystem.manifest-desktop
# done
```

Você acaba de criar dois arquivos. O `filesystem.manifest` contém a lista de pacotes atualmente instalados. Já o `filesystem.manifest.desktop` é a lista de pacotes que permanecerão após a instalação. Segundo a linha que começa com "REMOVE", o instalador (Ubiquity) removerá todos os pacotes que foram mencionados ali, incluindo ele próprio.

Agora, o principal, o sistema de arquivos compactado:

```
# mksquashfs /home/remaster/custom/squashfs
  /home/remaster/custom/cd/casper/filesystem.squashfs
```

Esse processo pode demorar, de acordo com a quantidade de pacotes que você instalou e com a velocidade de seu computador.

## - install

Copie os seguintes arquivos para a pasta:

```
# cp /boot/memtest86+.bin cd/install/memtest
# cp /boot/sbm.img cd/install/
```

## - isolinux

Copie os seguintes arquivos para a pasta:

```
# cp /usr/lib/syslinux/isolinux.bin
  /usr/lib/syslinux/vesamenu.c32 cd/isolinux/
```

Crie um arquivo com o comando "nano `cd/isolinux/isolinux.cfg`" e cole o seguinte conteúdo:

```
default vesamenu.c32
timeout 300

menu background splash.png
menu title Ubuntu 9.04 *Remix*
menu color title 0 #ff85b4f5 #00000000
menu color border 0 #0085b4f5 #00000000
menu color sel 7 #ffffff #33e0e0
menu color hotssel 0 #ff000000 #ffffff
menu color tabmsg 0 #ff000000 #00000000
menu color unsel 0 #ff000000 #00000000
menu color hotkey 7 #ffffff #ff000000
menu color timeout_msg 0 #ffffff #00000000
menu color timeout 0 #ffffff #00000000
menu color cmdline 0 #ffffff #00000000
menu vshift 12
menu rows 10
menu tabmsgrow 16
menu timeoutrow 17
menu tabmsg Pressione ENTER para boot ou TAB
  para editar uma entrada no menu

label live
  menu label Testar o Ubuntu sem qualquer mudança
  no seu computador
  kernel /casper/vmlinuz
  append file=/cdrom/preseed/ubuntu.seed
  boot=casper initrd=/casper/initrd.gz locale=pt_BR
  quiet splash --
  menu default
```

```
LABEL xforcevesa
menu label Iniciar em Modo Grafico Seguro
kernel /casper/vmlinuz
append file=/cdrom/preseed/ubuntu.seed boot=casper
xforcevesa initrd=/casper/initrd.gz locale=pt_BR quiet
splash --

label live-install
menu label Instalar o Ubuntu
kernel /casper/vmlinuz
append file=/cdrom/preseed/ubuntu.seed boot=casper
only-ubiquity initrd=/casper/initrd.gz locale=pt_BR quiet
splash --

label check
menu label Verificar se ha defeito no CD
kernel /casper/vmlinuz
append boot=casper integrity-check
initrd=/casper/initrd.gz quiet splash --

label memtest
menu label Teste de memoria
kernel /install/memtest

label hd
menu label Inicializar pelo primeiro disco rigido
localboot 0x80
```

Obs: troque o nome de initrd.gz para initrd.lz se estiver usando o Karmic.

Crie uma imagem em 640x480 com o nome de splash.png e coloque nessa pasta.

## - preseed

Crie um arquivo com o comando “nano cd/preseed/ubuntu.seed” e cole o seguinte conteúdo (do jeito que está, com os #):

```
# Only install the standard system and language packs.
tasksel tasksel/first multiselect
d-i pkgsel/language-pack-patterns string
# No language support packages.
d-i pkgsel/install-language-support boolean false
```

## - .disk

Faça os seguintes comandos:

```
# cd cd/.disk
# touch base_installable
# echo "full_cd/single" > cd_type
# echo 'Ubuntu 9.04 "Jaunty Jackalope Remix" - i386
(20090429)' > info
# echo "http://ubuntu-rescue-remix.org" >
release_notes_url
# cd ../..
```

## - Raiz do CD

Crie um arquivo com o comando “nano cd/README.diskdefines” e cole o seguinte conteúdo (do jeito que está):

```
#define DISKNAME Ubuntu 9.04 "Jaunty Jackalope" -
Release i386 **Remix**
#define TYPE binary
#define TYPEbinary 1
#define ARCH i386
#define ARCHi386 1
#define DISKNUM 1
#define DISKNUM1 1
#define TOTALNUM 0
#define TOTALNUM0 1
```

Caso queira o recurso de instalar dentro do Windows, baixe o wubi.exe aqui:

<http://wubi-installer.org/>

<http://sourceforge.net/projects/wubi/>

Copie-o para dentro da pasta cd e crie um autorun:

```
[autorun]
open=wubi.exe --cdmenu
icon=wubi.exe,0
label=Install Ubuntu
```

```
[Content]
MusicFiles=false
PictureFiles=false
VideoFiles=false
```

Crie, agora, um link para o CD:

```
# ln -s . ubuntu
```

Por fim, crie o arquivo md5sum.txt:

```
# cd cd
# find . -type f -print0 | xargs -0 md5sum | grep -v
"../md5sum.txt" > md5sum.txt
```

Com isso, todos os arquivos necessários estarão no CD, bastando agora criar a ISO com o seguinte comando:

```
# mkisofs -r -V "ubuntu-9.04-i386-remix" -b
isolinux/isolinux.bin -c isolinux/boot.cat -cache-inodes -
J -l -no-emul-boot -boot-load-size 4 -boot-info-table -o
../../isos/ubuntu-9.04-i386-remix.iso .
```

Sua ISO será gerada na pasta /home/remaster/isos. Utilize agora o VirtualBox ou outro programa para testar.

Em breve mostrarei como editar uma ISO pronta, adicionando e removendo programas. Também trarei uma lista de programas que podem ser instalados, e várias dicas de personalização (papel de parede, ícones, etc).

Até lá! 🇧🇷



Figura 1: sistema customizado rodando no VirtualBox

## Para mais informações:

[1] Site oficial Ubuntu:

<http://www.ubuntu.com>

[2] Mais detalhes sobre os repositórios podem ser encontrados na matéria de Igor Morgado, Ed. #10 da Revista Espírito Livre.

<http://revista.espiritolivre.org>



FILIPPO TARDIM é graduando em Letras pela UFRJ, professor da rede municipal de Duque de Caxias, ajuda a implementar o Linux nas escolas públicas do município e participa do desenvolvimento do Lineduc – Linux Educacional Duque de Caxias.





# Google Reader no Ubuntu

Por Márcio Rodrigues Pivoto

Uma coisa que eu sempre procurei no Ubuntu foi um agregador de Feeds que fosse possível sincronizar com a minha conta do Google Reader, procurei vários mais nenhum com tal integração, ate que um dia pelas minhas andanças pela internet encontrei o Read AIR. Um agregador de Feeds feito em AIR leve, rápido totalmente compatível com o Google Reader.

## Vamos a Instalação:

Antes de começarmos a instalação do Read Air primeiro precisamos instalar o Adobe Air, para isso devemos ir no site oficial da Adobe e baixar a instalação do Adobe Air.

<http://get.adobe.com/br/air/otherversions/>

Uma vez baixado vamos a instalação, ele vem no formato .bin para instalar arquivos com essa terminação devemos usar o Terminal.

Lembrando que se o arquivo estiver na pasta Downloads primeiro devemos entrar no diretório com o comando:

```
cd Downloads
```

No diretório onde se encontra o arquivo AdobeAIRInstaller.bin digite:

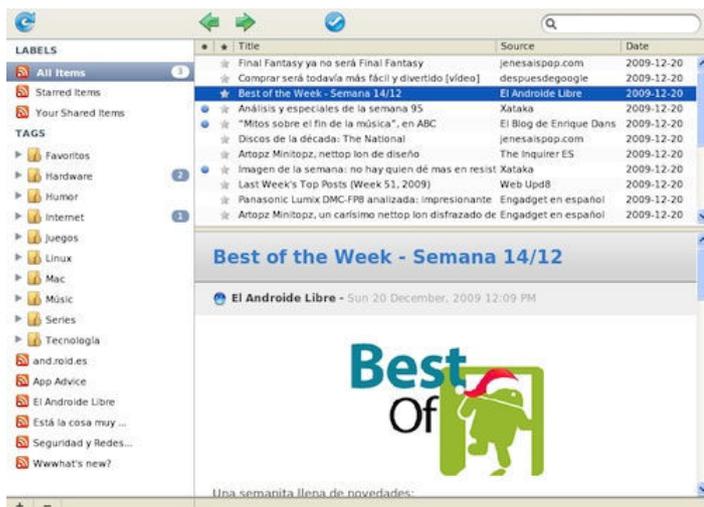


Figura 1 - Google Reader

```
chmod +x AdobeAIRInstaller.bin
```

Agora Digite isso para iniciar a instalação:

```
sudo ./AdobeAIRInstaller.bin
```

Esses comandos irão dar início a instalação do Adobe AIR.



Figura 2 - Início da instalação

Apenas clique no botão concordo.

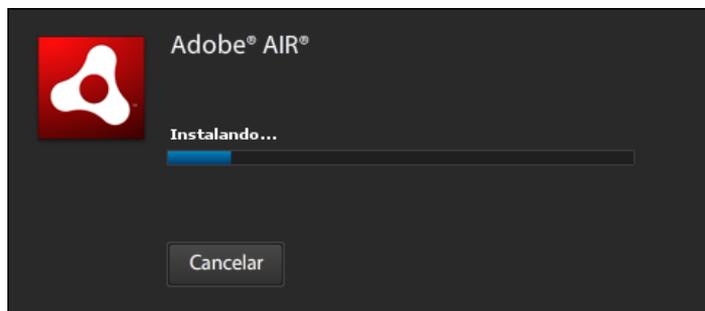


Figura 3 - Instalando o Adobe AIR

A instalação é bem rápida e fácil.

Agora devemos baixar o aplicativo Read Air, nosso leitor de Feeds.

Para isso devemos ir ao site oficial do Projeto e baixar a última versão do aplicativo.

<http://code.google.com/p/readair/>

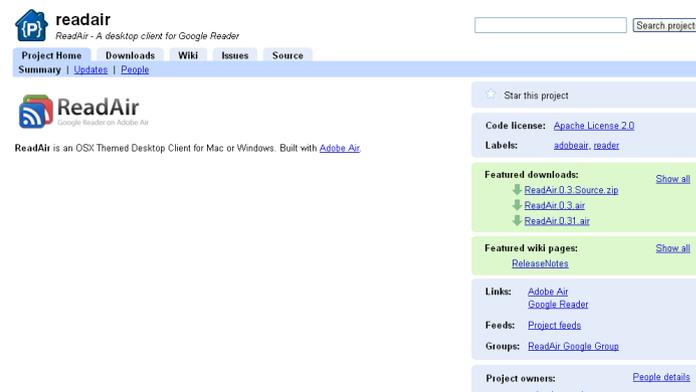


Figura 4 - Página do Projeto ReadAir

No site escolha no quadro verde ReadAir 0.31.air.

Uma vez baixado, um duplo clique no arquivo instala o aplicativo facilmente.

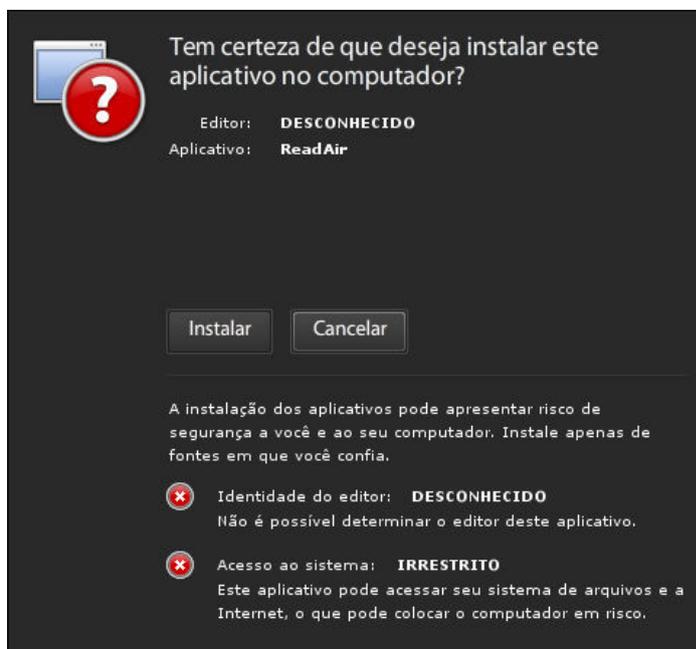


Figura 5 - Início da instalação do ReadAir

Apenas clique em instalar...

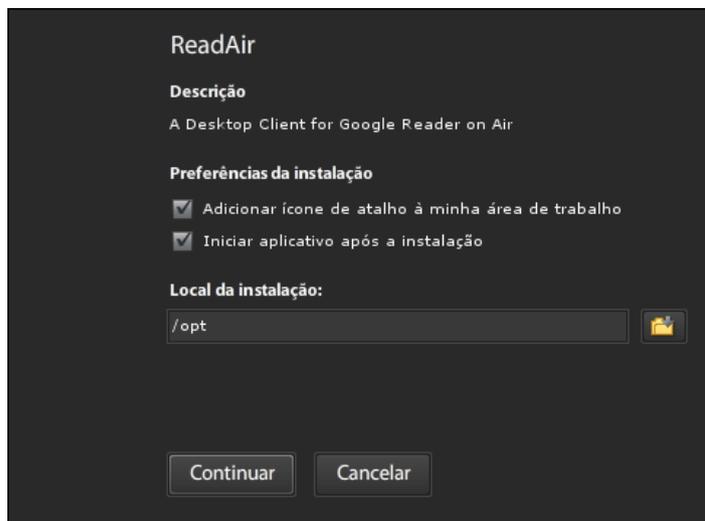


Figura 6 - Processo de instalação

...Depois em continuar, sem mudar nada nessa tela.

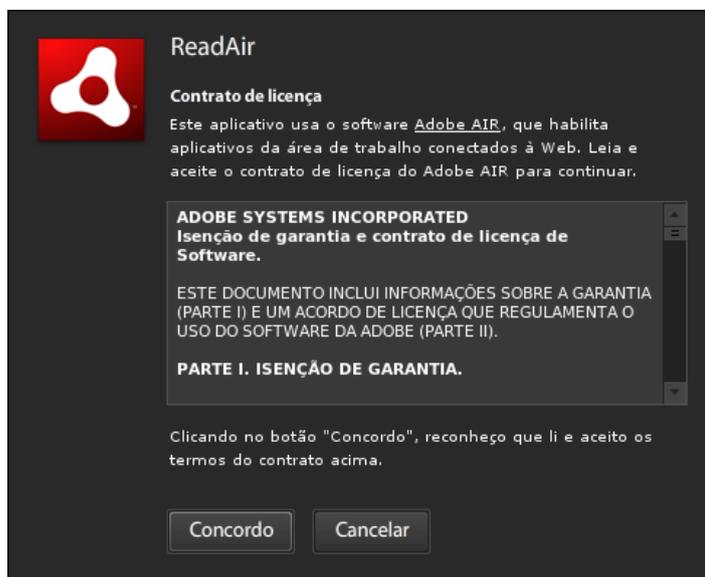


Figura 7 - Contrato de licença do ReadAir

Concorde com os termos e aguarde o fim da instalação.

Agora para ter acesso ao aplicativo devemos ir no menu: Aplicativos > Acessórios > Read Air.

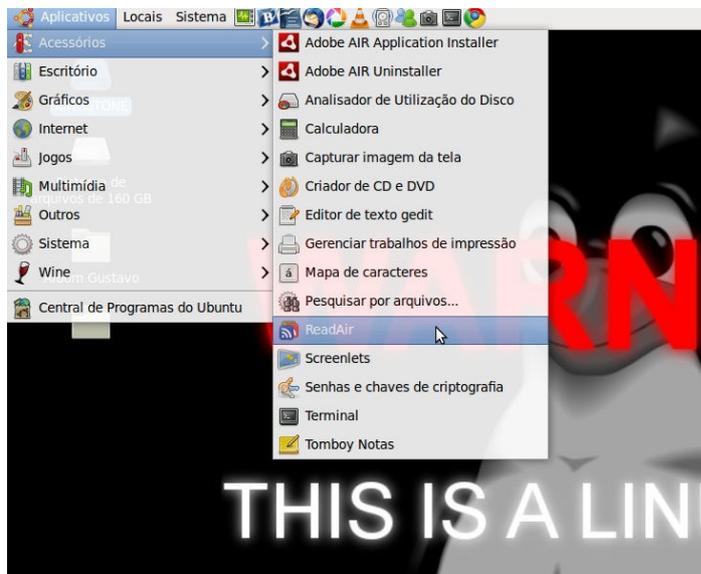


Figura 8 - ReadAir já instalado no Ubuntu

Para usá-lo devemos colocar o e-mail e senha que usamos no Google.

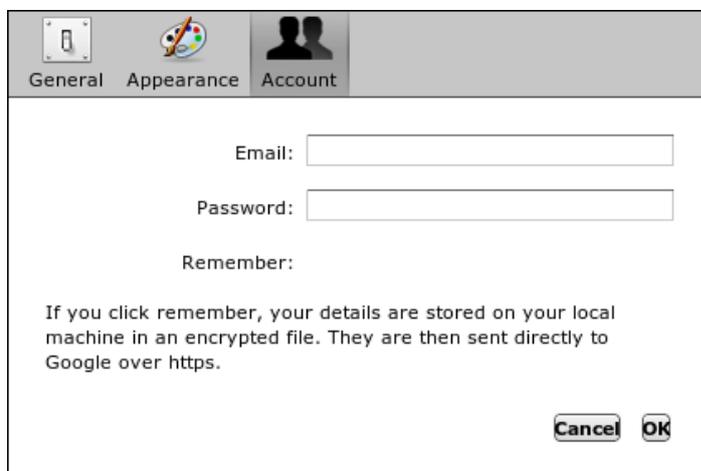


Figura 9 - Configurando contas de usuário do serviço

E pronto automaticamente ele vai começar a sincronizar com sua conta do Google Reader. 



**MÁRCIO RODRIGUES PIVOTO** é técnico de informática, amante de software livre, principalmente de Linux, especialmente do Ubuntu. Responsável pelo blog <http://ubuntuaazip.blogspot.com>.

# GIT: Fazendo controle de versões da forma certa

Por Gleudson Junior

Flavio Takemoto - sxc.hu

## 1. O que é controle de versionamento distribuído?

Os VCS<sup>1</sup> distribuídos, também chamado de DVCS<sup>2</sup> tem uma arquitetura muito mais complexa e até um tanto quanto complicada para compreensão de iniciantes, em contra partida nos apresenta uma série de vantagens que não são possíveis de aplicar ao modelo centralizado.

Dando vista ao processo no âmbito operacional, um DVCS não é muito desigual a um VCS centralizado, pois os desenvolvedores de um determinado projeto também fazem commit de suas alterações durante o desenvolvimento dos arquivos, da mesma forma quem geram tarballs, patches e atualizam seus arquivos ou analisam seu histórico de alteração.

A grande sacada do modelo distribuído está no fato de não haver um único repositório centralizado, mas sim diversos repositórios espalhados na rede e um repositório para cada desenvolvedor, sob a dependência da ferramen-

ta que está sendo utilizada pelo projeto. Desta maneira, por exemplo, pode-se fazer com que os repositórios “filhos” de um mesmo repositório “pai” compartilhem suas alterações paralelamente entre si.

Em um sistema de controle distribuído, uma conexão permanente com o servidor não é um requisito essencial para o funcionamento do sistema, visto que cada desenvolvedor é responsável por manter seu próprio repositório, dessa forma todas as alterações feitas no longo do desenvolvimento podem passar por um commit local e então somente quando necessário ou solicitado, devem atingir o repositório “pai”, ou paralelos a ele, caso exista algum. Incrementado a esse contexto, o “pai” destes repositórios “filhos” podem realizar um commit destas mudanças, no seu repositório principal, tornando com que suas modificações se tornem públicas para os demais membros do conjunto de repositórios.

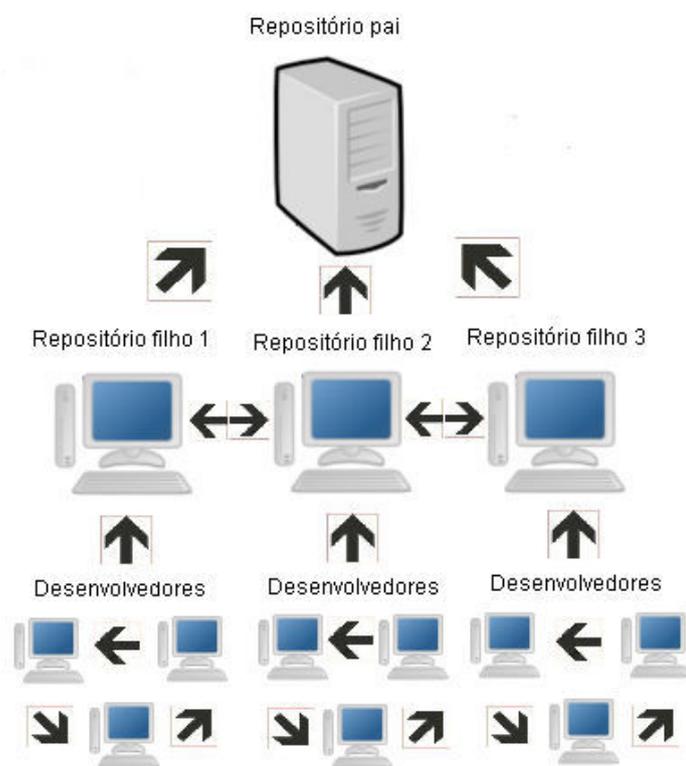


Figura 1: Modelo de controle de versão distribuído

Organizar e acessar os repositórios desta maneira torna os DCVS totalmente aplicáveis para equipes de trabalho que estão geograficamente espalhados pelo mundo, ao qual nem sempre possuem uma conexão estável entre os colaboradores do projeto. Porém isso não quer dizer que uma conexão de rede, ou ao menos uma forma de disseminar estas mudanças não seja importante para o funcionamento de DCVS, acontece que quando um desenvolvedor utiliza um sistema de controle distribuído ele pode ter acesso às informações do repositório “pai”, mesmo que não exista uma conexão com o mesmo. No entanto estas informações dizem respeito à última sincronização do repositório local, com o repositório paralelo ou o repositório “pai”. Além do mais, o fato de ser distribuído não impede necessariamente que o repositório “filho” replique suas alterações para o repositório “pai”, pois se isso não acontecesse, as suas modificações não seriam reproduzidas para os demais colaboradores do projeto.

Podemos citar como um exemplo semelhan-

te de utilização e complexidade desse modelo, as trocas (downloads e uploads) de arquivos na plataforma p2p, onde se torna um pouco mais difícil o controle sobre o acesso aos dados existentes em um determinado repositório. A partir disso é que vem a procura e empregabilidade de algumas ferramentas que possuem funcionalidades específicas capazes de solucionar este tipo de problema.

## 2. Conhecendo e entendendo o Git

Enquadrada a um sistema de controle de versão distribuído (DVCS) podemos encontrar uma aplicação bastante eficiente, que tende a propor diversos recursos a fim de sanar os problemas supracitados, estamos falando do projeto Git.

Ele é um projeto de Software Livre, liberado sob a licença GPL<sup>1</sup>, que foi inicialmente criado por Linus Torvalds<sup>2</sup>, visando o controle e desenvolvimento dos códigos fontes do Kernel Linux, hoje ele é mantido por Junio Hamano. O grande objetivo do Git é atender a todas as exigências do modelo de desenvolvimento do Kernel Linux, tais como manipulação de grandes conjuntos de arquivos, desenvolvimento de projeto distribuído, operação de ramificações e mesclagens (merges) complexas, integridade do código, desempenho, entre outras.

Cada diretório local de desenvolvimento Git é tido como um repositório, onde se encontra todo histórico de desenvolvimento dos commits locais e a capacidade de controle das revisões do código, tudo isso sem a dependência de um acesso direto a uma rede ou a um único servidor central.

O desenho inicial do Git foi inspirado por outras duas aplicações de controle de versão: o Bitkeeper, e o Monotone. Seu projeto original possuía uma arquitetura com um mecanismo de funcionamento bastante abstrato, difícil para a compreensão de desenvolvedores iniciantes. Com a evolução e a aparição de novas necessidades, o projeto original acabou se tornando em

um sistema de controle de versão completo, que passou a ser adotado por empresas de alto nível.

São inúmeros projetos de software que hoje empregam o Git para o controle de versão de código, podendo citar como principais exemplos: Android, Btrfs da Oracle, Gnome, Google, Kernel Linux, Perl, Qt (toolkit), Ruby on Rails, Samba, Servidor X.org.

### 3. Por que empregar o Git?

É totalmente indiscutível que todo projeto de software independente do seu tamanho ou complexidade, necessita hoje de um sistema para versionamento de código. Isso não é mais uma questão de escolha. No momento em que se inicia um projeto de software, mesmo que este esteja sendo manipulado por apenas um desenvolvedor, o controle de versões passa a não ser uma tarefa de teor trivial, ou algo que se possa considerar opcional.

Daí é que surge o grande problema da maioria dos projetos de software: qual VCS utilizar? Grande parte dos desenvolvedores se vê hoje mal acostumados com a utilização e manipulação do CVS, Subversion ou mesmo o SourceSafe da Microsoft. É bem verdade que alguns deles atendem aos requisitos de determinados projetos de forma aceitável, nem sempre satisfatória, mas felizmente hoje podemos contar com ferramentas que oferecem funcionalidades bem mais evoluídas nesse âmbito.

O Linus Torvalds, por exemplo, sofria desse mal: como controlar uma gama enorme de colaborações provenientes de todo o mundo? Nos primórdios a maneira de controle dos códigos se constituía num processo manual, onde se testava e implantava manualmente retalhos de código, mas isso com o passar do tempo tendia a se tornar impraticável, por questões que envolviam o grande esforço de trabalho dos envolvidos no projeto. Quando ele sentiu a necessidade de empregar um sistema para controlar os códigos do núcleo Linux, decidiu de pronto não utilizar nem

o CVS, nem o Subversion, por limitações já descritas. Então optou pelo Bitkeeper, que apesar de ser comercial, se enquadrava perfeitamente as necessidades do projeto. No entanto, como explicar e principalmente motivar a comunidade de um software livre a utilizar um aplicativo comercial? Perante essas dificuldades, decidiu então desenvolver seu próprio sistema para controle de versões, optando pelo modelo distribuído, daí nasceu o Git.

Citação: “Ser distribuído significa que você não tem uma localização central que se mantém a par de seus dados, um único lugar, mais importante que qualquer outro lugar. Este modelo centralizado não funciona...”.

Linus Torvalds – Palestra ministrada sobre o Git no Tech Talk Google.

Nos sistemas de controle distribuídos, cada colaborador do projeto obtém em sua máquina local um clone completo do repositório principal. O que difere o Git das demais ferramentas, é justamente a grande escala de otimização desse repositório. Um dos exemplos clássicos dessa otimização, é o que acontece com alguns projetos, onde a última revisão de código é apenas um pouco maior que todo o código armazenado no repositório, incluindo atualizações e modificações.

A grande jogada está na estrutura desse sistema, onde todo o repositório encontra-se armazenado localmente, como isso o desenvolvedor não tem nenhum problema com permissões de escrita. Está tudo armazenado localmente na sua estação de trabalho. Além da possibilidade de trabalhar no desenvolvimento do projeto em modo off-line, sem necessitar de um acesso a rede nem internet, gerando assim maior comodidade e mobilidade aos colaboradores.

No Git cada desenvolvedor tem o repositório completo em sua estação de trabalho, que também é conhecido como “repositório central”,

mas que na verdade é apenas um dos clones que foi eleito como “repositório principal” e que caso sofra uma perda irreversível, deve ser substituído por qualquer um desses outros repositórios. Isso mantém viva a arquitetura distribuída do desenvolvimento do software, onde se garante que qualquer um dos repositórios clonados possa ser eleito como “repositório principal”. O que acontece é que ao invés de se fazer um “checkout” e copiar o topo do projeto, o Git permite fazer um “clone” e obter uma cópia completa do repositório. Fazendo com que cada desenvolvedor tenha todo o repositório em sua máquina local e possibilitando a sua substituição caso haja alguma perda de informação.

## 4. Manipulação básica do Git

### 4.1 Instalando

Para instalar o Git no terminal de trabalho, basta digitar no terminal do Sistema Operacional o comando abaixo:

```
# sudo apt-get install git-core
```

Nota: o comando acima deve ser aplicado para distribuições Debian/Linux. Caso possua outra distribuição instalada, acesse: <http://code.google.com/p/msysgit/downloads/list> para baixar e instalar a versão mais recente.

### 4.2 Iniciando o repositório

Nesse momento deve-se criar o diretório em que será iniciado o projeto, em seguida iniciar verdadeiramente o repositório Git no diretório.

```
# mkdir novo_projeto
# cd novo_projeto
# git init
```

### 4.3 Trabalhando no repositório Git

Sempre que for necessário adicionar um novo arquivo no repositório devem-se utilizar os seguintes comandos:

```
# touch novo_arquivo.sh
# git add novo_arquivo.sh
# git commit -m "novo_arquivo.sh adicionado no repositório"
```

O Git possui o que chamamos de “área intermediária”. Nesta área será permitido configurar o modo em que o commit vai ficar antes de enviá-lo ao repositório. Com isso você pode utilizar os comandos: add e rm que são responsáveis pela adição e remoção de arquivos e diretórios. Depois será apenas necessário utilizar o comando commit para enviar as alterações para o repositório local.

Nota: deve-se atentar bastante na utilização do comando rm. Se caso ele for utilizado sem o parâmetro -cached o arquivo será removido sumariamente do repositório.

### 4.4 Submetendo as modificações para o repositório

De acordo com a arquitetura distribuída do Git, ainda é necessário que as modificações realizadas no braço local sejam submetidas para o repositório remoto, onde outros desenvolvedores terão acesso. Para tanto se deve utilizar o seguinte comando:

```
# git push origin master
```

Onde, o parâmetro origin é o nome do repositório remoto e master é o nome do branch. Para adicionar o repositório remoto basta utilizar o comando:

```
# git remote add origin
usuario@meu_repositorio.com.br:/home/usuario/repositorio.git
```

## 4.5 Clonando um repositório remoto

Para obter um clone do repositório remoto deve-se utilizar o comando:

```
# git clone
usuario@meu_repositorio.com.br:/home/usuario/reposi
torio.git
```

## 5. Conclusão

Este artigo nos motiva a acreditar que, uma vez que tenha escolhido um determinado sistema de controle de versão, cuidadosamente estudado e preparado para seu projeto, este pode e deve ser de grande valia para o aumento da produtividade e melhorias no gerenciamento e controle dos arquivos. No entanto, é importante salientar que para haver uma escolha correta do sistema de controle que se encaixe perfeitamente ao contexto do problema, é necessário obter conhecimento das características técnicas, funcionalidades, desempenho, vantagens e desvantagens de cada ferramenta disponível, dessa forma sendo possível sinalizar quais dessas características são mais relevantes para suprir as necessidades.

Foi justamente isso que se tentou mostrar neste documento. A indicação e o emprego do Git, uma ferramenta que atende de forma eficiente e satisfatória os requisitos relacionados ao controle dos códigos fonte para sistemas distribuídos, sendo totalmente aplicável em grande parte dos projetos atuais. 

## Para mais informações:

### [1] Wikipédia (CVS):

<http://pt.wikipedia.org/wiki/Cvs>

### [2] Introdução a Controle de Versão Distribuído:

<http://betterexplained.com/articles/intro-to-distributed-version-control-illustrated/>

### [3] Controle de versões com o Git:

<http://blog.softa.com.br/2009/9/controle-de-vers-es-com-o-git>

### [4] L Glassy. Using version control to observe student software development processes.

Jornal of Computing Sciences in Colleges. Janeiro 2006

### [5] Entendendo o Conceito do Git:

<http://www.eecs.harvard.edu/~cduan/technical/git/>

### [6] Palestra – Linus Torvalds on Git. Tech Talk Google. Maio 2007.



**GLEUDSON JUNIOR** é Analista de Suporte, graduado em Redes de Computadores, Pós-graduando em Segurança de Redes e Sistemas, membro colaborador assíduo de comunidades Software Livre/Open Source, atua também como freelancer nas áreas de Serviços e Segurança de Redes e Sistemas desde 2005. Apaixonado pelo sistema do pinguim, por acreditar que a liberdade é pra todos. Site:

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

## Escola Linux

### A melhor opção em Treinamentos Hands-On

Eficiência e Praticidade em cursos de curta duração

[www.escolalinux.com.br](http://www.escolalinux.com.br) - Tel: (21) 2526-7262





# As novidades do Ubuntu 10.04

Por Wagner Emmanoel

Pouco maior que um gato doméstico, podendo pesar até 30 kg, tem cauda curta e orelhas bicudas, com um tufo de pelos nas pontas e é um mamífero de ordem carnívora. Bom, não estamos falando do animal em si, mas, da nova versão do Ubuntu, à 10.04 que recebe o codinome Lucid Lynx. A data prevista para sua chegada é 29/04/2010.

Assim, como nas versões anteriores, os desenvolvedores estão trabalhando pesado nessa distribuição, para mais uma vez trazer de forma centralizada e organizada o que há de melhor em software livre, como sempre fizeram os que atuam nesse projeto junto à Canonical. Porém essa é mais uma daquelas épocas difíceis, já que não se corre para lançar mais uma versão como as que temos de seis em seis meses, pois chegou a vez de outra LTS (Long Term Support) e toda correria é para tornar-lá uma versão estável e corrigir bugs

decorrentes das inovações introduzidas em releases anteriores. Se você não conhecem o ciclo de lançamento do Ubuntu, estará se perguntando o que é LTS? Calma! irei explicar, para os que já conhecem, serve para lembrar.

## Ciclo de lançamento do Ubuntu

As distribuições Ubuntu são lançadas semestralmente, tendo a primeira versão 4.10, lançamento em 20 de outubro de 2004, com codinome Warty Warthog. Fator interessante é que essa numeração (**4.10**) é para referenciar o ano de lançamento **2004**, com o seu respectivo mês outubro (**10**). O que enseja a existência desse ciclo de lançamento é para que os usuários tenham sempre o que há de mais novo em termos de softwares, porém para aqueles que não gostam dessa constan-

te mudança, vale lembrar que cada distribuição lançada é suportada por um período de dezoito meses, e para empresas onde o trabalho de implementação é mais doloroso. Depreende-se que dificilmente uma empresa por menor que seja tem apenas uma máquina, e quando a empresa é de médio porte acima, geralmente falamos em “parques computacionais”, o que torna difícil essa constante mudança, devido a fatores como: tempo, mão-de-obra e treinamento. Foi pensando nisso que a Canonical junto com a comunidade Ubuntu criou as versões LTS, essas que por sua vez, têm um suporte por longo tempo, sendo três anos para versões Desktops, já que sofrem muitas modificações durante esse período e cinco anos para versões Servers. A primeira versão LTS foi a 6.06, de codinome Dapper Drake. Como pode ser visto pelo número da versão, a primeira LTS saiu do forno com dois meses de atraso, já que os lançamentos deveriam ocorrer sempre em abril (04) e outubro (10), fazendo assim seu ciclo semestral desde o seu primeiro lançamento. Vejam o cronograma abaixo para melhor entendimento.

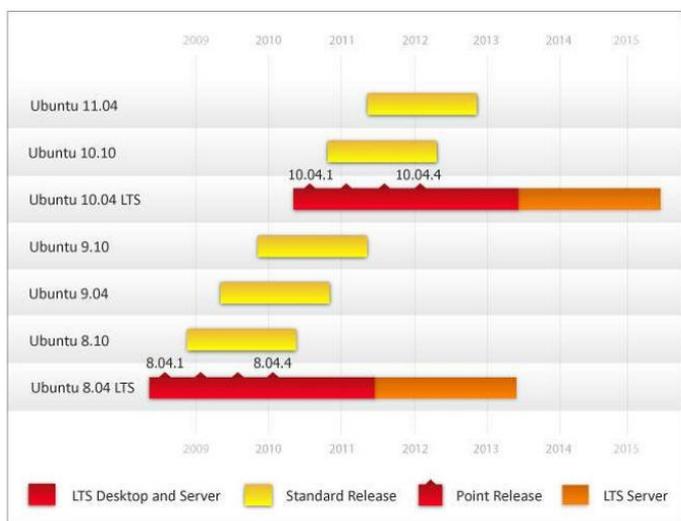


Figura 1 - Ciclo de evoluções

## Novidades da versão 10.04 Lucid Lynx

Bom, agora que já entendemos o ciclo de lançamento do Ubuntu, vejamos algumas novidades que o Ubuntu 10.04 Lucid Lynx trás.

## Kernel

O Ubuntu 10.04 virá com o kernel na versão 2.6.32.X, pois este já foi bastante testado e será ele que irá assegurar uma maior estabilidade, um melhor suporte a hardwares recentes e maior rapidez no dia a dia.

## Inicialização (boot)

Foi removido HAL (camada de abstração de hardware), tornando a inicialização mais rápida, além de facilitar a saída de uma suspensão. Essa experiência ao ligar o seu sistema e ver fluir de forma atrativa e rápida, tentando atingir os dez segundos de boot, em dispositivos equivalentes ao Dell Mini v10 é fantástico.

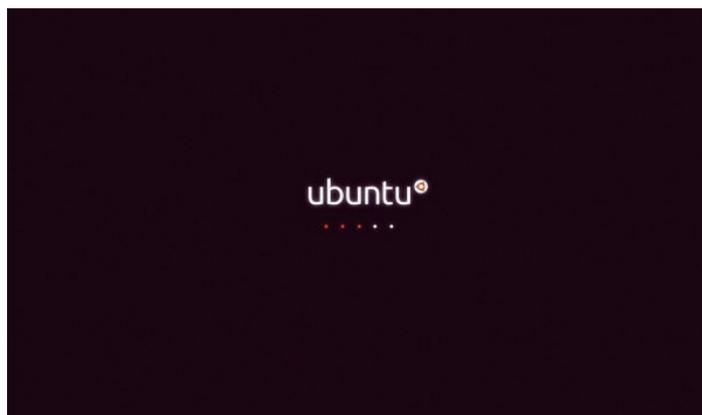


Figura 2 - Inicialização

Como pode ser visto, ocorreu uma grande mudança com sua identidade. Inspirando-se na ideia de luz, traz um visual totalmente inovador, que segundo a Canonical, esse tipo de software “light”, utiliza os recursos da máquina de maneira eficiente, além de executar rápido e ser de fácil manutenção quando necessário.

Para utilizadores de placas gráficas Intel, Nvidia e ATI, a sensação será ainda melhor. A única desvantagem desse processo é não poder piscar os olhos durante a inicialização, se fizer, não poderá ver o que aconteceu.

## Gnome

A versão 2.30 do ambiente gráfico Gnome trará melhoramentos gerais em todas as aplicações, incluindo o cliente de mensagens instantâ-

ne as Empathy, o gravador de disco Brasero, o leitor de documentos Evince (suporte a OCR, converter imagem em texto) e melhorias a nível visual nos ícones e no painel. Para os que utilizam o Kubuntu, a versão do KDE será 4.4.

## GTK

O novo tema GTK, apesar de não fugir tanto do esquema de cores da distribuição, sofreu algumas modificações e correções de bugs anteriores. O pacote de fantásticos ícones Humanity será melhorado (especialmente para se tentar obter um painel apenas com ícones no estilo Humanity acinzentado). Além disso, teremos ainda algumas modificações na janela de login, pa-

ra se tentar obter uma total de harmonia e estética em todos os componentes que formam o sistema. A modificação mais notada é, de longe, os botões de maximizar, minimizar e fechar. Agora, os botões estão do lado esquerdo, como no Mac OS X.

## Indicadores de sistema

No painel superior do Ubuntu, existem vários ícones que fornecem informações sobre o sistema, e até um que concentra informações e ações das aplicações de comunicação. Para o Lucid Lynx foi criado e unificado uma interface comum e consistente que permite agregar informações inteligentemente.

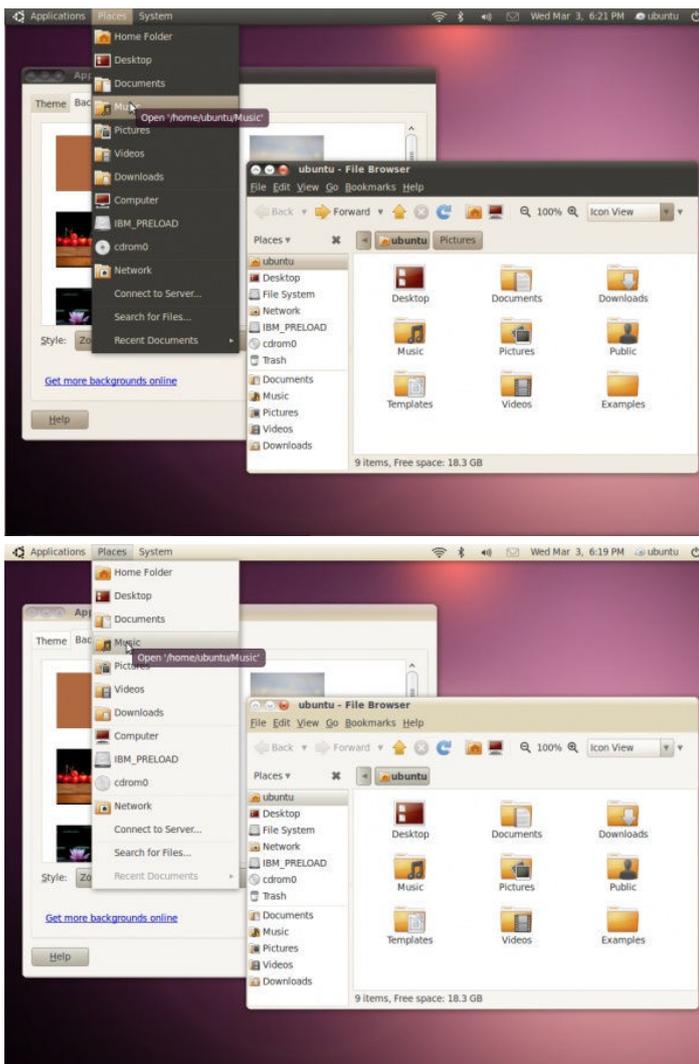


Figura 3 - Mudança na posição dos botões: uma das modificações mais notadas

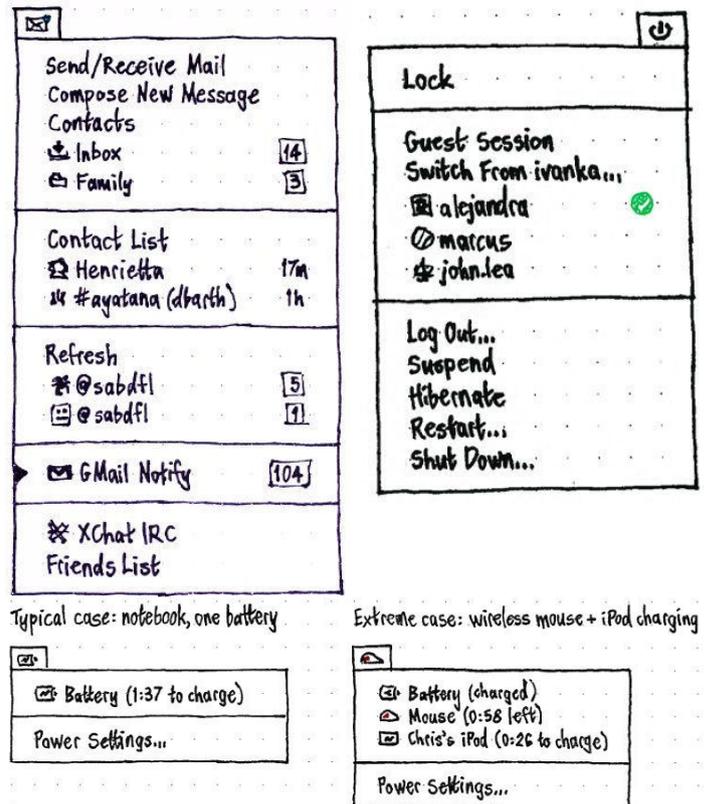


Figura 4 - Indicadores de sistema

## Melhorias nas notificações

Cada vez mais, as notificações do Ubuntu representam um dos melhores sistemas de notificação atualmente (quer visualmente quer em termos de facilidade de integração nas aplicações). Para o Lucid, as notificações estarão pre-

sentes com um novo modo, o modo “ocupado”. Por exemplo, se estiver vendo um filme em tela cheia, não será notificado de coisas triviais como por exemplo, e-mail, mensagens de chat, já avisos de bateria fraca ou de carga em bateria, esses sim, considerados “críticos” serão mostrados.

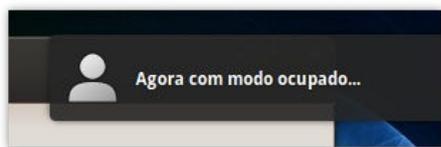


Figura 5 - Melhoria nas notificações

## Melhoramentos no Centro de Software

O Centro de Software Ubuntu caminha rapidamente para se tornar uma das soluções mais simples para instalação/remoção de programas, em qualquer dos 3 Sistemas Operacionais principais: Windows, outras variantes Linux e Mac OS.

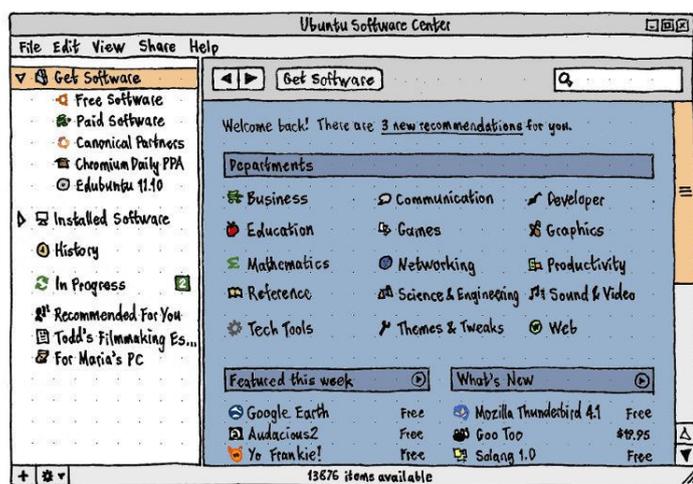


Figura 6 - Melhoramentos no Centro de Software

O Ubuntu 10.04, vai se tornar um centro onde pode instalar programas através de pacotes .deb de sites externos (substituindo o GDebi), onde pode adicionar/remover repositórios (Substituindo a aplicação Fontes de Aplicação) e onde poderá também atualizar o seu sistema.

## Ubuntu One Music Store

Usuários do Linux, sentiram falta por muito tempo de uma aplicação que lhe permita comprar música online, a partir do seu ambiente de trabalho, como o iTunes, por exemplo. A Ubuntu One Music Store, pretende resolver esse problema. Foi integrado ao player Rhythmbox, recursos que permitirão comprar músicas a partir do seu ambiente de trabalho e guardar no seu computador ou no storage do Ubuntu One. A loja é baseada na plataforma 7Digital e o Ubuntu servirá apenas como ligação entre o utilizador e o vendedor do conteúdo digital.

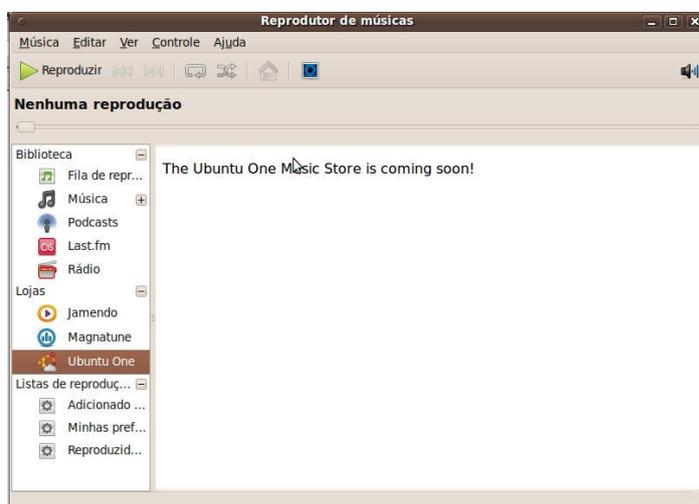


Figura 7 - Reprodutor de músicas

## Melhorias no F-Spot para edição de imagens simples

Serão adicionadas ao F-Spot funcionalidades básicas de corte, edição e retoque de imagem, porém existe a possibilidade de substituir esta aplicação por uma outra já com essas funcionalidades como o gThumb ou Shotwell. O utilizador comum quer apenas editar algumas fotos com retoques básicos, remoção de olhos vermelhos, cortar, um efeito de luz aqui e outro ali, e o GIMP revelava-se complexo demais para essa tarefa. Portanto, esperam-se novidades nesta área. O F-Spot é muito amigável e se os recursos citados acima forem implementados ele não deverá sair de cena.

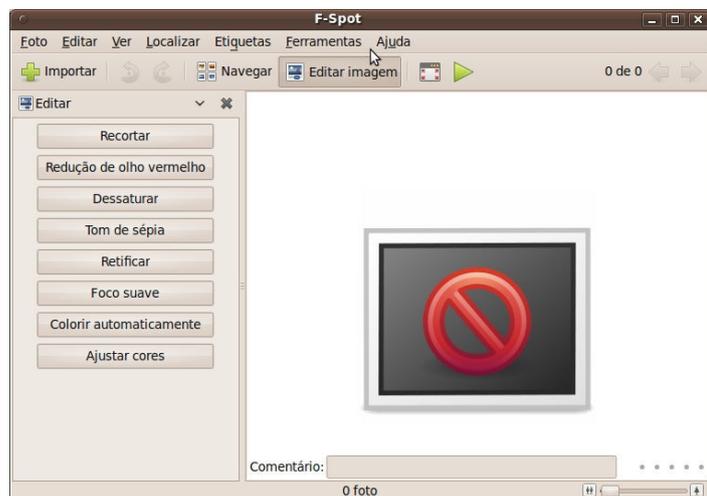


Figura 8 - F-Spot

## Gimp será substituído pelo Pitivi

O leque de aplicações que acompanham o cd também foi alvo de mudanças. O Gimp, considerado uma aplicação apenas para utilizadores profissionais e avançados, não virá instalado sendo substituído pela aplicação de edição de vídeo Pitivi, porém o Gimp continuar instalável a partir do Centro de Software. O Pitivi atualmente deixa a desejar algumas funcionalidades de um bom editor de vídeo, como o projeto OpenShot. A decisão ainda está polêmica e espera-se novidades nos próximos meses...

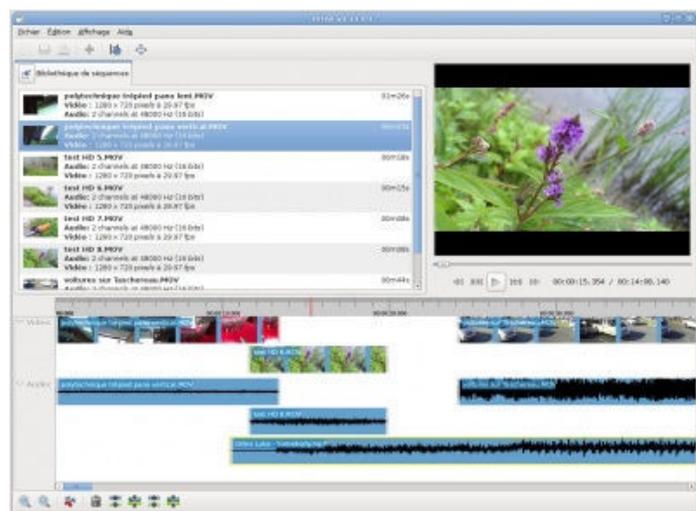


Figura 9 - PiTiVi

## Melhor seleção de jogos

Os jogos pré-instalados também vão ser repensados. A escolha vai recair em menos mas

melhores jogos. Atualmente o Ubuntu conta com variados jogos “inúteis”, que não são atualizados há muito tempo. Um dos jogos a ser incluído é o gbrainy, jogo de brainstorming e estimulação mental é bastante desafiador e viciante. Existem vários jogos de qualidade nos repositórios de Ubuntu utilize-os.

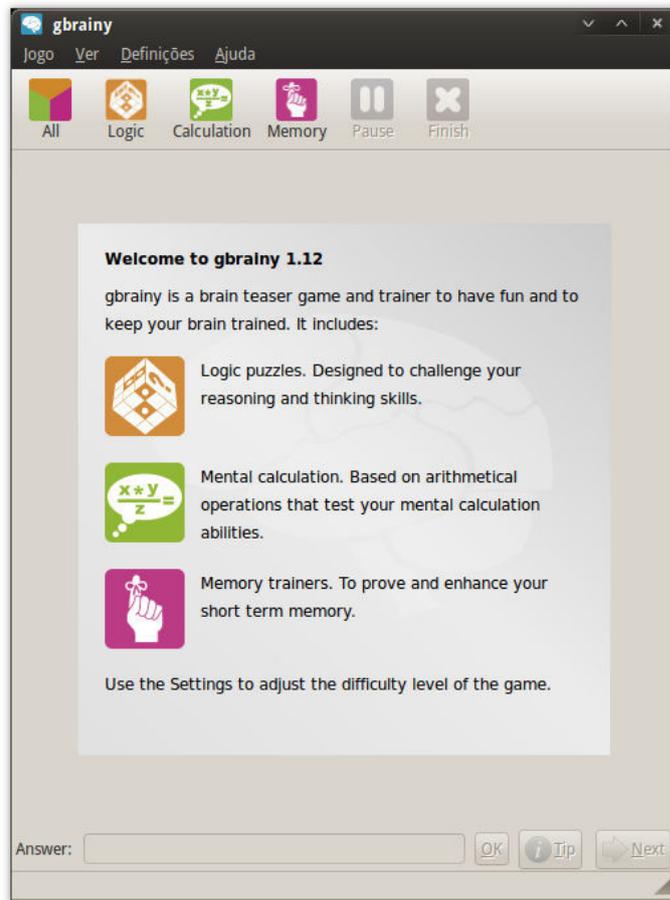


Figura 10 - Melhor seleção de jogos

## Nova ferramenta de Digitalização “Simple Scan”

O Ubuntu, embora muitos nunca tenham reparado, sempre trouxe consigo uma ferramenta de Digitalização, neste caso o XSane. O XSane é uma ferramenta poderosa e com elevado grau de compatibilidade, mas a sua integração no restante ambiente de trabalho e a sua interface em geral era tudo menos amigável. Por isso foi desenvolvida uma nova aplicação para o

substituir, de nome “Simple Scan”, uma interface simples de utilizar para facilmente digitalizar todo e qualquer tipo de documento em todo e qualquer tipo de impressora, e iniciando esse processo através de todo e qualquer tipo de aplicação. “Simple Scan”.

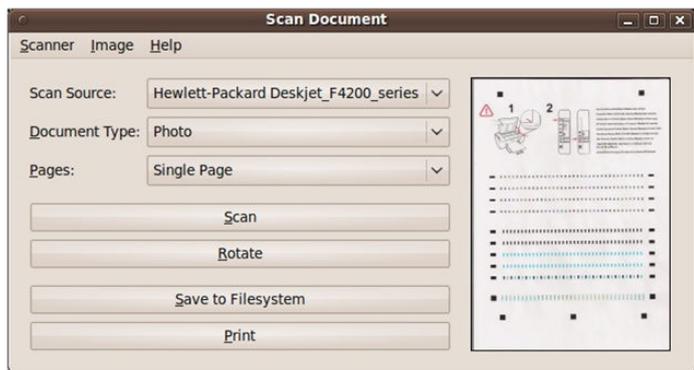


Figura 11 - Simple Scan facilita processo de digitalização

## Social por padrão (Gwibber)

Temos agora funções de integração com o Twitter, identi.ca, Facebook e outras redes sociais com o MeMenu no painel, que é construído sobre o projeto Gwibber, que tem um completamente novo backend, mais confiável construída em cima de desktopcouch. Gwibber agora também oferece suporte a multi-coluna de exibição para monitorar simultaneamente vários feeds.



Figura 12 - Gwibber entra em ação com redes sociais

## Novo padrão do driver de código aberto para hardware nVidia

O driver de vídeo Nouveau é agora o padrão para hardware da nVidia. Este driver ofere-

ce a configuração do modo kernel, o que dará uma melhor resolução de detecção. Este controlador fornece a funcionalidade de hardware acelerado em 2D, como o driver nv que substitui. O driver nouveau está sendo ativamente desenvolvido a montante e foi antecipado para permitir correções de bugs mais rápido.

## O suporte aprimorado para nVidia drivers gráficos proprietários

Três diferentes drivers proprietários da NVIDIA estão disponíveis atualmente: nvidia-current (190,53), nvidia-173, e nvidia-96. Graças a um sistema com novas alternativas, é possível instalar todos os três pacotes ao mesmo tempo (embora isso seja possível, só podemos ter configurado para uso um por vez).



Figura 13 - Melhor suporte a drivers da nVidia

## Novas funcionalidades para o Ubuntu Enterprise Cloud (UEC)

O instalador do Ubuntu Enterprise Cloud foi vastamente melhorada a fim de apoiar topologias de instalação alternativo. UEC componentes são automaticamente detectados e registrados, inclusive para topologias complexas. Finalmente, UEC agora é alimentado por Eucalyptus 1.6.2 codebase.

## Projeto “100 Papercuts”

O projeto “100 Papercuts”, pretende identificar e corrigir bugs mínimos de usabilidade no Ubuntu e nas suas aplicações. Este projeto já

iniciou-se no Karmic Koala, e continuará nesta nova versão do Ubuntu. Bugs do ciclo Karmic, integração e acesso fácil ao Compiz (Compiz é o decorador de responsável pelos tão falados efeitos 3D), Rhythmbox, Pitivi (ou a aplicação de vídeo que possa eventualmente substituí-la), Gwibber e Empathy serão alguns dos alvos deste projeto, e serão assim corrigidos alguns dos problemas mais proeminentes que afetam estas aplicações.

### Projeto B-Sides

O projeto B-Sides pretende facilitar a instalação de vários pacotes como: codecs, utilitários, fontes, temas, Flash, aplicações multimídia, comunicação e de produtividade, que não estão incluídos no CD do Ubuntu, mas que são igualmente úteis e essenciais, complementando assim o sistema. Bastará instalar o pacote 'b-sides' e todas essas aplicações serão instaladas.

### Inclusão de uma ferramenta de backup?

Esta é outra das aplicações que cada vez mais é essencial para o usuário. As propostas para ferramenta de cópia de segurança são o Déjà-Dup e Back in Time, ambos com suporte a backups automáticos regulares, backups seletivos para pasta/dispositivo externo/rede/Servidor online, e a restauração do sistema, baseado em um determinado backup. Estas são provavelmente as funcionalidades mais úteis e essenciais para a grande maioria dos usuários, uma ferramenta como esta é sempre bem-vinda.

### Nova versão do likewise-open

O pacote igualmente aberto, que fornece a autenticação do Active Directory e suporte do servidor para Linux, foi atualizado para a versão 5.4. O pacote de suporte a atualizações de ambas as versões suportadas oficialmente 4,0 (Ubuntu 8.04 LTS) e 4,1 (Ubuntu 9.10), bem co-

mo o likewise-open5 pacotes a partir do universo. Desde que esta atualização envolve uma série de alterações na configuração de arquivos e atualizações dos dados, testes e feedback é apreciado. 🇧🇷

### Para mais informações:

**Site Oficial Ubuntu:**

<http://www.ubuntu.com>

**Informações sobre o Alpha 3 do Ubuntu 10.04:**

<http://www.ubuntu.com/testing/lucid/alpha3>

**Artigo na Wikipédia sobre o Ubuntu:**

<http://pt.wikipedia.org/wiki/Ubuntu>

**Artigo no Kerodicas sobre o Ubuntu:**

<http://www.kerodicas.com/novidades/artigo=28837>

**Artigo no Guia do PC sobre o Ubuntu:**

<http://www.guiadopc.com.br/noticias/13634/canonical-renova-logotipo-e-visual-do-ubuntu.html>



**WAGNER EMMANOEL** é Instrutor da Fectura Tecnologia, Ubuntu Master South America



# O ZEN E A ARTE DE GERENCIAMENTO DE REDES

Por Igor Morgado

“Se você conhece o inimigo e conhece a si mesmo, não precisa temer o resultado de cem batalhas.”

- Sun Tzu

Estas palavras extraídas do livro “A arte da Guerra” de Sun Tzu, escrito quatro séculos A.C. já guiaram diversos generais em incontáveis batalhas. Nos dias atuais ele também é utilizado dentro do mercado corporativo ajudando executivos a tomarem decisões que definem o futuro de suas empresas. Este mesmo princípio pode ser utilizado no gerenciamento (e gestão) de redes de dados. Certamente a área de TIC<sup>1</sup> de diversas empresas hoje vive para apagar incêndios, um cenário típico em um ambiente que necessita da tecnologia para o seu negócio, mas muitas vezes ela foi colocada sem o total conhecimento do ambiente em que iria atuar, ou ainda uma tecnologia que foi inicialmente especificada para atender um cenário e no presente enfrenta algo muito diferente do que foi idealizado.

O gerenciamento de redes não tem como objetivo ser um elixir para solucionar todos os problemas. Ele simplesmente é a ponta do ice-

berg de um processo muito maior, mas que sem ele todas as mudanças subsequentes tendem a falhar<sup>2</sup> (para descrever todo processo seriam necessários muitos livros).

E qual a relação entre Sun Tzu e o gerenciamento de redes? O gerenciamento de redes fornece ao gestor (e obviamente sua equipe e toda empresa) dados indispensáveis do estado e funcionamento da rede, isso permite o “*conhecer a si mesmo*” (neste caso referindo-se a rede de infra-estrutura e serviços). Esse conhecimento irá conduzir suas decisões, presentes e futuras, e o bom uso dele irá transformar a “*equipe de incêndio*” em uma equipe alinhada com os objetivos da empresa.

O aumento do grau de complexidade das redes e também do seu tamanho exigem um sistema de gerenciamento que proporcione a melhoria na qualidade de serviço, gerenciamento pró-ativo e a integração da informática com os negócios.

### Formas de gerenciamento

Colocando de forma simples existem duas formas de gerenciar uma rede, e duas formas de atuar no gerenciamento de rede.

A forma de gerência mais comum encontrada em pequenos ambientes é a **gerência centralizada** onde um único gerente controla todo o processo. Neste modelo o gerente tende a ter conhecimento de tudo que acontece, mas os problemas deste modelo tornam-se cada vez mais críticos na mesma proporção do crescimento da rede. Outra forma é a **gerência descentralizada** onde as atividades são distribuídas entre os vários responsáveis, este modelo demanda uma forma de comunicação eficiente entre as diversas gerências que são responsáveis por uma determinada atividade. Um modelo não é necessaria-

“ Você não pode gerenciar o que você não mensura, e você não pode mensurar o que você não coleta. ”

Igor Morgado

mente melhor do que o outro. Em pequenos ambientes o modelo centralizado supri todas as necessidades, mas com o crescimento é desejado uma divisão de responsabilidades, onde cada gerência atua em um grupo de tarefas. Cabe ao gestor identificar quando é o momento para abandonar um modelo e seguir para outro, e como fazer isso sem impactar todo ambiente.

Em relação à forma de atuar, ela pode ser **reativa** onde os administradores são alertados dos problemas ocorridos e atuam na solução para encontrar a raiz do problema (*root cause*), ou **pró-ativa** onde os administradores acompanham a todo momento a saúde dos nós gerenciados de forma que a interrupção do serviço seja evitada. Obviamente a meta é resolver (ou evitar) sempre o problema antes de que ele aconteça, mas dispender equipes para monitorar tudo a todo tempo torna o processo caro e propenso a falhas. Um bom meio termo é definirmos quais serviços são críticos, e também definir o nível máximo de falhas aceitável, também conhecido como SLA<sup>3</sup>.

### O Modelo FCAPS

Manter o bom funcionamento de toda infraestrutura e serviços é um pré-requisito nos dias atuais em que tanto clientes como funcionários dependem da tecnologia para realização de todas suas tarefas. Por isso um modelo conceitual foi idealizado pela ISO<sup>4</sup>, que tem como objetivo criar diretrizes para o gerenciamento de rede. Seus pontos chaves são o gerenciamento dos

seguintes campos: Falhas (Fault), Configuração (Configuration), Contabilidade (Accountability), Desempenho (Performance) e Segurança (Security). As iniciais (em inglês) dão nome ao modelo FCAPS.

### Gerência de falhas

O principal objetivo da gerência de falhas é detectar, isolar, notificar, corrigir e anotar (log) todas as falhas que ocorram em uma rede. Adicionalmente estes dados podem ser utilizados para prever erros, desta forma a rede estará sempre disponível.

Quando uma falha ocorre, um agente monitorado irá enviar uma notificação para um gerente sobre o evento. Esta notificação pode disparar um alarme ou um outro evento, como um mail para o administrador, uma ferramenta de backup ou um sistema de alta disponibilidade.

### Gerência de configuração

Tem como função coletar e armazenar dados sobre hardware e software em todos os agentes monitorados, suas versões e configurações utilizadas (seja local ou remotamente). Também é responsável pelo agendamento de todas as paradas para substituição, atualização, modificação e eliminação dos sistemas existentes.

### Gerência de contabilidade

Contabilidade normalmente é chamada de gerenciamento de bilhetagem (em ambientes onde haja a cobrança pelo uso dos serviços disponibilizados). Esta gerência é responsável pelo re-

gistro do uso da rede por parte de seus usuários, seja pela cobrança ou pela regulamentação do uso.

Estas informações são úteis na distribuição dos recursos disponíveis, ajudando a minimizar o custo de operações, tornando o uso mais efetivo, também é responsável por assegurar um faturamento adequado.

Diversos tipos de estatísticas podem ser utilizadas como parâmetros para esta divisão/cobrança, entre os mais comuns estão: Uso de disco,

uso de link (carga ou tempo), ciclos de CPU.

### Gerência de desempenho

Gerência responsável pela medição e disponibilização das informações sobre aspectos de desempenho dos serviços de rede. Estes dados são usados para garantir que a rede opere em conformidade com a qualidade de serviço acordados com seus usuários. Também são usados para análise de tendência.

Ela permite a um gestor preparar a rede para o futuro, além de determinar a eficiência da rede atual. As métricas mais utilizadas são: carga de rede, percentual de uso, taxas de erro e tempos de resposta (latência).

Níveis de desempenho podem ser configurados para disparar um alarme. Os alarmes são tratados da mesma forma que os eventos gerados pela gerência de falhas, obviamente que eles variam de acordo com sua criticidade.

“O que deve ser entendido aqui é que os serviços chave da empresa devem estar rodando, sempre e bem.”

Igor Morgado

## Gerência de segurança

É responsável por controlar o acesso aos recursos de rede ou impedir o uso incorreto por parte de seus usuários, de forma intencional ou não. A segurança de dados pode ser obtida de muitas formas, como autenticação, criptografia e formas de acesso aos dados. O escopo da gerência de segurança ultrapassa a rede lógica e pode entrar na parte física como segurança pra acessar um servidor localmente, identificação por voz, ou até pitbulls bem treinados. Também é responsável por manter a confidencialidade das informações do usuário.

## Competências do FCAPS

Cada gerência do modelo FCAPS tem suas "obrigações", na tabela 1 podemos ver uma listagem de algumas das suas funções, esta não é uma lista completa nem mesmo restritiva, é uma simples recomendação.

Obviamente que em empresas de porte pequeno e médio, estas funções tendem a recair sobre uma mesma pessoa ou a um grupo pequeno de pessoas. O mais importante é ter a certeza que cada responsável é capaz de executar a

sua tarefa sem prejudicar as outras executadas.

## Sistemas de Gerenciamento de Redes

Controlar todas as informações disponibilizadas pelo modelo FCAPS não é uma tarefa simples e virtualmente impossível de ser atingida por seres humanos, é exatamente neste ponto em que entra o sistema de gerenciamento de redes (em inglês: NMS - Network Management System). O gerenciamento da rede de computadores é uma atividade indispensável para garantir a continuidade de serviço e também para manter um grau de qualidade de serviço elevado.

Cada empresa tem suas próprias necessidades e perspectivas com relação aos dados e informações a serem coletadas e monitoradas. Por isso o primeiro passo é saber exatamente qual o objetivo a ser atingido antes de iniciar um projeto deste tipo, isso fornecerá uma métrica para avaliar o sucesso do projeto.

## Elementos de gerência de redes

O sistema de gerencia de redes de um modo geral é composto por quatro elementos bási-

F	C	A	P	S
Detecção de falha	Organização de recursos	Uso de recursos	Taxas de erro	Acesso seletivo aos recursos
Correção de falhas	Distribuição de rede	Acesso à serviços	Taxas de uso	Registros de acesso
Isolamento de falhas	Identificação de rede automática	Limites da conta	Níveis de desempenho consistentes	Alertas de segurança
Tratamento de alarmes	Becape e recuperação	Combinar custos para múltiplos recursos	Coletar dados de performance	Relatórios de eventos
Geração de alarmes	Desativação de recursos	Estabelecer cotas	Analisar dados de performance	Privacidade dos dados
Filtragem de alarmes	Gerenciamento de inventário	Auditorias	Relatar problemas	Checkagem dos acessos dos usuários
Recuperação de rede	Gerenciamento de ativos	Relatórios de fraudes	Planejamento de capacidade	Cuidar de falhas de segurança
Correlacionamento de eventos	Cópia das configurações	Suporte a diferente tipos de contas	Gerar estatísticas com dados de performance	Realizar testes de segurança
Testes diagnósticos	Configuração remota		Manter e examinar registros de sistema antigos	Auditar registros de segurança
Armazenamento de erros	Execução e acompanhamento de jobs			Distribuição de dados de forma segura
Tratamento de erros	Distribuição automática de softwares.			Acesso seguro aos recursos disponibilizados
Estatísticas de erros				

Tabela 1: Competências do modelo FCAPS

cos: gerente, agente, MIB e protocolo de gerenciamento (a relação entre eles é representada na ilustração 1). Em gerência de redes o termo servidor e cliente não é muito utilizado, pois em alguns casos o agente atua como servidor de informação e em outras como cliente. Abaixo vemos a função de cada um destes elementos.

**Gerente:** É um computador responsável por coletar e armazenar todos os dados dos clientes, normalmente o gerente também disponibiliza as informações estatísticas e relatórios dos dados coletados. Também é conhecido por console de gerenciamento ou simplesmente console.

**Agente:** É um daemon (software) executado em um nó gerenciado. Normalmente o gerente conecta-se ao agente para coletar informações (**get**) ou mudar informações (**set**), deste modo o agente atua como servidor e o gerente como cliente, mas o inverso pode ocorrer quando algo no agente dispara um evento para o gerente (**trap**).

**MIB:** Banco de gerenciamento de informação (do inglês: Management Information Base) – é um tipo de banco de dados usado para o gerenciamento de dispositivos em uma rede. Ele contém uma coleção de objetos em um banco de dados (virtual) usado para gerenciar entidades em uma rede. Milhares tipos de dados podem estar definidos dentro de uma MIB, por exemplo: Uso de CPU, número de pacotes transferidos por uma interface ou o nome do servidor.

**Protocolo de gerenciamento:** Fornece os mecanismos de comunicação entre o gerente e

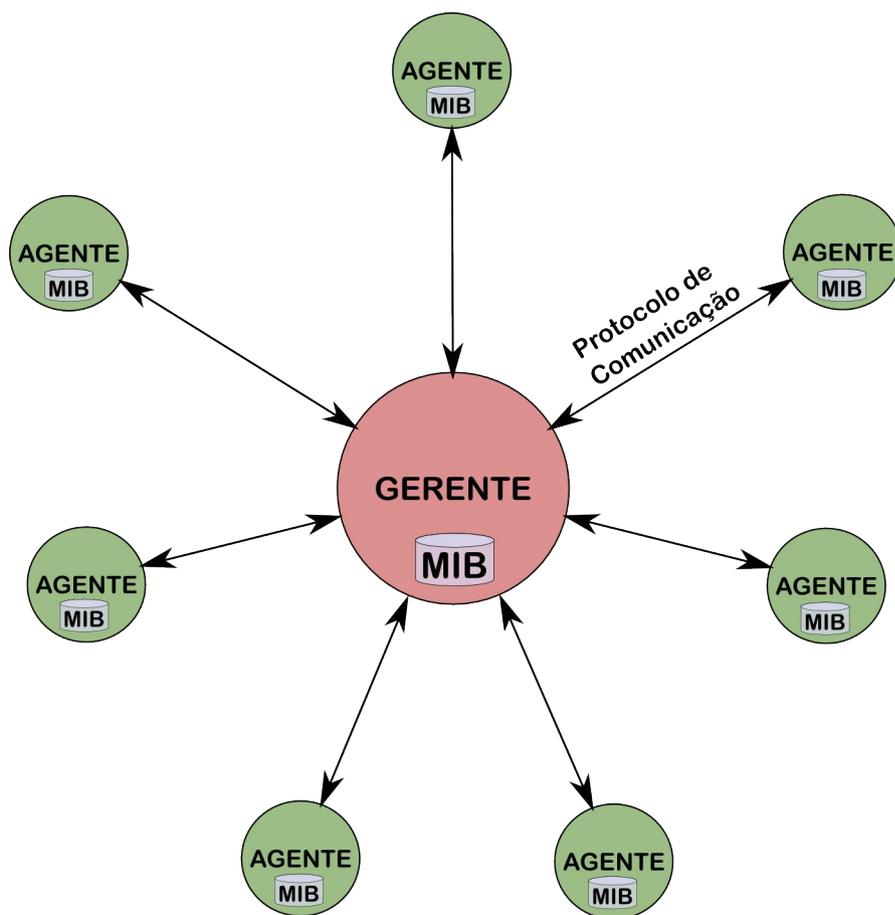


Ilustração 1: Elementos de um sistema de gerenciamento genérico

o agente, possibilitando a troca de informações independente da arquitetura adotada.

## Funcionalidades de um NMS

Um sistema de gerenciamento de rede executa tarefas simples as principais são:

**Levantamento:** Periodicamente analisa a rede para identificar novos nós a serem monitorados. Não é um elemento indispensável mas reduz bastante o tempo de implantação de um projeto.

**Coleta de dados:** Coleta os dados previamente indicados pelo administrador de rede, estes dados são armazenados em algum banco de dados. Estes dados podem ser um estado de um determinado serviço, como ativo/inativo ou um valor, como a quantidade de memória usada.

**Eventos:** Gera eventos relacionados com os dados coletados; ou correlaciona eventos múltiplos para gerar um evento mais complexo. Um evento simples poderia ser a parada de uma interface de rede identificando que o servidor está inalcançável, um mais complexo poderia ser a uma identificação de múltiplas interfaces de rede indisponíveis, indicando por exemplo que um roteador está com problemas.

**Alarmes:** Dispara alarmes quando determinados eventos são identificados. Esses alarmes podem também disparar ações corretivas como enviar um comando para um sistema de alta-disponibilidade ou desativar um dispositivo gerenciado.

**Notificação:** Notifica os administradores para que um diagnóstico seja elaborado e um plano de ação para solução do problema seja criado.

Outras tarefas podem ser encontradas em um NMS (ou integradas a ele), todas elas com o objetivo de suprir as competências descritas no FCAPS como: Inventário, distribuição de configuração, distribuição de software e jobs, gerenciamento de chamados e funcionalidades relacionadas.

### NOC/GIRS

Centro de operações de rede (do inglês: Network Operations Center). É um local onde fica centralizada toda gerência de uma rede, no Brasil foi cunhado um nome próprio que é Gerência Integrada de Redes e Serviços (GIRS). Através deste centro especialistas e softwares de gerenciamento de rede monitoram toda a intra-estrutura de uma empresa. Uma das grandes vantagens em ter um NOC é que o tempo de parada de serviços é drasticamente reduzido (da-

“ Uma das grandes vantagens de um NOC, é a redução drástica dos tempos de parada, tornando as empresas mais competitivas. ”

Igor Morgado

da a estrutura focada em monitorar pró-ativamente todo o ambiente), com isso as empresas se tornam mais competitivas.

De uma forma geral o processo de criação de uma GIRS (ou NOC) é dividido em 4 fases: centralização, consolidação, interconectividade e interoperabilidade.

**Centralização:** Nesta fase todos os administradores locais se tornam subordinados à um departamento único, com amplo controle sobre todas as ações relativas de operação de rede.

**Consolidação:** em um segundo momento, deve ser elaborada todas as diretrizes para os sistemas que irão auxiliar na gerência de todos os equipamentos de rede.

**Interconectividade:** na terceira fase, deve ser criada uma interface integrada para operação e manutenção de todos os equipamentos que utilizem uma mesma tecnologia. Utilizando interfaces padronizadas.

**Interoperabilidade:** o objetivo da última fase é a um sistema de operações que integre todos os dados da rede baseados no modelo FCAPS, através de um sistema e base de dados totalmente integrado.

### Considerações Finais

Uma abordagem de gerenciamento de redes através do modelo FCAPS dará a sua empresa informações suficientes para:

- Compreender quando um problema ocorre;
- Identificar os problemas de desempenho de TIC;
- Elaborar especificações mais precisas baseados no uso real do seu ambiente;
- Compreender quando mudanças de configuração ocorrem para que você possa atuar de forma correta;
- Identificar falhas de segurança quando elas ocorrerem.

Um segundo passo seria a criação de modos de visualizar estas informações de formas mais naturais como o número de negócios fechado através de um sistema de comércio eletrônico, ou o tempo de acesso de um visitante no seu web site. Este tipo de informação é muito mais importante para a alta gerência do que o tempo de resposta entre a central e os servidores da uma filial no Acre. O que importa neste nível é se a empresa está funcionando, esta informação justifica os investimentos feitos para a criação deste ambiente.

Tendo estas informações a empresa pode partir para um terceiro nível que é otimizar seus processos baseados no conhecimento que ela tem para que a TIC possa se alinhar cada vez mais com o negócio da empresa. O que deve ser

entendido aqui é que os serviços chave da empresa devem estar rodando, sempre e bem.

O FCAPS é uma visão completamente focada na tecnologia enquanto o modelo ITIL é focado nos processos e está na outra ponta da solução. Uma empresa não chega a lugar nenhum se *“não conhece a si mesma”*.

### De sua opinião sobre o artigo

Gostaria de saber o que você tem a dizer sobre este artigo. Envie-me um e-mail com a sua opinião e comentários para [igor-artigos@gnutech.info](mailto:igor-artigos@gnutech.info). 



**IGOR MORGADO** é administrador de sistemas e de redes, e instrutor há 13 anos. Atuou em grandes empresas como IBM, Sun e HP se especializando em gerenciamento de rede e segurança. Hoje é Diretor de Tecnologia da Gnutech e pratica Aikido sempre que possível.

### Para saber mais

1. Tecnologias de Informação e Comunicação  
[http://pt.wikipedia.org/wiki/Novas\\_tecnologias\\_de\\_informação\\_e\\_comunicação](http://pt.wikipedia.org/wiki/Novas_tecnologias_de_informação_e_comunicação)
2. From FCAPS to ITIL: An Optimized Migration, 21 de Fevereiro de 2005, por Suparno Biswas
3. Um SLA (acordo de nível de serviço, em inglês service level agreement) é uma parte do contrato de serviço onde um nível do serviço prestado é formalmente definido, normalmente relacionado à tempo ou desempenho. Fonte:  
[http://en.wikipedia.org/wiki/Service\\_level\\_agreement](http://en.wikipedia.org/wiki/Service_level_agreement)
4. International Standards Organization  
<http://www.iso.org>



# Knowledge is Power!

Por Francilvio Roberto Alff

Durante o Inverno de 2008 estive na cidade de Padova na Itália em um seminário sobre Gestão do Conhecimento, um dos palestrantes, Frank Meyer, CIO da companhia aérea Swissair falou um pouco sobre a importância da gestão do conhecimento e da inovação tecnológica, da qual pude extrair muitos benefícios, foi da li que iniciei a interessar-me um pouco mais sobre o maravilhoso mundo que está nos bastidores da Informática, naqueles dois dias, não imaginei que aquelas palavras de Meyer me levariam a apresentar um proje-

to incrivelmente vencedor dentro da minha empresa, foi então que realmente aprendi uma coisa já conhecida por muitos, ela se resume em uma frase: "Knowledge is Power!"

Mas afinal, do que estou falando?

A inovação tecnológica ainda é vista por muitas empresas como uma obrigação, uma espécie de preço a se pagar para continuar no mercado, e não como um instrumento para aumentar a produtividade, confiabilidade e qualidade de seus serviços e



...mas afinal de contas no que se resume o termo Knowledge Management? A resposta é simples: é a gestão de potencialidades do conhecimento de grupo, que alguns espertos preferem chamá-la de Patrimônio Humano Empresarial.

Francilvio Alff



produtos. Obviamente isso é em certa maneira aceitável uma volta que pensamos que a a Gestão Empresarial através de sistemas informativos é algo muito novo nos ambientes de Small Business. Logicamente a informática e todo o mar de possibilidades que ela carrega com si provavelmente não precisará de muito tempo para demonstrar seu potencial total.

Atualmente no setor informático de qualquer empresa, a quantidade de informações que escorem entre softwares, queries, análises e mãos humanas é sem nenhuma dúvida assustadora, haja ela cinquenta ou cinco mil colaboradores. A estrutura empresarial que hoje chamamos Information Techno-

logy tem uma importância significativa em todos os principais pontos de uma empresa em maneira direta ou indireta. E para gerir essa importante e massiva quantidade de informações que a T.I. detêm nas empresas precisamos de um sistema colaborativo organizado, dinâmico e funcional para manter o “trem” de talentos humanos sobre os “trilhos” da inovação, alguns já sabem bem sobre o que estou tentando falar. A sua filosofia colaborativa começou a ser desenvolvida a muitos anos, e de algum tempo para cá tem-se falado muito, vos apresento: Knowledge Management.

Quando falamos em Knowledge Management (abrevia-

ção: KM) muitos se assustam ou até mesmo subestimam a importância desse termo. O KM ou simplesmente chamado em português Gestão do Conhecimento infelizmente é ainda um horizonte longe de muitas empresas, mas afinal de contas no que se resume o termo Knowledge Management? A resposta é simples: é a gestão das potencialidades do conhecimento de grupo, que alguns espertos preferem chamá-la de Patrimônio Humano Empresarial.

Knowledge Management pode ser resumida em identificar, administrar e valorizar tudo aquilo que uma organização ou empresa sabe, ou poderia saber. Capacidade e experiências pessoais, arquivos, documentos, bibliotecas, a intimidade entre cliente e fornecedor e todas as informações presentes em data bases informáticos. É assim que Thomas H. Davenport define o Knowledge Management.

Para entender a importância do KM precisamos olhar um pouco para o nosso passado recente. Analisando as mutações empresariais do último século não é tão difícil prever o rumo que ele seguirá nos próximos anos.

Nos anos 20 a única preocupação de uma empresa era a produção em massa. Nos anos 50 o objetivo era produzir e vender, no final dessa década os primeiros sinais de valorização pessoal começaram a

serem vistos. Nos anos 70 foi quando começou uma grande revolução chamada Qualidade, foram nestes anos que as capacidades pessoais foram realmente valorizadas pela primeira vez. Nos anos 90 o cliente se transformou no alvo da empresa, e a principal arma para atingi-lo revelou-se ser o colaborador, essa última foi chamada também de Era da Competitividade ou Era do Marketing. Mas e agora, verso qual rumo estamos caminhando? A nossa era hoje pode ser chamada de Diversidade, onde não são as grandes empresas a conquistar mais clientes, mas sim empresas quase sempre pequenas e médias que oferecem produtos e serviços completamente dinâmicos e personalizados, atraindo assim não um grande número de clientes, porém clientes fiéis e totalmente satisfeitos.

“Como uma empresa pode expandir ao máximo a capacidade de seus colaboradores através de um projeto de Knowledge Management?” Foi essa a pergunta que me fizeram no último mês de novembro a Bolzen durante um encontro universitário sobre Gestão de Sistemas de Business Intelligence. Minha resposta foi simples: “Wiki! A maneira mais fácil para introduzir a Gestão do Conhecimento dentro de uma empresa é um sistema de Wiki.” Obviamente o Knowledge Management vai muito além de uma Wiki, mas a me-

 A World Wide Web teve, e tem ainda um papel fundamental na evolução do sentido e das aplicações da Gestão do Conhecimento, a rede com a sua liberdade de inovação possibilitou o nascimento de inúmeros grupos baseados em Open Source, onde o objetivo principal é o compartilhamento e a organização de informações... 

Francilvio Alff

lhor maneira para se iniciar é através de um portal comunitário como esse. Mas de onde vem toda essa certeza? Para explicar, deverei falar outra vez sobre internet.

A World Wide Web teve, e tem ainda um papel fundamental na evolução do sentido e das aplicações da Gestão do Conhecimento, a rede com a sua liberdade de inovação possibilitou o nascimento de inúmeros grupos baseados em Open Source, onde o objetivo principal é o compartilhamento e a organização de informações. Um exemplo notável e talvez o mai-

or de todos é a famosíssima enciclopédia online Wikipédia, que com um pouco de ambição e um mundo de colaboração se transformou em pouco tempo em uma espécie de divindade para os habitantes do mundo virtual, e não só. O funcionamento é simples e como base tem dois fundamentos primordiais do Open Source: toda informação pode ser modificada e melhorada por qualquer usuário e todos tem a liberdade para inserir novos artigos.

A Wikipédia hoje é um ponto de referência não somente para os curiosos de plantão,

“ Talvez o mais importante de tudo seja lembrar que qualquer sistema de KM tem como um dos princípios básicos a primeira lei do Open Source: a Colaboração. ”

Francilvio Alff

mas também para os administradores e gerentes de Information Technology, que através desse exemplo de sucesso buscam inspiração e um ponto de apoio para o desenvolvimento e implantação de sistemas similares voltados à grupos organizacionais como empresas ou entes públicos (entes públicos: leia-se escolas, universidade, centros de pesquisas e similares).

Acredito que todos, já tenham lido alguma página da Wikipédia, ou pelo menos tenham ouvido falar dela, já faz algum tempo que a fundação Wikipédia disponibiliza através da MediaWiki.org o software utilizado no site oficial gratuitamente, sob licença GPL2.

O que as empresas e os profissionais de recursos huma-

nos devem entender, é que a Gestão do Conhecimento não é sinônimo de gratuidade eterna, melhor dizendo, a sua implementação inicial requer um custo, que é ligada de forma muito íntima com a área de atuação e target comercial ou cultural de uma organização. Um bom sistema de KM, como um serviço de Wiki, custa relativamente pouco para ser projetado, mas requer algum tempo e muito envolvimento de todos os personagens de um cenário empresarial, no caso de uma Fabrica de Software, deve ser cultivada a intenção e a disponibilidade de colaboração partindo do analista financeiro, passando pelos programadores, testers e analistas de suporte e chegando até o diretor de marketing e às áreas administrativas, só assim podemos dizer que realmente existe um sistema de colaboração que

contribui à gestão do conhecimento.

Talvez o mais importante de tudo seja lembrar que qualquer sistema de KM tem como um dos princípios básicos a primeira lei do Open Source: a Colaboração. Não é preciso ser muito inovador para entender que em uma empresa que deseja uma estrada feita de sucesso vale mais um grupo sólido e colaborativo do que quatro ou cinco gênios com um enorme egocentrismo e total desconhecimento da palavra altruísmo. 🇧🇷



**FRANCILVIO ALFF** é duovizinhense, estudante de Arquitetura e Administração de Sistemas Informativos na Universidade de Verona/Itália. Profissionalmente é Analista de Riscos e Virtualização para empresa GlaxoSmithKline. Certificado como Cisco Certified Network Associate, Analista de Riscos na Virtualização VMWare e IT Admin pela EUCIP - European Certification of Informatics Professionals.



# Liberado Wordpress 3.0 Beta 1 para download

Por Rafael Cirolini

Um dos lançamentos mais esperados para a comunidade do WordPress neste ano tem sido a versão 3.0 do sistema. No dia 03 desta mês foi liberada a versão beta desta mesma versão, já recheada das novas funcionalidade que a tanto tempo esperavam os desenvolvedores que tinham que se virar com plugins ou temas que fizesse aquilo que eles precisavam. Teremos diversas novas funcionalidade, entre elas:

## **1 - O novo editor de menus**

Entre as novas funcionalidade talvez a mais aguardada.

Inicialmente desenvolvido pela empresa WooThemes, mas agora adaptado pela equipe de desenvolvimento para a construção do novo painel de menus. No construtor de menus será possível incluir paginas, categorias, e links externos tudo através de um editor gráfico. Além disto vai ser possível, criar os itens de forma que fiquem aninhados em submenus, com classes especificas, links de relacionamentos (XFN), descrição, entre outras funcionalidades.

## **2 - Fundo do site e Imagem de Cabeçalho customizável**

Para o usuário final vai ficar muito mais fácil controlar a aparência do seu site/blog com estas funções, ele pode além de trocar a cor, usar imagens que ele mesmo vai enviar do seu computador, e ainda no desenvolvimento do tema vai ser possível definir várias imagens padrão para que o usuário possa escolher entre elas.

### 3 - Um novo tema padrão

Com um layout simples e atraente, a maior ideia é que ele seja fácil de ser adaptado para as novas possibilidades de se editar o fundo e o cabeçalho. O tema é em duas colunas a segunda sendo o sidebar, o rodapé tem suporte a widgets, e ainda possui um menu em estilo dropdown. O "Twenty Ten" como é chamado, agora será o

único tema que vem com o WordPress.

### 4 - A morte do usuário Admin

Agora não é mais obrigatório que o usuário inicial do seu site seja o admin, já na instalação vc poderá escolher o usuário que deseja.

### 5 - Para desenvolveres

São muitas as promessas para desenvolvedores nesta versão, entre elas vão estar um novo mecanismo para criar posts customizados de uma forma muito mais simples, foi feita uma fusão entre o código do WordPress e o WordPress MU (multiuser), além de novas funções, melhorias em códigos, novos arquivos de template, e muito mais.

Fica o aguardo e o desejo que seja liberado logo a versão final. 🙏

### Para mais informações:

Site Oficial Wordpress  
<http://www.wordpress.com>

Site Oficial Comunidade Wordpress  
<http://www.wordpress.org>



**RAFAEL CIROLINI** trabalha com desenvolvimento de sites baseado no conceito do WordPress. Contribui no fórum da comunidade WordPress e desenvolve plugins para o projeto.

REVISTA  
**espírito**  
**livre**

LIBERDADE E  
INFORMAÇÃO

<http://revista.espiritolivre.org>



DIVULGAÇÃO

# CONHEÇA O BANCO DE DADOS FIREBIRD

Por Fernando Medeiros

Firebird é um SGBD SQL completo. É poderoso e leve ao mesmo tempo, com mínima necessidade de configuração e administração. Atende igualmente bem a aplicações de um único usuário e a aplicações corporativas. Um simples servidor Firebird pode manipular múltiplas bases de dados independentes, cada uma com múltiplas conexões clientes. E o melhor de tudo: é verdadeiramente Open Source, assim, livre de qualquer exigência de licenças, mesmo para uso comercial.

O Firebird foi derivado do

código fonte do Interbase Open Source em agosto de 2000. O Interbase não é um produto novo, está no mercado há mais de 20 anos e nesse tempo adquiriu respeito e admiração de muitos programadores e clientes (entre eles podemos citar a NASA, o exército americano, etc...).

Possui excelente performance e expressiva escalabilidade, indo desde modelos embarcados (mono-usuário) até sistemas empresariais com múltiplos bancos de mais de 500GB e centenas de conexões simultâneas.

Um dos recursos chave do Firebird é a arquitetura multi-geracional, que permite o desenvolvimento e suporte de sistemas híbridos OLTP e OLAP. Isso faz com que o Firebird seja capaz de servir simultaneamente dados analíticos e operacionais, pois, na maioria das situações, as leituras não bloqueiam a escrita quando acessando os mesmos dados.

Suporta o padrão ACID, triggers, stored procedures, UDF e eventos; possui suporte compreensivo ao padrão SQL92, além de inúmeras opções de conectividade. Alta compatibilidade com os padrões da indústria em diversas frentes tornam o Firebird uma escolha óbvia para o desenvolvimento de aplicações interoperáveis para ambientes homogêneos ou híbridos.

### **POR QUE USAR FIREBIRD ?**

A combinação de alta performance, baixo consumo de recursos, escalabilidade suprema, instalação simples e silenciosa, e distribuição 100% livre de royalties fazem do Firebird uma escolha atrativa para todos os tipos de desenvolvedores e fornecedores de software.

### **OPEN SOURCE**

Sua licença é baseada na Mozilla, sem restrições, sem li-

cença dupla ou limitações. Por isso, o Firebird é verdadeiramente Open Source.

### **MULTI-PLATAFORMA**

Suporta diversas plataformas de hardware e software: Linux, MacOS, HP-UX, AIX, Solaris, Windows entre outras. Funciona nas arquiteturas x386, x64, PowerPC, Sparc, etc. oferecendo migração fácil entre elas.

### **FERRAMENTAS**

Amigável para o desenvolvedor com várias ferramentas livres e comerciais: Java, PHP, Ruby, Python, C++, Delphi, Visual Studio, UIB, FIBPlus, IBOjects, etc...

O Firebird é utilizado por mais de 1 milhão de desenvolvedores em todo mundo e as comunidades mais ativas estão no Brasil, Alemanha, Rússia, Austrália, Itália, França, Turquia, Espanha entre outros. Aqui no Brasil, temos o evento FDD Firebird Developers Day, realizado anualmente com renomados palestrantes.

### **ESCALÁVEL**

São 4 arquiteturas para escalar de 1 Mb até 1 Tb, o limite do BD é de 32 Tb (banco real já testado com 1 Tb). Usuários de 1 a 1000 (escala de forma transparente). De DLL embutida até múltiplos cores/CPUs.

### **ARQUITETURAS:**

- Classic - Arquitetura para uso pesado em máquinas SMP com muita RAM e HDs rápidos.

- SuperServer - Leve, robusto, fácil de embutir e instalação silenciosa. Ideal para ISVs.

- SuperClassic - Combina o melhor do SuperServer e do Classic, ideal para ambientes virtualizados.

- Embedded (Mono-usuário, DBMS em DLL) - BD mono-usuário, dispensa instalação. Upgrade transparente para as outras arquiteturas.

### **VISÃO GERAL DOS CONCORRENTES**

#### **MySQL**

O MySQL não é gratuito para fornecedores de software independentes (ISVs),

"Para os distribuidores de aplicações comerciais, OEMs, ISVs e VARs: Os que combinam e distribuem comercialmente softwares licenciados com o MySQL e não desejam distribuir o código fonte do software comercial sob a licença GNU versão 2, deverão entrar em um acordo de licenciamento comercial com a SUN/ORACLE"

<http://mysql.com/about/legal/licensing/oem/>

### **MICROSOFT SQL**

O ISV precisa assinar um contrato de royalties para vender o SQL Server como parte da sua solução. Você e seu cliente terá que licenciar uma versão do Windows para servidores.

### **ORACLE**

Oracle é um ótimo banco de dados, mas é caro. Se você pode pagar a licença do Oracle e o custo de um administrador para manter seu BD saudável, bom para você. O ISV precisa assinar um contrato especial com a Oracle para poder embutir o BD em suas soluções. Muitas empresas usam o Firebird juntamente com o Oracle - onde a base de dados principal é Oracle e as subsidiárias ou remotas usam o Firebird.

### **INTERBASE**

O Interbase não é mais Open Source, é um software proprietário. Você precisa considerar seriamente as vantagens e desvantagens de um produto que não seja open source, que tem um concorrente de sucesso como o Firebird.

### **POSTGRESQL**

Entre outras bases de dados open source, o PostgreSQL é tão desenvolvido e maduro quanto o Firebird. É amplamente utilizado em muitas aplicações web e adotado por

várias grandes empresas. O PostgreSQL não possui uma versão embarcada, fazendo com que você tenha que usar um outro banco de dados para uma versão "light" da sua aplicação.

### **EXPRESS DATABASES**

Diversos BDs com versões "express" como MS SQL Express, Oracle Express, DB2, etc... Possuem limitações como tamanho máximo de 4Gb para o BD e usa somente uma CPU, upgrade não é transparente, necessitando de uma nova instalação da versão full.

### **EMPRESAS QUE ADOTA- RAM O FIREBIRD**

O Firebird é utilizado em todas as indústrias e aplicações (varejo, financeiro, farmacêutica, governamental, serviços, etc), veja algumas:

#### **CAIXA ECONÔMICA FEDER- RAL**

O SEFIP, programa distribuído pela Caixa Econômica Federal é utilizado em milhões de empresas no Brasil, trocou (na versão 8.3) o InterBase 6.0 pelo Firebird. Com isso, o número de servidores Firebird instalados no Brasil cresceu em alguns milhões.

#### **BAS-X**

Bas-X fica na Austrália,

sendo provedora líder de tecnologia de informação empresarial para varejistas independentes, operadores multi-site e grupos de gestão. Atualmente com uma base de dados de 450Gb.

### **WATERMARK TECHNOLOGI- ES**

Watermark Technologies fica no Reino Unido e atende empresas no ramo de finanças e setores governamentais, produzindo software que usa o firebird para gerenciamento de documentos, que inclui OCR indexados para pesquisa textual, utilizado por consultores financeiros, empresas de seguros e assim por diante. Atualmente, possui várias bases de dados com mais de 300Gb cada.

### **PROFITMED**

Profitmed fica na Rússia. Uma sociedade anônima que é um dos maiores distribuidores farmacêuticos do país. Possui banco de dados relativamente pequenos ( 60Gb, crescendo 2Gb/mês). Possui alto número de conexões simultâneas, atendendo a centenas de revendedores de pequeno porte e farmácias em toda a Rússia.

### **FIREBIRD NA INTERNET**

Muitos sites, blogs, listas de discussão e comunidades oferecem conteúdo gratuito para que você aprenda a utilizar o Fi-

rebird. São notícias, dicas, tutoriais passo a passo, cursos etc..

<http://firebase.com.br>  
<http://ibphoenix.com>  
<http://firebirdsql.org>  
<http://firebirdnews.org>  
<http://fernandomedeiros.com.br/blog/category/firebird>

## CONHEÇA A CAMPANHA MINDTHEBIRD

A campanha MINDTHEBIRD é a campanha mundial de divulgação do lançamento do Firebird 2.5. A ideia é colocar o Firebird sob os holofotes da mídia especializada. No site oficial da campanha há bastante material de apoio (banners, slides, documentos, logomarcas, etc). A contribuição dos membros e usuários do Firebird é de muita importância em

sites, blogs, lista de discussão e comunidades. A campanha ainda promove diversos concursos para escolha de logomarca, artigos, stored procedures, etc... Visite: <http://www.mindthebird.com>. 



**FERNANDO MEDEIROS** desenvolve softwares utilizando o banco de dados Firebird há 8 anos. Disponibiliza dicas, cursos e tutoriais passo a passo sobre programação e banco de dados Firebird em seu blog - <http://fernandomedeiros.com.br/blog/category/firebird>.



# Vem aí o ubuntu Guia do Iniciante



Por Carlos Eduardo do Val

O livro Ubuntu – Guia do Iniciante, terá como objetivo trazer informações e tutoriais básicos tanto para quem está chegando ao Ubuntu e ao mundo Linux, quanto para quem nunca usou um computador.

Fugindo dos jargões técnicos muito comuns nos livros do gênero, o livro Ubuntu – Guia do Iniciante trará os principais comandos, configuração do sistema para melhor desempenho em multimídia, tarefas de instalação e manutenção do sistema com uma linguagem simples e acessível.

Embora o Ubuntu seja a versão do Linux que mais atrai usuários de outras plataformas para o uso do Linux, seja pela gratuidade do sistema, pela segurança ou pela comodidade da instalação de seus pacotes, ainda é difícil para novos usuários encontrarem as informações básicas que precisam condensadas em uma única página da internet ou em uma obra escrita e é exatamente este público que o livro irá atender.

Entre instalação do sistema, instalação de pacotes multimídia e do Wine, o livro abordará a virtualização de sistema em uma linguagem prática. Tudo com muitas capturas de tela e explicações detalhadas.

O livro é a primeira obra do tipo no mercado nacional sobre a versão em lançamento do Ubuntu, que é construído sobre o Linux e softwa-

re livre, abordando como tema central o Ubuntu 10.04 Lucid Lynx, a ser lançado no dia 29 de Abril e estará disponível do blog Orgulho-Geek.net.

A Revista Espírito Livre apóia a iniciativa, além de ser parceira deste projeto.

O lançamento oficial da obra acontecerá no dia 29 de Abril de 2010. 

## Para mais informações:

Site oficial Orgulho Geek  
<http://www.orgulhogeek.net>



**CARLOS "KADU" EDUARDO DO VAL** é capixaba, tem 26 anos, é blogueiro e responsável pelo antigo "O Pirata Digital" (atual [OrgulhoGeek.net](http://OrgulhoGeek.net)), entusiasta do Ubuntu e vendedor de livros nas horas vagas.

# GIMP E A PROVA DE FOGO

Por Carlos Eduardo

Ola pessoal, nesta edição decidi fazer um artigo que foge ao usual, antes de faze-lo eu produzi a imagem final e postei num fórum muito sério que analisa este tipo de imagem para ver se minha manipulação passava no teste de fogo. A imagem final foi esta.



Figura 1



Notem que perto da geladeira existe um “fantasma”.



Figura 2

Para criar este efeito, eu utilizei 2 imagens.



Figura 3



Figura 4

Recortei a segunda imagem usando a ferramenta laço  o segredo é na propriedade do laço deixar opção suavizar e enevoar bordas habilitadas, assim nas bordas os pixels da segunda imagem se mesclarão com os pixels da primeira imagem.



Figura 5

O próximo passo é colocar a imagem 2 na imagem 1, com a imagem 2 devidamente recorta-

da aperte as teclas CTRL + C (menu Editar/Copiar), vá para a imagem 1 digite as teclas CTRL + V (menu Editar/Colar) e na aba camadas  clique no botão nova camada . Com a ferramenta mover  e redimensionar  ajuste a imagem até parecer estar perto da geladeira, lembrando que a proporção é muito importante, pois este tipo de geladeira normalmente fica na altura de nossos ombros.

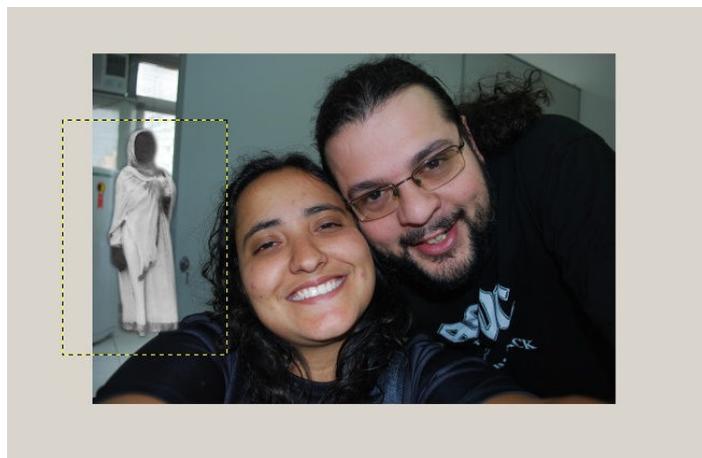


Figura 6

Agora temos que colocar nosso “fantasma” atrás da parede, para isto voltemos para nossa ferramenta laço  lembrando que devem estar habilitados as opções suavizar e enevoar bordas! E recortamos as partes do “fantasma” que estão na parede, ao final apertamos a tecla DELETE para excluí-los.

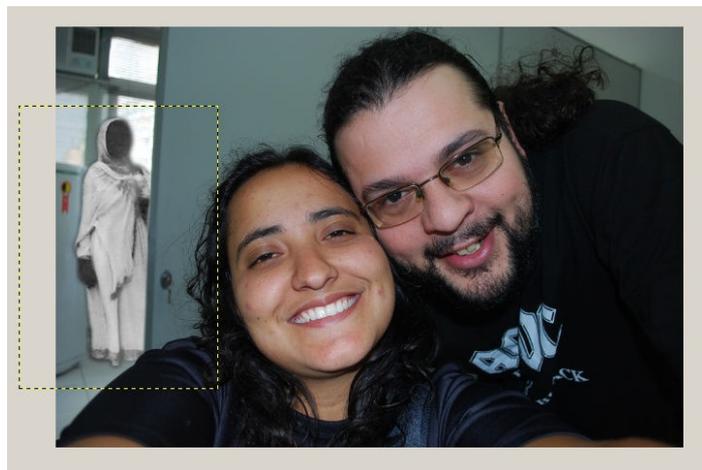


Figura 7

Agora começa a parte de detalhes que possibilitou dar a ilusão de realismo a foto, notem que a luz da imagem 2 não é a mesma da imagem 1 para corrigir isto vamos usar a ferramenta Colorizar a partir de amostra, com a camada da imagem 2 selecionada, vá ao menu Cores/Mapear/ Colorizar a partir de amostra, ajuste os seguintes parâmetros:

- No campo destino selecione a imagem 2.
- No campo amostra, selecione a imagem 1.
- Clique no botão ABTER AMOSTRA.
- O nível de saída tem de estar ajustado nos tons de cinza da amostra.
- Ao final dos ajustes clique em APLICAR e depois em FECHAR.



Figura 8

Na aba camada  da imagem 2 no campo modo selecione Mesclar Grãos e na opacidade coloque 18.



Figura 9

E pronto temos um fantasma!

O fórum onde postei se chama Portugal Paranormal o link é este <http://portugalparanormal.com/index.php/topic,5629.0.html>, para ver as imagens tem de estar logado, achei muito interessante como a técnica realmente funcionou bem, teve gente que colocou a foto no negativo e não foi identificado os pixels diferentes das duas imagens.



Figura 10

Teve também membros mais cautelosos que me pediu um imagem com EXIF Exchangeable image file format, é uma especificação seguida por fabricantes de câmeras digitais que gravam informações sobre as condições técnicas de captura da imagem junto ao arquivo da imagem que dificulta e muito forjar um imagem digital.

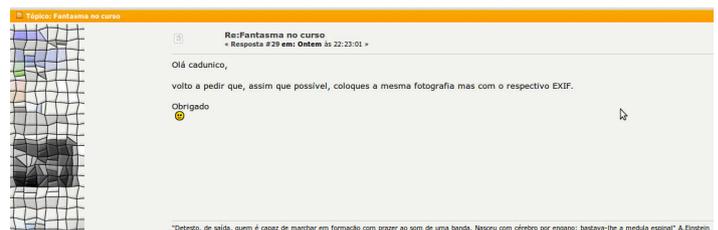


Figura 11

Meu intuito com este artigo foi somente mostrar que com o Gimp podemos fazer edições pesadas de imagens que podem até passar por verossímeis, não quero com isto denegrir a crença de qualquer pessoa dentro ou fora do fórum que a imagem foi postada, mesmo porque o escolhi por sua seriedade nas análises

deste tipo de imagem. Peço desculpas desde já se ofendi alguém durante a produção deste tutorial, mas meus intuítos foram estritamente acadêmicos.

E até a próxima edição pessoal! 

**Gostou deste tutorial?!  
Gostaria de sugerir algo?  
Tem habilidade com o GIMP,  
INSKCAPE ou outra  
ferramenta gráfica e de  
código aberto?  
Entre em contato. Quem sabe  
o próximo tutorial desta seção  
pode seção seja o seu!**

### Para mais informações:

**Site Oficial GIMP:**

<http://www.gimp.org>

**Forum onde a foto foi publicada:**

<http://portugalparanormal.com/index.php/topic,5629.0.html>



**CARLOS EDUARDO MATTOS DA CRUZ - CADUNICO** - atua a 18 anos como designer e a dois anos utilizo somente software livre em suas criações. É membro dos grupos LINUERJ, Debian RJ e SLRJ. Idealizador do GNUGRAF [<http://gnugraf.org>].



REVISTA

# espírito livre

LIBERDADE E  
INFORMAÇÃO

MANTENHA-SE  
INFORMADO!

<http://revista.espiritolivres.org>



## Como fiz uma edição de vídeo bem precisa com o Cinelerra

Por Antonio Misaka

Em 2009 tive uma experiência muito gratificante com o uso do software para vídeo Cinelerra. O projeto começou no ano de 2008, em março mais precisamente. Na época trabalhava em uma ONG aqui em Ottawa, Ontario, Canadá. Anteriormente, eu já havia gerenciado alguns projetos que ensinaram jovens de baixa renda a manejar uma vídeo-câmera e fazer a edição usando produtos da Apple como o iMovie e mais tarde usando o Movie Maker da Microsoft. Os projetos com os jovens foram bastante simples, mas possibilitou mostrar o potencial desses garotos e garotas com uma vídeo-câmera e algumas ideias. Como resultado foi possível criar e assistir a vídeos bem interessantes, considerando as limitações de recursos que tivemos à nossa disposição.

Voltando ao projeto de março de 2008, eu fazia parte do grupo de aprendizes para tocar Taiko (tambores japoneses) e eles, os mais graduados, estavam precisando registrar a apresentação que aconteceria no dia das Mães. Nessa altura, já tinham a confirmação da participação de um grupo vindo do Japão – Kamo Tsunamura Taiko, da província de Sendai. Com isso estariam fazendo um con-

certo com os dois grupos como celebração especial para o dia das Mães.

A diretora artística, Jack Steele, soube que eu poderia oferecer as vídeo-câmeras para fazer a gravação do show, na verdade dois shows, uma matine e uma durante à noite. Como onde eu trabalhava a organização possuía várias vídeo-câmeras, eu preparei todo o planejamento de gravação, comprei as fitas de vídeo e no dia combinado fomos lá para gravar o show.

Fizemos a gravação onde foram utilizadas 15 fitas de vídeos de uma hora cada, tendo 5 fitas para cada vídeo-câmera. Havia uma vídeo-câmera no fundo do teatro registrando o palco inteiro e duas vídeo-câmeras registrando os detalhes, estando uma na esquerda e a outra na direita do palco. Depois de tudo terminado entreguei as cópias digitalizadas que quase preencheram um disco rígido de 250 Gigabytes.

Com a cópia no disco rígido entregue, considerei isso como missão cumprida. Agora, eles iriam encontrar alguém com um estúdio de edição de vídeos e poderiam fazer o restante do projeto. Gerando como resultado um DVD para ser lançado no aniversário de 25 anos que aconteceu em março de 2009.

Mais ou menos em meados de setembro de 2008, a diretora artística me envia um email pedindo para encontrá-la e conversar a respeito dos vídeos que haviam sido gravados. Achei que havia ocorrido algum problema nas cópias geradas no disco rígido e me preparei para levar as fitas originais.

Na reunião que tive com ela, fui de uma certa forma pego de surpresa. Ela me disse: “Já que você fez a gravação, não daria para fazer a edição desses vídeos?” Como

“Fazer estes ajustes finos foi o que mais me impressionou (...) Ela permite fazer isso ajustando frações de segundo.”

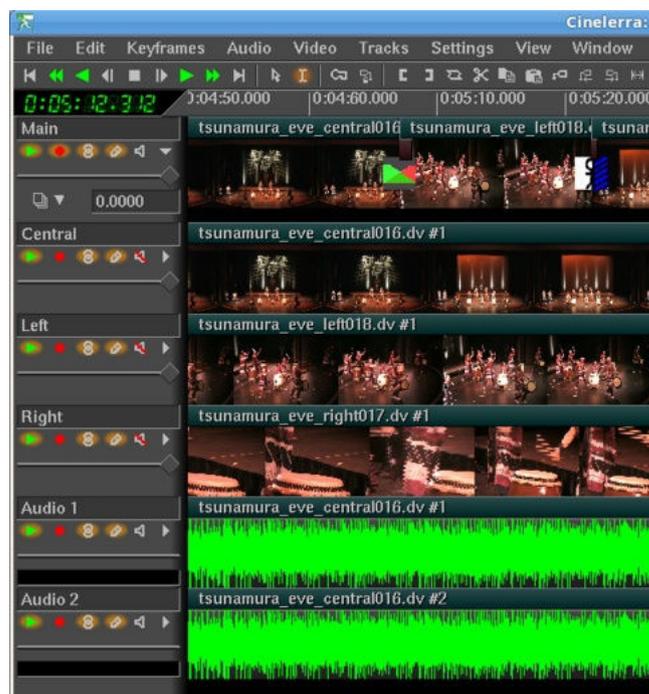


Figura 1: Detalhes das trilhas no Cinelerra

tudo foi feito na base da amizade e voluntariado, eu quase respondi: “Jack a minha parte já está encerrada, acho que seria interessante conseguir outra equipe para finalizar a edição.” No entanto, eu sabia que essa era a parte mais trabalhosa de todas e também concluí que se ela não encontrasse alguém para ajudar na edição aconteceriam uma das duas coisas, ou o vídeo finalizado não sairia mais, ou se saísse haveria a chance de cobrarem bastante pelo serviço de edição, o que de qualquer forma poderia inviabilizar o projeto de venda do DVD como arrecadação de fundos.

Com o coração mole e a chance de testar um produto que poderia me trazer alguma experiência na criação de vídeos mais complexos, aceitei a proposta, considerando que ela disse que isso poderia ser finalizado em março de 2009. Uma janela de praticamente seis meses, o que para mim parece ser um bom tempo para desenvolver o projeto nas horas vagas.

Comecei os testes com o Kino e acabei não tendo muito sucesso com o resultado da edição de vídeo. Não encontrava muita informação a respeito da utilização dele de como fazer a sincronização de três vídeos e selecionar as melhores cenas do show. Decidi com isso abandonar a ideia de usar o Kino como uma ferramenta de edição.

Fui então atrás do Cinelerra e comecei a preparar a

instalação no meu laptop, um Toshiba A100, com processador dualcore, 1 Gi-gabyte de memória e 80 GB de disco – rodando Linux Ubuntu 8.04 na época. Fiz alguns testes com alguns pedaços de vídeos e depois de tudo funcionando, comecei a acessar os vídeos no disco rígido externo que tinha as cópias da gravação. Depois que entendi como fazer a edição, a única coisa que tomava tempo era o processo de renderização, mas era possível deixar renderizando e ir editando outra música. Cheguei a ter momentos em que renderizava duas músicas enquanto editava uma terceira.

Neste momento começou a minha jornada, estava em meados de outubro de 2008 e sem a mínima noção de como começar o processo de sincronização. Parecia que tinha entrado em uma máquina com um monte de janelas, botões, trilhas de vídeo, trilhas de som, opções de formatos, onde as únicas coisas que estava reconhecendo eram os botões de iniciar e parar a exibição do vídeo. Achei que estava em uma barca furada e no início de um pesadelo me vendo com todos os meus colegas em olhando e se perguntando, e esse vídeo será que sai ou não sai?

Bem decidi parar tudo e começar a fazer algumas pesquisas no Google para descobrir o que havia sobre o Cinelerra. Consegui achar alguns tutoriais no formato pdf. Fiz a impressão desse material e comecei a ler. Que leitura terrível, eu tinha que ler e entender o que os autores descreviam sobre cada elemento da fabulosa ferramenta. Pensava com os meus botões, fabulosa para eles, para mim parece mais uma mistura de grego com latin e um pouco de figuras. Talvez se a minha formação fosse para a área de cinema e vídeo, muitas das terminologias seriam fáceis de entender. Infelizmente não é a minha área de formação, então fazer o que?

Dezembro chegou e não tinha avançado nenhum quadro de vídeo editado. Felizmente, o pessoal não me cobrava para saber como andava o projeto. Continuei fazendo as minhas buscas e então descobri alguns tutoriais em



*Ensaio do grupo de taiko*

vídeo que me deram alguma noção melhor de como usá-lo para o processo de edição. Os autores neste ponto foram bem didáticos, explicando passo-a-passo o que poderia ser feito e demonstrando no próprio vídeo que eles criaram usando o Cinelerra.

Comecei aos poucos a entender o processo de edição com a ferramenta e comecei a aplicar no projeto. Agora sim, as coisas pareciam fazer sentido, os ícones, as trilhas de áudio e vídeo, os recursos de zoom, os botões de desligar e ligar as trilhas, etc. O site onde que consegui os tutoriais é chamado “the source show” <http://www.thesourceshow.org/node/11>. O episódio 3 – Revenge of Cinelerra mostra alguns dos recursos de edição possíveis com a ferramenta. Também é possível ver que ele, Aaron Newcomb, disponibiliza os vídeos para baixar em pelo menos três formatos diferentes. Todos gerados pelo Cinelerra.

Em meados janeiro deste ano, quando dominei o processo de edição tudo começou a fluir de forma conjugada, as diferentes tomadas de cada vídeo-camera, a seleção e edição do som. Neste momento, comecei a perceber como é produtivo o uso do Cinelerra. Tudo muito fácil e rápido para selecionar, recortar e colar. Consegui finalmente gerar o meu primeiro vídeo de uma das músicas do grupo local (Oto-Wa Taiko), selecionando as melhores to-



O compositor do Cinelerra

mas colocando os efeitos de transição existentes no Cinelerra. A Figura 1 mostra um detalhe da edição com 4 trilhas de vídeo e duas de áudio. As trilhas ativas são apenas as de áudio e a primeira de vídeo que contém os trechos que interessam para renderizar. As outras três são para acertar a sincronização e seleção das melhores tomadas.

Existe um dos vídeos gerados que está postado no Youtube (<http://www.youtube.com/watch?v=pvwd8wV84U4>), este ainda não havia sido comentado pela diretora artística porque ela estava no Japão e foi a forma de enviar a ela o resultado da edição de vídeo. Neste vídeo é possível ver as três vídeo-câmeras já sincronizadas na combinação do vídeo final.

No começo de fevereiro, gravei o primeiro DVD com todas as músicas do grupo de taiko local. Encaminhei para a diretora artística e aguardei os comentários dela. Achei que tinha sido o máximo com o resultado obtido. Então comecei a trabalhar com o segundo grupo (Kamo Tsunamura Taiko), que veio do Japão. Passada algumas semanas recebi a ligação de Jack pedindo para fazer uma reunião a respeito do primeiro conjunto de vídeos.

Ela elogiou o que vi

e disse que gostaria de fazer algumas sugestões para melhorar. No entanto, ela me disse que haviam algumas músicas que estavam com problema de sincronismo entre a batida que se via no vídeo e o som que

se ouvia. Eu não acreditei que ela conseguia ver isso. Eu olhava e não conseguia ver e ouvir, achava que era perfeccionismo da mulher. Depois de assistir algumas vezes com ela me mostrando onde acontecia isso, concordei que tinha que fazer os ajustes.

Fazer esses ajustes finos foi o que mais me impressionou na ferramenta. Ela permite fazer isso ajustando frações de segundo e isso acontece por trilha. Foi a coisa mais incrível que vi no processo de edição, bom isso para mim que nunca tinha trabalhado com essa complexidade de sincronismo. Ampliar, selecionar, recortar e colar, era tudo o que eu tinha que fazer. Também havia possibilidade de fazer diferentes versões do vídeo para que ela escolhesse o melhor. Tinha música que em alguns momentos tinha quinze ou dezoito trilhas, sendo um terço de vídeos e dois terços de áudio (estéreo) no processo de edição.

No começo de março finalizei todos os vídeos dos dois grupos e preparei para que fossem empacotados em DVD para entrar em produção no final de março. Neste processo, uma outra pessoa com conhecimento de design gráfico nos ajudou usando o Macbook, gerando o menu de seleção e a apresentação com animação gráfica. Minha expectativa é que fosse feito com um produto de software livre, mas o pessoal aqui que trabalha na área gráfica é toda fã da Apple. 🍏



**ANTONIO MISAKA** é Mestre em Ciência da Computação pelo ICMC-SC – Instituto de Ciências Matemáticas e de Computação de S. Carlos, USP, tornando-se especialista em requisitos de software. Atualmente está morando na cidade de Ottawa, Ontario, Canadá e trabalhando na SUMNet Innovation como Vice-presidente de projetos e infraestrutura.

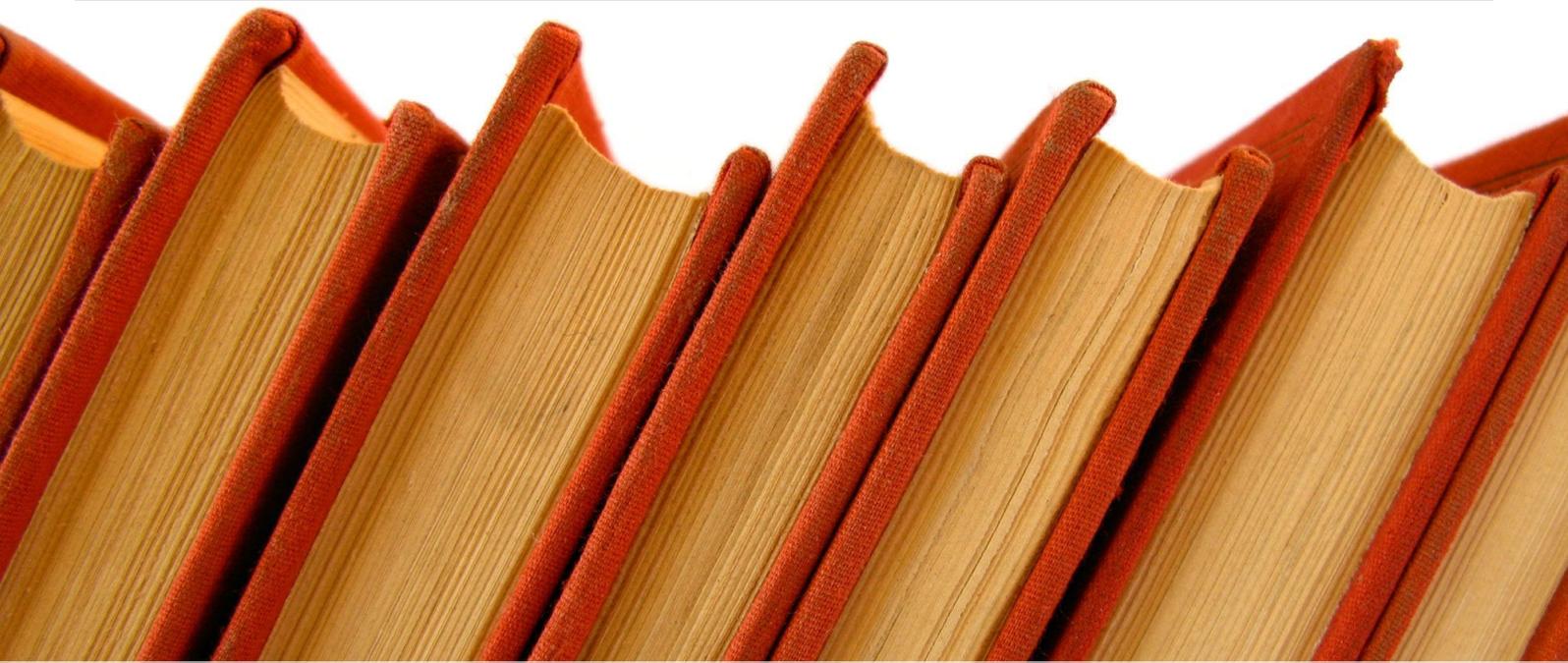
### Saiba mais

Site do Cinelerra - <http://cvs.cinelerra.org/>

Site do Lumiera - <http://lumiera.org>

Vídeos tutoriais da The Source -  
<http://www.thesourceshow.org/>

Site da SUMNet Innovation - <http://www.summet.ca/>



# SOFTWARE LIVRE EM TRABALHO DE CONCLUSÃO DE CURSO

Por Jandson Damasceno Rabelo

Zsuzsanna Kilian - 2010

As empresas hoje em dia vêm cada vez mais aumentando o número de softwares livres nos seus parques tecnológicos, sejam por questões econômicas, como redução de custos, ou por perceberem que o software livre cumpre o mesmo papel que o software proprietário, sendo muitas vezes mais eficiente que o proprietário. Mais o que não vemos com frequência são empresas, principalmente pequenas e médias, formadas somente com software livre, desde o servidor à estação de trabalho, participando na execu-

ção de projetos/ produtos, na qual o consumidor não saberia diferenciar ou sequer pensar que o mesmo foi feito com programas livres, em todas as etapas de sua produção.

Sou o leitor Jandson Damasceno Rabelo, cursando o 7º período do curso de Sistemas de Informação, após esta breve introdução, venho informar que estou desenvolvendo o meu Trabalho de Conclusão de Curso (TCC), sobre como criar uma empresa de design e publicidade, 100% livre, na qual a ideia é mostrar como é possível criar uma empresa de

pequeno e médio porte, num ramo onde o serviço / produto prestado e visível a todos, mostrando que software livre é sinônimo de qualidade, além de substituir todos os softwares pagos, desde o administrativo ao do setor de produção, onde na grande maioria das empresas deste ramo predomina soluções da empresa Adobe.

O objetivo desta matéria, além de apresentar um fato bastante interessante, e pedir a colaboração da comunidade de usuários de softwares livre na elaboração do meu projeto de TCC, na coleta de material, ideias, sugestões e tudo mais que quiserem colaborar, já venho desenvolvendo este trabalho há algum tempo, porém encontro algumas dificuldades que es-

pero que alguém possa vir me ajudar, gostaria de mostrar, a diferença no uso do processamento e memória na produção de um mesmo trabalho, em diferentes sistemas operacionais, sendo no mesmo padrão de configuração de equipamento para todos, por exemplo, o uso do Blender, em sistemas como Ubuntu e Debian, o do Studio Max 3D no Windows e computadores Mac, porem com configurações similares, mostrando a diferença que cada um gasta de recursos de suas máquinas, para fazer o mesmo trabalho, como não possuo computadores Mac, fazer tal comparação, torna-se inviável.

Os interessados em ajudar, podem mandar suas opiniões, pontos de vista, críticas

entre outros para o meu email [jandson.dr@gmail.com](mailto:jandson.dr@gmail.com). Espero a colaboração e cooperação de vocês, desde já obrigado. 🙏



**JANDSON DAMASCENO RABELO** é graduando em Sistemas de Informação pela UNESC - ES, certificação Linux Professional Institute Certified Level 1 - LPIC-1, Novell Certified Linux Administrator (Novell CLA). E-mail: [jandson.dr@gmail.com](mailto:jandson.dr@gmail.com)





Relato do evento

# Document Freedom Day

31 de Março de 2010

Por João Fernando Costa Júnior

A Iniciativa Espírito Livre, mais uma vez, esteve a frente da organização de mais um DFD. O Dia da Liberdade dos Documentos (DFD, Document Freedom Day) é um dia internacional para Libertação de Documentos com ações de base no sentido de promover Formatos de Documento Livres e Padrões Abertos em geral.

No estado do Espírito Santo o evento aconteceu no dia 31 de março, no auditório da FAESA -



Figura 1: Participantes durante o evento



Figura 2: O palestrante João Fernando, Albino Biasutti da comunidade Tux-ES e o palestrante Guilherme Chaves

Campus 1 em Vitória/ES. O Document Freedom Day aconteceu simultaneamente em diversos países ao redor do mundo, incluindo Vietnã, Alemanha, Eslovênia, África do Sul, Grécia, Japão, Áustria, e outros.

Na grade do evento houveram 4 palestras que prenderam a atenção dos mais 80 inscritos. O evento teve início às 18 horas com a entrega dos alimentos pelos participantes. A primeira palestra teve início às 19:00 hs, com o tema "Soluções Livres", proferida por Guilherme Chaves. O palestrante se ateve a apresentar de forma simples e clara as facilidades que os softwares livres proporcionam a quem usa, além de abordar ainda tópicos



Figura 3: Palestra com João Fernando, sobre o case da Revista Espírito Livre



Figura 4: Eliane Dobrawolske em sua palestra sobre o case da Refrigerantes Coroa

como pirataria e licenças. Às 19:40 hs teve início a palestra "Um case de sucesso: O desafio da Refrigerantes Coroa na adoção do BrOffice.org", apresentada por Elaine Dobrawolske, gerente do departamento de TI da referida empresa. Elaine expôs aos presentes o processo que a empresa passou desde a decisão de utilizar a suite BrOffice.org até o presente momento. Neste caminho, foram narrados os problemas encontrados durante a trajetória de migração, treinamento, conscientização, etc. A palestra "Construindo uma publicação com software livre e formatos abertos: Um case da Revista Espírito Livre" teve início às 20:20 hs, apresentada por João Fernando Costa Júnior, editor responsável e idealizador da publicação. Foi exposto o processo de criação e diagramação da revista, apresentado as ferramentas em software livre disponíveis, além dos formatos de arquivos também utilizados. Ao final, o palestrante mostrou aos presentes editou alguns arquivos exemplo para que os participantes vissem como o edição é feita. Um detalhe que merece ser lembrado é o fato da maioria dos presentes ser formada por leitores da revista, fato que deixou o palestrante bastante satisfeito. Às 21 horas foi proferida a última palestra do evento, "Sistema de Gerenciamento de Atendimento Livre: acabe com as filas", genuinamente apresentada por Adilson Oliveira Cruz, da Dataprev.



Figura 5: Adilson Oliveira, da Dataprev, em sua palestra sobre Gerenciamento de Atendimento Livre

A realização do evento contou ainda com a colaboração da comunidade TUX-ES.

A galeria de fotos do evento poder ser vista [aqui](#).

Com o término das palestras foi feito um balanço do evento e o saldo foi mais que positivo. Foram arrecadados 50 kg de alimentos que posteriormente serão doados para o Projeto Amor e Vida, do bairro Jardim América, Cariacica/ES. A entrega dos alimentos aconteceu no dia 19 de Abril.

Agradecimentos a todos que tornaram o evento possível, em especial ao pessoal da FAESA - Campus 1, que gentilmente cedeu suas instalações para a realização do evento, aos professores que direcionaram seus alunos para as palestras e a toda a comunidade que marcou presença em peso. 🇧🇷

### Para mais informações:

Site oficial Document Freedom Day

<http://www.documentfreedom.org>

Site do DFD em Vitória/ES

<http://dfd.espiritolivres.org>

Grupo do DFD no Identi.ca

<http://identi.ca/group/dfd>



**JOÃO FERNANDO COSTA JÚNIOR** é líder do GUBRO-ES, responsável pela Iniciativa Espírito Livre / Revista Espírito Livre, moderador da lista PSL-ES, e contato regional do Partido Pirata do Brasil.





# IV Encontro de Software Livre da Paraíba

## 6 a 9 de maio de 2010

Por Fernando Brito

O Grupo de Usuários GNU/Linux da Paraíba, G/LUG-PB, e o Projeto Software Livre, PSL-PB, realizarão neste ano o IV ENSOL, Encontro de Software Livre da Paraíba, que ganhou já ganhou prestígio nacional e atrai participantes de todo Brasil. Saiba um pouco mais deste evento e o que acontecerá de novo nesta edição.

### IV ENSOL

O ENSOL tem como objetivo difundir, promover, apoiar e fomentar a utilização e produção de Software Livre, além de hardware, cursos, técnicas pedagógicas, manuais, livros e traduções cujos autores e/ou proprietários autorizem sua utilização e/ou distribuição sem cobrar remuneração, mediante a renúncia de seus direitos patrimoniais e/ou autorais.

Como na edição anterior, João Pessoa, capital da Paraíba, sediará o encontro, que neste ano será de 6 a 9 de maio. O lema do evento continua "Liberdade no Extremo", o que condiz bastante com a cidade e até mesmo o local do evento, a Estação Cabo Branco - Ciência, Cultura e Arte, que está situada a poucos minutos de caminhada do Farol do Cabo Branco, ponto mais oriental das Américas.

### Convidados

Neste ano de 2010, o IV ENSOL promete su-

perar todas as expectativas deixadas – com toda a razão – pelas edições anteriores. Para quem não lembra o ENSOL já trouxe grandes figuras internacionais, como o Diretor Executivo da Linux International, Jon "maddog" Hall, e o criador da Free Software Foundation (FSF) e do projeto GNU, Richard M. Stallman.

Nesta edição contaremos com a presença de Rasmus Lerdof, criador do PHP, Ryan Ozimek, Diretor da Open Source Matters, organização que cuida do Joomla e Jim McQuillan, fundador e líder do LTSP. Além desses novos nomes o IV ENSOL contemplará a volta do Jon "maddog" Hall, que foi cativado pela nossa receptividade e aceitou o convite de vir ao evento mais uma vez. Para atingir todo o público do evento o ENSOL irá garantir tradução simultânea em todas as palestras cuja língua não seja português.

Mas é claro que não podemos nos esquecer dos representantes nacionais. Desde empresários, engenheiros, passando por estudantes e até mesmo oficiais do Exército, essa diversidade de pessoas é o que garante palestras sobre uma grande gama de assuntos relacionados ao Software Livre. Palestrantes renomados como Julio Neves, Sérgio Amadeu, Fabianne Balvedi, Paloma Costa, Eriberto Mota, Alexandre Oliva, Felipe Augusto van de Wiel, Jomar Silva, Paulino Michelazzo, Marlon Dutra, Sady Jacques, Rodrigo Padula, Rubens Queiroz e tantos outros já estão confirma-

dos no IV ENSOL e o ajudam a se tornar referência nacional quando o assunto é eventos de Tecnologia da Informação e Software Livre.

### João Pessoa

A terceira cidade mais antiga do Brasil, João Pessoa possui uma história de 422 anos (fundada em 1585), bem guardada nos seus monumentos e preservada no verde, que é uma de suas características mais fortes e que lhe rendeu o título de segunda cidade mais arborizada do mundo, atrás, apenas, de Paris. A cidade, que nasceu às margens do Rio Sanhauá, cresceu em direção ao mar. As belas praias também são uma marca de João Pessoa. Dona de um litoral privilegiado, porque possui cerca de 30 quilômetros de praias, todas belas, limpas e quase intocadas.

João Pessoa é Ponto Extremo Oriental das Américas, ou seja, o ponto mais próximo do continente africano. O local onde os raios solares primeiro despontam na América do Sul. Durante todo o ano, a temperatura, em torno de 29 graus, é um convite ao lazer e ao descanso nas águas sempre azuis de praias como: Tambaú, Manaíra, Bessa, Cabo Branco e toda a Costa do Sol. Entre as cidades brasileiras, consideradas com potencialidade turística, João Pessoa é a que apresenta o menor índice de criminalidade.

### Local

O local, por si só, já é algo que fará você se lembrar do ENSOL por um bom tempo. A Estação Cabo Branco - Ciência, Cultura e Arte foi construída recentemente e projetada pelo grande arquiteto Oscar Niemeyer. Sua aparência futurista e localização privilegiada fizeram esta esplêndida construção se tornar rapidamente um dos mais novos e requisitados pontos turísticos de João Pessoa. Com poucos minutos de caminhada você estará no Farol do Cabo Branco, marco do ponto mais oriental das américas, razão para João Pessoa ser conhecida como “a cidade onde o sol nasce primeiro”. De lá é possível obter uma vista

única, de tirar o fôlego, do litoral de João Pessoa.

A comissão organizadora do ENSOL conseguiu, após muito esforço, preencher as palestras e organizar a grade de programação usando apenas 3 auditórios. Após ouvir um feedback da comunidade foi possível perceber que várias pessoas criticam grandes eventos onde ocorre mais de 4 ou 5 palestras simultâneas, pois muitas vezes ocorre um conflito de interesse e o participante acaba tendo que escolher uma dentre várias palestras cujo tema lhe interessa, pois estas ocorrem ao mesmo tempo.

Além dos três auditórios o IV Encontro de Software Livre da Paraíba também contará com um espaço aberto a todo o público, inclusive às pessoas que não realizaram a inscrição no evento. Este espaço, localizado na torre do Estação Ciência, estará reservado ao encontro de diversas comunidades e grupos de Software Livre. Já confirmaram presença e representantes oficiais o Grupo de Usuários Debian, Grupo de Desenvolvedores PHP, GNOME-BR, PSL-PE e vários outros. Este mesmo espaço físico também hospedará a Desconferência, a sala VIP dos palestrantes e um espaço dedicado aos patrocinadores e apoiadores do evento.

### Novidades

Sempre inovando, o ENSOL conta com mais uma série de novidades. A primeira delas é o sorteio de um netbook. Para participar do sorteio tudo que o participante precisa fazer é trazer 3 quilos de alimentos não-perecíveis ao evento. Além de escolher se quer ou não participar do sorteio através da doação, o inscrito também pode, no formulário de inscrição, escolher uma instituição de caridade de preferência. No encerramento das inscrições os votos serão computados e a organização do IV ENSOL irá entrar em contato com as instituições vencedoras para dividir o material arrecadado entre elas.

O IV ENSOL também contará com uma área chamada de Desconferência. Este espaço, locali-

zado na terra do Estação Cabo Branco, irá contar com cerca de 60 cadeiras, um projetor e um quadro. Qualquer um que estiver interessado em simplesmente divulgar suas idéias ou até mesmo realizar uma palestra ali no local basta escrever o tema da apresentação e o horário no espaço disponível no quadro, sempre respeitando aqueles que já reservaram o horário antes de você (incluindo pessoas convidadas pela organização do ENSOL).

Com a vinda de Rasmus Lerdof, criador do PHP, a comunidade PHP se mostrou bastante interessada e após um contato com a comissão do ENSOL foi anunciado, com grande entusiasmo, o I PHP-NE - Encontro Nordestino de PHP, um evento que acontecerá dentro do ENSOL, no dia 6 de maio, às 10h. Mais informações em <http://phpne.ensol.org.br>.

Outra notícia que também repercutiu bastante na comunidade foi a promoção da camisa gratuita. Como as camisas demoram um certo tempo para serem confeccionadas, inicialmente apenas aqueles que se inscreveram no site e realizaram o pagamento até o dia 24 de abril receberiam este belo presente do ENSOL, mas após uma certa pressão e conversa com o nosso fornecedor, conseguimos renegociar os prazos e prorrogar o limite da promoção para o dia 31 de abril.

Todos que já fizeram as inscrições já garantiram a sua camisa. Quem fizer a inscrição (disponível através do nosso site) e efetuar o pagamento até o dia 31 também vai participar da promoção. É possível escolher o tamanho desejado através do próprio formulário de inscrição. Aqueles que deixarem o pagamento para a última hora ou a inscrição para o dia do evento, ficará sem camiseta. E o risco é ainda maior: provavelmente as vagas já terão se esgotado e a pessoa acabará sem poder participar do evento.

### Inscrições

Como já dito anteriormente serão abordados diversos temas em diferentes palestras e workshops. Desde assuntos mais práticos como se-

gurança, desenvolvimento, inclusão digital e social, redes e até mesmo alguns mais teóricos, como filosofia do Software Livre e direitos autorais. O ENSOL conta com conteúdo e informação para toda a sorte de pessoas, seja você um estudante, empresário, entusiasta, curioso, professor ou um profissional da área de Informática.

As inscrições, abertas no dia 25 de janeiro, faltando exatamente 100 dias para o início deste magnífico evento, estão disponíveis através do site oficial. Ao participar do encontro você receberá certificado de presença e um Kit Congressista. As traduções simultâneas necessárias às palestras internacionais também já estão inclusas no pacote. Quem se inscrever até o dia 30 de abril ganhará gratuitamente uma camiseta do evento. Garanta já o seu lugar pois restam pouquíssimas vagas que devem se esgotar dentro dos próximos dias.

A coordenação do evento lhe espera de braços abertos aqui em João Pessoa, ponto mais oriental das Américas, repleto de belas praias e com um clima tropical, tudo isso apenas esperando por você! Venha você também participar do IV ENSOL! Informações no <http://www.ensol.org.br>. 

### Para mais informações:

Site oficial ENSOL  
<http://www.ensol.org.br>



**FERNANDO BRITO** é estudante de Estatística na UFPB e trabalha como programador bolsista no LAViD, Laboratório de Aplicações de Vídeo Digital. Esta é sua primeira vez na organização de um evento. Ajuda como pode, seja atualizando informações no site do ENSOL, seja escrevendo textos de divulgação.

# QUADRINHOS

Por André Farias Oliveira e Wesley Samp

SUPORTE\_



<http://andre.openintelligence.com.br>  
<http://andre221281.wordpress.com/>

OS LEVADOS DA BRECA



<http://www.OSLEVADOSDABRECA.com>

# AGENDA

## MAIO

**Evento: Dia Internacional  
contra o DRM**

Data: 04/05/2010

Local: Em todo o mundo

**Evento: IV ENSOL - Encontro  
de Software Livre da Paraíba**

Data: 06 a 09/05/2010

Local: João Pessoa/PB

**Evento: Tchelinux Erechim URI**

Data: 08/05/2010

Local: Erechim/RS

**Evento: Hora Livre -  
Descobrimo a qualidade do  
desktop no ambiente Linux**

Data: 08/05/2010

Local: Campo Grande/MS

**Evento: 5º Circuito CELEPAR  
de Software Livre**

Data: 21 e 22/05/2010

Local: União da Vitória/PR

**Evento: III Fórum de  
Tecnologia em Software Livre**

Data: 26 a 28/05/2010

Local: Curitiba/PR

**Evento: 5º Circuito CELEPAR  
de Software Livre - Guaíra,  
Paraná**

Data: 28 e 29/05/2010

Local: Guaíra/PR

**Evento: Ruby + Rails no mundo  
Real**

Data: 29/05/2010

Local: São Paulo/SP

Quer seu evento de  
tecnologia divulgado aqui?!  
Então entre em contato  
conosco através do  
contato@espiritolivres.org.

## ENTRE ASPAS · CITAÇÕES E OUTRAS FRASES CÉLEBRES SOBRE TECNOLOGIA

---

“ O DRM ataca nossa liberdade em dois níveis. Seu propósito é restringir o uso das cópias de seu trabalho publicado. Obriga você a utilizar software proprietário, o que significa que você não controla o que ele faz. Quando as companhias se organizam para desenvolver produtos que nos restringem, devemos nos organizar para derrotá-las.

Richard Stallman é presidente do Projeto GNU e da Free Software Foundation

Fonte: <http://www.defectivebydesign.org/>

“Ser grande, é abraçar uma grande causa.”

**William Shakespeare**

