

Realização



projeto software livre
B A H I A

Apoio



III CIBERCON

III Congresso Internacional de Direito
e Tecnologias da Informação

Projeto Software Livre Bahia
<http://www.psl-ba.softwarelivre.org>



Abelmar Bastos

Cartilha de software livre



software livre:
socialmente justo,
economicamente viável e
tecnologicamente sustentável



Software Livre

Histórico

O que é um software? Como surgiu o software livre? O que o diferencia do software proprietário? Como o software livre evoluiu até os dias de hoje? Por que o movimento Software Livre cresce tanto em nosso país? Porque encontramos nele a solução para tantos problemas?

Esse breve histórico não possui a intenção desmedida de responder todas essas perguntas em tão pouco espaço. Esforça-se contudo em ser o pontapé inicial que dará impulso aos novos navegantes de um universo ainda pouco difundido, criado sobre uma nova perspectiva que preza pela colaboração e pela liberdade da informação. Sejam bem-vindos!

O software

O software corresponde à qualquer programa de computador. Ao contrário do hardware (monitores, impressoras, mouse, placas, memórias, etc) o software não é algo físico e por isso não sofre desgaste ao longo do tempo.

Um software é, portanto, uma estrutura lógica, um programa, que realiza funções dentro de um sistema computacional, e é geralmente desenvolvido por programadores que utilizam linguagens de programação para construí-lo. Isso corresponde aos sistemas operacionais (Windows, Linux, Mac OS, etc), drivers que controlam o comportamento de alguns hardwares (driver de modem, de impressora, de placa de vídeo, etc) e todos os aplicativos utilizados pelos usuários finais, como editores de texto (Winword, Notepad, Gedit, OOWriter), planilhas eletrônicas (Excel, OOImpress, Gnumeric), navegadores internet (Mozilla, Internet Explorer, Opera), processadores de imagem (Gimp, Paint, CorelDraw, Adobe PhotoPaint), dentre outros. O software, por não ser físico e sim lógico, pode ser duplicado e armazenado em disquetes, cds, discos rígidos (HD) e sua cópia pode ser transportada de um computador para o outro desde que estejam conectados à uma rede como a internet.

Como surgiu o software livre?

Para entendermos o surgimento do software livre é válido compreender primeiro o que é o software proprietário e o papel das grandes empresas de software na monopolização do conhecimento.

Como mencionado antes, um software é uma estrutura lógica desenvolvida por programadores. Essa estrutura lógica, isto é, os "tijolos" que compõem um software, correspondem aos bits: 0's e 1's. Porém, um programador não

Referências

- [1] http://www.stayfreemagazine.org/ml/assignments/copyright_handout.pdf
- [Debian1] <http://www.debian.org/devel/developers.loc.pt.html>
- [2] www.tableless.com
- [3] <http://www.inference.phy.cam.ac.uk/dasher/portuguese/>
- [4] <http://www.debian.org/devel/debian-accessibility/index.pt.html>
- [pretto, 2003] <http://www.faced.ufba.br>

Interfaces Gráficas (Gerenciadores de Janelas)

- Gnome: www.gnome.org
- KDE: www.kde.org
- Kurumin: www.kurumin.org
- XFCE: www.xfce.org
- WindowMaker: www.windowmaker.org
- FluxBox: www.fluxbox.org

Alguns Softwares Livres bastante utilizados:

- Escritório

- OpenOffice.org(em Pt- Br) : www.openoffice.org.br
- Mozilla : www.mozilla.org.br
- Xmms: www.xmms.org
- Xine: <http://xinehq.de>
- Mplayer: <http://www.mplayerhq.hu>

- Editores de Imagem

Editores Vetoriais:

- Sodipodi : www.sodipodi.com
- Inkscape : inkscape.sf.net

Editores de mapas de bits:

- The Gimp (com características vetoriais) : www.gimp.org

- Jogos

- Armagetron : armagetron.sf.net
- Stratagus game engine: <http://stratagus.sourceforge.net>
- Chromium: <http://www.reptilelabour.com/software/chromium/index.htm>
- Portal de jogos GNU/Linux: <http://tuxgames.gratishost.com>
- Outro portal de jogos GNU/Linux: <http://www.happypenguin.org>

pode desenvolver um programa se utilizando de 0's e 1's. Isso é uma linguagem somente compreensível pela máquina e não por seres humanos. Para tanto, os programadores se utilizam de linguagens de programação que possui palavras chaves e estruturas que permitem enxergar um programa como uma receita de bolo, isto é, um conjunto de instruções denominado código fonte. Por exemplo:

- 1) Programa pede que usuário digite sua idade e tecla enter
- 2) Programa recebe a informação (idade)
- 3) Se idade for maior ou igual a 18 mostra mensagem na tela: "Você já é um adulto!"
- 4) Se idade for menor que 18 mostra mensagem na tela: "Você ainda é adolescente"

obs.:As linguagens de programação disponíveis não são tão simples de entender como a do exemplo acima, mas se esforçam no sentido de se tornarem cada vez mais fáceis de programar e de serem entendíveis.

O código fonte seria a representação desses passos em uma denominada linguagem de programação (C++, C, Java, Pascal, etc). Esse código fonte ao ser processado por um outro programa chamado compilador se transforma em 0's e 1's para que possa ser entendido pela máquina. Quem quiser modificar o programa precisa ter o código fonte já que 0's e 1's não são compreendidos pelo homem.

Antigamente os programadores compartilhavam seus códigos fonte uns com os outros e assim todos podiam modificar o programa e compartilhar essas mudanças também. Este hábito era bastante difundido nas grandes universidades norte- americanas das décadas de 60 e 70 e sempre foi bastante condizente com o espírito acadêmico, cujos princípios de liberdade e cooperação assemelham- se com a cultura hacker.

O software proprietário

Quando o computador foi abraçado pelo mercado como uma boa idéia para seus negócios, as coisas começaram a mudar de rumo. As pesquisas em desenvolvimento de novas soluções cresceram e empresas começaram a criar seus produtos para se firmar. Começaram a aparecer, então, programas que tinham seu código fonte escondido como segredo comercial. As empresas vendiam seus softwares, mas não disponibilizavam seus códigos fontes. O usuário só recebia o programa em 0's e 1's o que tornava possível somente utilizar o software mas não mais possível modificá- lo. Isso era estrategicamente interessante para as empresas: ela poderia desenvolver ótimos softwares e ninguém saberia como isso foi feito, nem seus clientes e muito menos seus concorrentes.

Além disso surgiram outros artifícios: os programadores dessas empresas

assinavam termos de compromisso de não divulgação dos segredos da empresa e os softwares vendidos possuíam licenças cheias de restrições que deviam ser acatadas pelos seus clientes. Essas licenças incluíam regras para controle de pirataria: os clientes, além de impossibilitados de modificar o programa, não poderiam fazer cópias dos programas que eles adquiriram, e muito menos poderiam distribuir essas cópias. Alguns programas também só poderiam ser utilizados para fins específicos. Resumindo, o cliente não era o dono do software. Ele podia somente comprar uma licença de uso que restringia o uso daquele programa em apenas 1 (um) computador. Essa lógica logo foi se tornando o pensamento comum dentro de uma sociedade constantemente bombardeada por propagandas de conteúdo duvidoso. Além disso, licenças de software são mais caras do que deveriam. A má notícia é que poucas pessoas possuem recursos para adquirir um software proprietário e utilizá-lo em casa. Até mesmo os revendedores de computadores evitam fornecer computadores com softwares proprietários embutidos. Como consequência, os computadores já vêm com cópias ilegais destes softwares. A conhecida pirataria está presente na maioria dos lares do mundo em que há um computador.

É importante ressaltar que o software distribuído pelas empresas de software contém, embutido, o conhecimento de milhares de programadores geniais, que por sua vez se valeram do conhecimento de várias pessoas que vieram antes deles, e desenvolveram teorias e conhecimento que deixaram para a humanidade. Estas empresas agora apropriam-se de todo esse conhecimento acumulado e que poderia ser compartilhado com outras pessoas.

A reação: Uma licença para a liberdade

Tudo permaneceu assim até que um grupo de hackers programadores do MIT (Instituto de Tecnologia de Massachussets, nos EUA) perceberam que podiam fazer a diferença.

Os hackers, diferentemente de como são conhecidos e tratados pela mídia, são pessoas com princípios éticos, defensores da cooperação e da disseminação do conhecimento através da liberdade da informação. São caracterizados como pessoas de elevado conhecimento técnico que amam o que fazem e que sentem prazer em compartilhar isso com os outros. Os hackers, diferentemente dos crackers e dos defacers não são criminosos digitais e possuem uma estrutura hierárquica implícita baseada na cultura da dádiva (Gift Culture). Quanto mais contribuições um hacker oferece para a sua comunidade, mais status ele possui dentro dela. Os hackers também são chamados de geeks.

Essa reação começou pequena, com poucos colaboradores. Um dos programadores do MIT, Richard Stallman, enfurecido pela comercialização do conhecimento, fundou o Projeto GNU: um projeto para criar uma

Links

Mais sobre Software Livre

- Projeto GNU: <http://www.gnu.org/home.pt.html>
- Manifesto de Hipatia (Português): www.hipatia.info/mh.pt.html
- Manual Foca Linux: <http://focalinux.cipsga.org.br>
- Noticias sobre o mundo Linux: www.noticiaslinux.com.br
- Tabela de softwares equivalentes entre Windows e Linux: www.linuxshop.ru/linuxbegin/win-lin-soft-en/table.shtml
- Projeto Software Livre Brasil: www.softwarelivre.org

Projeto Software Livre nos estados brasileiros

- Projeto Software Livre Brasil: www.softwarelivre.org
- Projeto Software Livre Bahia: www.psl-ba.softwarelivre.org
- Projeto Software Livre Distrito Federal: www.psl-df.softwarelivre.org
- Projeto Software Livre Espírito Santo: www.psl-es.softwarelivre.org
- Projeto Software Livre Mato Grosso do Sul: www.psl-ms.softwarelivre.org
- Projeto Software Livre Minas Gerais: www.psl-mg.softwarelivre.org
- Projeto Software Livre Paraná: www.psl-pr.softwarelivre.org
- Projeto Software Livre Pernambuco: www.psl-pe.softwarelivre.org
- Projeto Software Livre Rio de Janeiro: www.psl-rj.org.br
- Projeto Software Livre Rio Grande do Sul: <http://www.psl-rs.softwarelivre.org>
- Projeto Software Livre Santa Catarina: www.softwarelivre.sc.gov.br
- Projeto Software Livre São Paulo: <http://www.psl-sp.org/>
- Software Livre no INPA - Amazonas: <http://softwarelivre.inpa.gov.br/>

Distribuições GNU/Linux

- Kurumin Linux: www.kurumin.com.br
- Debian GNU/Linux: www.debian.org
- Slackware Linux: www.slackware.org
- Suse Linux: www.suse.com
- Mandrake GNU/Linux: www.mandrakelinux.com
- Red Hat: www.redhat.com
- Knoppix GNU/Linux: www.knoppix.net
- Gentoo Linux: www.gentoo.org

nada é possível fazer.

Soluções livres existem, e muitas delas são bastantes eficientes. As opções de acessibilidade do gerenciador de janelas KDE e do Gnome são ótimos exemplos.

A filosofia do software livre permite que o código dos programas seja alterado, de maneira que eles possam ser integrados com os programas que visam oferecer acessibilidade. Assim eles podem, por exemplo, ser acessados e até controlados por outros programas. Isso facilita que sejam implementados teclados virtuais, controladores por voz, dispositivos de mouse alternativos (como aqueles controlados pelo movimento do globo ocular ou até por movimentos respiratórios).

Um fantástico exemplo de aplicação para deficientes motores é o Dasher[3]. Ele permite escrever, mas não é exatamente um teclado virtual. Utiliza uma idéia inovadora para inserir texto através do teclado. É especialmente útil para escrever em palmtops, pois permite escrever muito mais rápido do que na forma tradicional (clitando sobre cada letra numa figura de teclado).

Na seção de "Acessibilidade do projeto Debian"[4] é possível encontrar muitas outras opções livres de acessibilidade, para as mais diversas necessidades.

plataforma de software totalmente livre, ou seja, que qualquer pessoa pudesse:

1. Utilizar o software para qualquer fim
2. Estudar o código do software
3. Modificar o código do software
4. Redistribuir cópias do software

A única condição para que a pessoa pudesse usufruir desses direitos era que ela passasse para o próximo os mesmos direitos que usufruiu. Esses quatro direitos ficaram conhecidos como as quatro liberdades do Software Livre. O Projeto GNU conseguiu transformar essas liberdades em uma nova licença de uso chamada GNU GPL (GNU General Public Licence). E dessa forma todo software distribuído com essa licença estava "livre". Vale lembrar que a licença não proíbe a venda do software livre, somente garante que o software conterà as 4 liberdades mencionadas e que essas liberdades nunca poderão ser retiradas do software. O software livre já nasce livre e permanecerá livre.



O projeto GNU também foi o responsável pelo desenvolvimento de inúmeros softwares livres. Para cada software proprietário existente os desenvolvedores faziam um outro similar, livre. Um dos softwares mais essenciais, contudo, foi desenvolvido pelo finlandês Linus Torvalds que por conta própria desenvolveu o núcleo do sistema operacional (kernel) e o batizou de Linux. O Linux em conjunto com os softwares desenvolvidos pelo Projeto GNU formaram o primeiro sistema computacional livre, o GNU/Linux que atualmente é a principal

alternativa ao Microsoft Windows.

Como o Software Livre não é uma entidade única, ele não pode ser comprado pela Microsoft ou qualquer outra empresa. O Software Livre também não vai à falência, já que a única premissa para que ele exista é a de que existam pessoas com uma necessidade e com disposição de compartilhar seus códigos fontes. O número de indivíduos que participam dessa iniciativa e o número de empresas que tem apostado suas fichas nela indicam que isso está em crescimento acelerado.

O nosso contexto

O software livre surge então da necessidade de abandonarmos o velho papel de meros usuários da tecnologia e passarmos a desenvolvê-la e usá-la para o bem de todos. O Brasil particularmente, precisa acabar com a sua dependência tecnológica e passar também a desenvolver softwares, ao invés de continuar

refêm dos preços abusivos impostos pelas grandes corporações e seus monopólios.

O software livre é a nossa chance de tornar a tecnologia a nossa aliada no desenvolvimento nacional. O governo, sensível à essa oportunidade, vem incentivando cada vez mais o uso de softwares livres nas repartições públicas, reduzindo drasticamente os custos com licenças de software proprietário. Esse recurso economizado poderá ser redirecionado para setores mais problemáticos como a saúde e a educação, minimizando a injustiça social, ou até mesmo para investimentos em tecnologia nacional.

Conceito de Software Livre

Software Livre é uma questão de Liberdade, não de preço. O termo Software Livre se refere à liberdade que o usuário tem de executar, distribuir, modificar e repassar as alterações sem, para isso, ter que pedir permissão ao autor do programa.

Pode ser definido mais claramente pelas 4 liberdades defendidas pela Free Software Foundation para os usuários de software:

- A liberdade de executar o programa, para qualquer propósito;
- A liberdade de estudar como o programa funciona, e adaptá-lo para as suas necessidades. Acesso ao código-fonte é um pré-requisito para esta liberdade;
- A liberdade de redistribuir cópias de modo que você possa beneficiar o próximo;
- A liberdade de aperfeiçoar o programa, e liberar os seus aperfeiçoamentos, de modo que toda a comunidade se beneficie. Acesso ao código-fonte é um pré-requisito para esta liberdade.

Um programa será considerado livre se todos os seus usuários tiverem essas 4 liberdades.

Notem que os quatro itens acima não fazem nenhuma referência a custos ou preços. O fato de se cobrar ou não pela distribuição ou pela licença de uso do software não implica diretamente em ser o software livre ou não. Nada impede que uma cópia adquirida por você seja revendida, tenha ela sido modificada por você ou não.

Nada impede, também, que as alterações feitas por você num software para uso próprio sejam mantidas em segredo. Você não é obrigado a liberar suas modificações, se não quiser. Porém, se escolher fazê-lo, é obrigado a distribuir de maneira livre. Essa é uma observação importante a se fazer porque muitas pessoas (especialmente corporações) têm receio de usar software livre porque temem que seus "concorrentes" tenham acesso a informações e métodos de trabalho privados. As personalizações não têm

Acessibilidade

Quando falamos em acessibilidade, a idéia é tentar dar a todos a oportunidade de usar os recursos que o computador oferece. Porém nem todos esses recursos estão disponíveis para quem é portador de algum tipo de deficiência.



Imagine que você tenha uma deficiência motora, que lhe impeça de mover um dos braços. Isso já pode constituir uma enorme dificuldade, pois além de tornar a digitação extremamente lenta, é impossível usar mouse e teclado ao mesmo tempo. Se os movimentos são restritos em ambos os braços, então a situação praticamente impossibilita o uso do computador.

Outra restrição grave são as deficiências visuais. Uma pessoa cega ou com dificuldades de enxergar, ou mesmo dificuldades de leitura, ou problemas para diferenciar as cores praticamente não tem recursos que lhe permitam usar o computador de forma eficiente. Algumas soluções são propostas para esses problemas.

A maioria das soluções proprietárias entretanto, são muito caras, e algumas delas não são satisfatórias. Para deficientes motores, existem softwares de teclado virtual, que permite escrever utilizando o mouse, e softwares controlados por voz, que permitem acessar aplicações e menus através de comandos de voz.

Existem ainda opções de acessibilidade como criação de atalhos de teclado, teclas de alternância, teclas de repetição, de repercussão, bips de alerta, etc. Essas opções podem ajudar bastante quem tem dificuldade de digitar, ou tem restrições motoras. Para deficientes visuais, há também os leitores de tela. Elas estão disponíveis tanto para os sistemas operacionais proprietários quanto para os livres. As dificuldades encontradas, entretanto, ainda são muitas.

Páginas web mal feitas são praticamente impossíveis de serem lidas por alguns programas. Uma solução para isso é programar páginas para internet utilizando as técnicas de tableless[2], que deixam o código limpo e fácil de entender (tanto para os seres humanos quanto para o software que vai tentar interpretá-lo).

Para implementar um software que ofereça acessibilidade aos deficientes, é necessário que os programas com os quais ele tem que interagir (como editores de texto, menus, etc) permitam uma certa integração com este software. Programas de código fechado normalmente não permitem que essa integração exista. O motivo é óbvio: se não temos como alterar um programa, e ele não foi projetado para permitir que fosse controlado por outra aplicação,

estabelecida, os excluídos digitais apresentam-se como os marginais aos meios de acesso à informação e geração de conhecimento. Com efeito, estes excluídos estão alheios ou dificilmente se revelam aos processos de alcance das necessidades subsistenciais, de cidadania, de interação social e de consciência do mundo em volta.

A inclusão digital possui o papel de resgatar os excluídos digitais ao contexto da sociedade movida pelos processos de criação, produção e sublimação da informação em conhecimento. Significa efetivar os excluídos digitais na sociedade da informação, por meio de políticas que visem ao seu crescimento auto-sustentável de forma colaborativa e gradual, não com medidas emergenciais e paliativas. Conseqüentemente, inclusão digital remete à busca da reflexão do mundo e da localidade, das condições de sobrevivência (emprego, alimentação, moradia, etc.), do estímulo ao conhecimento renovado e à crítica do já existente e da diminuição das desigualdades sociais.

Tabuleiros Digitais

Projeto desenvolvido na FAGED/UFBA[] com o Liceu de Artes e Ofícios. Trata-se de um modelo de móvel que representa mais do que uma simples bancada para suporte de computadores - inspirado em soluções do cotidiano das trabalhadoras do acarajé [pretto, 2003].

O Tabuleiro Digital é reto, sem encostos, sem almofadas, projetado para uso rápido e ágil como o do tempo de comer um bom acarajé ou ler e responder meia dúzia de e-mails.

Por que um tabuleiro digital?

É simples de montar e pode não ter grandes sofisticação de acabamento. Foi desenvolvido para poder ser construído a partir de uma folha padrão de compensando, utilizando-se da sabedoria da baiana de acarajé. Basta cortar, lixar e pintar com verniz naval fosco as peças. Após a montagem, os tabuleiros podem ser dispostos em “ilhas” (agrupamento de tabuleiros isolados). É um modelo prático para adoção em espaços de acesso à internet, como infocentros, telecentros, ciberparques.



que ser distribuídas. A restrição é que, se elas forem distribuídas de alguma maneira, têm que manter as 4 liberdades descritas acima.

A liberdade de utilizar um programa significa a liberdade para qualquer tipo de pessoa física ou jurídica utilizar o software em qualquer tipo de sistema computacional, para qualquer tipo de trabalho ou atividade, sem que seja necessário comunicar ao desenvolvedor ou a qualquer outra entidade em especial.

A liberdade de redistribuir deve incluir a possibilidade de se repassar tanto os códigos-fonte quanto os arquivos binários gerados da compilação desses códigos, quando isso é possível, seja o programa original ou uma versão modificada. Não se pode exigir autorização do autor ou do distribuidor do

software para que ele possa ser redistribuído.

Para que seja possível modificar o software (para uso particular ou para distribuir) é necessário ter o código-fonte. Por isso o acesso aos fontes é pré-requisito para esta liberdade. Caso ele não seja distribuído junto com os executáveis, deve ser disponibilizado em local de onde possa ser copiado, ou deve ser entregue ao usuário, se solicitado.



Para que essas liberdades sejam reais, elas têm que ser irrevogáveis. Caso o desenvolvedor do software tenha o poder de revogar a licença o software não é livre.

Licenças de Software

Para falar sobre Software Livre é indispensável que comecemos falando em Direitos Autorais e Copyright, uma expressão criada pelos estadunidenses com o objetivo de dar exclusividade de edição de materiais de imprensa escrita aos seus detentores.

Dessa forma, autores que possuíssem o Copyright de suas obras poderiam designar quem poderia, e como poderia, copiar e distribuir cópias de seus livros, artigos ou revistas. Entretanto, como não é necessário ser autor da obra para deter o seu Copyright, não podemos dizer que direito autoral é igual a Copyright.

A definição formal de Copyright [1] é a que segue abaixo:

“O único direito legal concedido a um escritor, a um editor, a um compositor, ou a um distribuidor para publicar, produzir, vender, ou distribuir um trabalho artístico.”

E na Constituição Estadunidense (em <http://www.copyright.gov>), na lei que define Copyright, está o texto abaixo (tradução dos editores):

“Usado para promover o progresso da ciência e das artes úteis, fixando-se por épocas limitadas aos autores e aos inventores o direito exclusivo sobre suas respectivas escritas e descobertas.”

Enquanto isso, o Direito Autoral é o direito que o autor do software tem de deixar seu produto sob um Copyright (veja que o copyright pode não ser seu), sem necessariamente estabelecer regras de uso, cópia e distribuição. Por exemplo, um funcionário faz um Software em uma empresa e concorda em ceder-lhe o seu copyright, ele abre mão de definir como será usado, copiado e distribuído o programa, mas não deixa de ser o autor e pode reclamar se o software for usado para um fim não previamente acordado com a Empresa.

O sistema de copyright atribui “donos” ao software, e esses, pela definição apresentada acima, têm o direito de estabelecer regras de distribuição do software. Devido a esse poder, surgiu o conceito de Software Proprietário, um paradigma de distribuição de software vigente desde o início da década de 80, que foi responsável pela criação do chamado “software de caixinha”.

Porém, ao contrário do que muitos pensam, simplesmente existir o copyright de um software não basta para que ele seja Software Proprietário. O copyright, ou Direito de Uso, permite que o detentor desse direito estabeleça regras para o uso e distribuição do software. Essas regras são chamadas de licenças de uso e podem estabelecer por exemplo se o software pode ou não ser distribuído gratuitamente, se pode ou não ser copiado, entre outras regras...

Software Livre não é de domínio público, apesar de essa ser uma interpretação compreensível do conceito de “liberdade”, para quem ainda não se familiarizou com a quebra do paradigma do software proprietário.

Um programa de domínio público é aquele em que o criador abre mão de seus direitos de autoria e de licenciamento de cópias. Nesse caso, quem estiver de posse do código tem o direito de fazer dele o que desejar, sem ter que obedecer qualquer restrição ou norma.

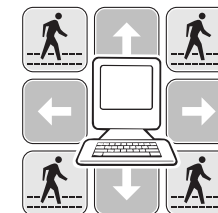
No paradigma do Software Livre, o autor do software resguarda seus direitos de criador, mantendo livres o uso e o conhecimento (do código fonte) do software para quem deles precisar através da redação adequada de um copyright.

Exatamente dessa subversão do sentido do copyright para proibir que haja restrições, em vez de restringir o uso, vem o termo Copyleft. Um trocadilho com a palavra em Inglês (deixe copiar, ao invés de direito de cópia), mas que não tem significação legal.

Inclusão Digital

O que é Inclusão Digital ?

"Inclusão Digital" é a denominação dada, genericamente, aos esforços desenvolvidos por governos, organizações do terceiro setor e empresas no sentido de possibilitar que as pessoas possam:



- obter os conhecimentos necessários para utilizar com um mínimo de proficiência os recursos de tecnologia de informação e de comunicação existentes
- dispor de acesso físico regular a esses recursos.

As estruturas e o funcionamento da sociedade contemporânea estão sendo significativamente alterados pelas tecnologias de informação e de comunicação. Favorecer a todos cidadãos o acesso a esses meios é um importante passo no combate à exclusão social.

No entanto, há uma discussão emergindo sobre o uso do termo inclusão digital. Existe, por exemplo, a crítica da banalização do termo, especialmente por conta da exploração política oportunista. Há também o "modismo" vinculado ao uso sensacionalista do termo.

Todavia, já que se caminha por essas trilhas, seria enriquecedor apurar os significados atribuídos ao termo, para que assim se possa anular possíveis "explorações", trazendo-o para um contexto sócio-político real, que o requisita em toda a sua profundidade e possibilidades. Então, iniciemos um percurso com esse objetivo.

Divagações sobre Inclusão Digital

Quer parecer politicamente correto? Fale em "inclusão digital". Não porque a expressão tornou-se recentemente vulgar de tão pronunciada, sendo um aparente modismo, mas pela relevância por ela alcançada. Em tempos de sociedade determinada pelo conhecimento adquirido e sua possibilidade de aplicação, o acesso à informação acena como peça-chave, a partir do qual muitos caminhos dentro do ambiente social podem ser contemplados. À primeira vista, isto decorre do contato com as tecnologias da informação, seja através do reconhecimento inicial do conteúdo em formato digital, do uso de ferramentas tecnológicas existentes na produção de conhecimento novo, ou da construção de novas ferramentas capazes de atender às demandas permanentes do mundo da informação.

Entretanto, um numeroso contingente de indivíduos está de fora deste já complexo universo sintético de consumo e produção de dados. Este são os denominados "excluídos digitais". Do mesmo modo que se convencionou tratar os desprovidos dos bens sociais básicos como marginais à sociedade

<i>Internet</i>	<i>Windows</i>	<i>GNU/Linux</i>
Navegadores	Internet Explorer	Mozilla, Firefox, Opera, Konqueror, Epiphany
Cliente de e-mail	Outlook Express, Eudora, Netscape Messenger	Evolution, Mozilla Messenger, Thunderbird, Kmail
Downloader de arquivos	Getright, DAP	Prozilla, Kget, Gnome Transfer Manager, GetLeft
Cliente FTP	CuteFTP	Gftp, Konqueror
Cliente IRC	Mirc	Xchat, Kvirc, BitchX
Cliente de Mensagens ICQ	ICQ Lite, MSN, AIM, Yahoo	Licq, Micq, GnomeICU, Gaim, Kopete, aMSN, Yahoo Messenger for Unix, AIM, Everybuddy, centericq
Videoconferência	NetMeeting	GnomeMeeting, NeVoT, IVS
Compartilhador de arquivos	Morpheus, Napster, Kazaa, eDonkey, Bittorrent	Lopster, Gnapster, eDonkey, Xmule, FreeNet, Bittorrent, Nicotine, aMule, LimeWire

<i>Jogos</i>	<i>Windows</i>	<i>GNU/Linux</i>
Tetris	Tetris	Ltris, Xwelltris
Minas	Mines	Kmines, Perlmines, Dmines
Estratégia	Civilization, Sim City 3000, Warcraft 2	FreeCiv, Sim City 3000, Stratagus
Quake 1,2,3	Quake 1,2,3, QuakeForge, DarkPlaces	Quake 1,2,3 QuakeForge, DarkPlaces
Corrida	Need for Speed	Tux Racer, KartlingRace
Simulador de vôo	MS Flight	Simulator FlightGear
Lemmings	Lemmings	Pingus
Xadrez	ChessMaster	Glchess, Xboard, Eboard



Ao invés de encararmos o software como uma idéia, uma forma de se solucionar um problema, nós ainda o vemos como um item final de uma linha de produção; um artigo físico, que se desenvolveu como qualquer outro produto, em escala industrial, e pelo qual se deve pagar. Nessa definição, o software se assemelha mais a uma cadeira ou uma mesa, do que a uma fórmula matemática, ou uma especificação científica.

Imaginem se o algoritmo para achar as raízes de uma equação do segundo grau, desenvolvido por Bháskara há dezenas de séculos, tivesse uma licença que restringisse seu uso. Todo o desenvolvimento matemático estaria restrito à boa vontade do criador da fórmula, a quem teríamos que solicitar autorização para utilizar o método.

Coisa semelhante acontece com programas de computador. Se considerarmos o programa como uma maneira de se resolver determinado problema, com a ajuda do computador, fica mais simples aceitar o software como uma idéia, não como um bem.

E o que queremos é que as boas idéias sejam utilizadas em benefício de todos, e que todos possam usufruir das idéias e das novidades da tecnologia e da ciência.

Para garantir essa liberdade, no caso do software para computador, a Free Software Foundation redigiu algumas licenças, as quais aplicadas ao programa, mantém os direitos de autoria ao implementador do software, dando aos usuários do programa certas liberdades.

No caso da mais conhecida das licenças para Software Livre, a GNU General Public License (GPL), essas liberdades incluem o direito de estudar, alterar para que se adeqüe aos seus interesses, copiar e redistribuir o software, desde que não retire do próximo esses mesmos direitos.

Há outras licenças, algumas mais, outras menos restritivas. A Lesser GPL (LGPL) -- também redigida pela FSF, por exemplo, é mais permissiva que a GPL. Bibliotecas de funções distribuídos sob a LGPL podem ser utilizadas por software proprietário, apesar de serem livres. Se essas mesmas bibliotecas estiverem sob a GPL, elas podem ser usadas apenas por programas também sob a GPL.

Por que usar Software Livre?

Para obter uma melhor compreensão dos motivos que cercam os usuários de Software Livre, vale lembrar a forma como ele surgiu e o que ele representa para a sociedade no contexto atual. O Software Livre surgiu baseado no conceito de liberdade no qual as pessoas tem o direito garantido às 4 liberdades já mencionadas. Na visão filosófica do Software Livre, a liberdade não é um direito individual, é um direito coletivo e por isso deve ser mantido e passado de pessoa para pessoa. Além disso a premissa de qualquer projeto de Software Livre é a colaboração entre as pessoas interessadas, sem concentração de poder ou qualquer outro artifício que venha a ferir as liberdades já mencionadas. Tendo em vista esse cenário podemos facilmente citar algumas das razões de se utilizar Software Livre:

- Poder utilizar o software para qualquer finalidade.
 - Ter acesso ao código fonte e poder modificá-lo, sem quaisquer restrições.
 - Poder copiá-lo e executá-lo em quantas máquinas desejar.
 - Poder distribuí-lo, sem violar, é claro, essas liberdades a que todos tem direito.
 - Ter o seu computador equipado com software de qualidade a um custo baixo ou nulo.
 - Não ficar preso às restrições impostas pelas licenças de softwares proprietários.
 - Não ficar dependente de novas versões com preços abusivos que eventualmente apresentam incompatibilidades com versões antigas.
 - Ficar livre da pirataria.
- Incentivar o desenvolvimento de tecnologia nacional.
- Interagir e compartilhar soluções com sua comunidade, seja ela física ou virtual.
 - Lutar contra o monopólio de grandes corporações que buscam se apropriar do conhecimento intelectual coletivo.

Tabelas de softwares equivalentes

Essa tabela é um resumo baseado na tabela de equivalências entre softwares livres e proprietários feita polonês Valery V. Kachurov.

Escritório	Windows	GNU/Linux
Suite Office	MS Office	OpenOffice.org, GnomeOffice, Koffice
Processador de textos	MS Word	Abiword, OpenOffice.org Writer, Kword
Planilhas	MS Excel	Gnumeric, OpenOffice.org Calc, Kspread
Apresentações	MS PowerPoint	OpenOffice.org Impress, Kpresenter
Banco de Dados	MS Access, Oracle, MS SQL Server	PostgreSQL, MySQL, Firebird, PrevaYler
Finanças	MS Money	GNU Cash
Gerenciador de Projetos	MS Project	Mr. Project

Multimídia	Windows	GNU/Linux
Tocador de MP3/OGG	Winamp, Windows Media Player, MusicMatch Jukebox	XMMS, Mplayer, Xine
Gravador de CD	Nero, Easy CD Creator	K3b, XCDRoast, Gnome Toaster, WebCDWriter, CRecord
Tocador de CD	CD Player, Winamp, Windows Media Player	Grip, Gnome CD
Visualizador de vídeos	Windows Media Player, RealPlayer, QuickTime, Winamp3	Mplayer, Xine, KDE Media Player, VLC
Tocador de DVD	WinDVD, MicroDVD, Windows Media Player	Mplayer, Xine, Aviplayer, Ogle, VLC
Extrator de MP3	MusicMatch, Real Jukebox	Grip, Lame, NotLame
Editor de áudio	SoundForge, CoolEdit	WaveForge, GNUSound, Glame, Sweep

Gráficos	Windows	GNU/Linux
Visualizador de imagens	ACDSee, Image Viewer	Eye of Gnome, GQView, Kview, Gthumb, GTKSee, Kuickshow
Visualizador de fotos (máquina digital)	Polaroid Drivers	Gtkam, Gphoto2
Editor de imagem (básico)	Paint	Kpaint, Tuxpaint, Gpaint
Editor de fotos	Adobe Photoshop, Corel PhotoPaint, Macromedia Fireworks	Gimp, ImageMagick, CinePaint
Editor de imagens vetoriais	Corel Draw	Sodipodi, OpenOffice.org Draw, Dia, Xfig

Ferramentas Desktop	Windows	GNU/Linux
Visualizador de PDF	Adobe Acrobat Reader	GhostView, Xpdf, kghostview
Gerador de PDF	Adobe Acrobat Distiller	PDFLatex, GV, Xfig, Ghostscript, kghostview
Programas de Scanner	Cds que acompanham o scanner	Xsane, kooka
Anti-vírus	Norton Antivirus, TrendMicro, Dr. Web	OpenAntiVirus, Clam Antivirus

GNU/Linux. O Windows é um sistema onde tudo, inclusive a interface gráfica, faz parte de uma estrutura inflexível, não estendível e pouco configurável.

No GNU/Linux a interface gráfica se tornou uma realidade viável e acessível aos usuários leigos a pouco tempo. Porém como o sistema é modular e a interface gráfica representa apenas uma parte estendida do sistema operacional, a sua manipulação e extensibilidade se evidencia na enorme quantidade de gerenciadores de janelas disponíveis na internet. Cabe ao usuário escolher aquele que mais lhe agrada e configurá-lo ao seu gosto.

XFce

Xfce é um gerenciador de janelas bastante leve desenvolvido para qualquer sistema operacional baseado no Unix, incluindo o próprio GNU/Linux. Ele objetiva ser rápido e leve, enquanto mantém uma aparência agradável e possui facilidade de uso. O Xfce consiste em um número de componentes que juntos provêm todas as funcionalidades do gerenciador de janelas. Esses componentes são empacotados separadamente e assim você pode escolher e instalar alguns dos pacotes disponíveis para criar uma interface gráfica mais personalizada.

Window Maker

Esse gerenciador de janelas foi criado por um "brazuca" e tem sua interface gráfica toda baseada em quadriculos. Suporta integração com o Gnome e com o KDE, é rápido, rico em funcionalidades, super configurável, e relativamente fácil de utilizar. Possui também um editor de configurações em modo gráfico chamado WPrefs que facilita a configuração do ambiente. É um ambiente leve que consome poucos recursos da máquina.

Fluxbox

O Fluxbox é um gerenciador de janelas baseado no Blackbox, e possui total compatibilidade com este. O Fluxbox herda portanto as características do Blackbox que por sua vez é um dos gerenciadores mais rápidos e leves. Além disso a sua instalação requer poucas bibliotecas, ao contrário da maioria dos gerenciadores de janela.

Gnome

É o gerenciador de janelas oficial do Projeto GNU. Possui um interface gráfica intuitiva e atrativa para usuários finais e é altamente configurável. É desenvolvido por uma comunidade e é muito mais que um gerenciador de janelas já que possui centenas de aplicações associadas. Possui várias características que sinalizam uma preocupação crescente com a inclusão de usuários no mundo do Software Livre: internacionalização e acessibilidade são exemplos disso. Isso quer dizer que o Gnome é desenvolvido e documentado em diversos idiomas além de possuir programas para usuários especiais e deficientes físicos. O Gnome porém é relativamente pesado e não é recomendável para máquinas obsoletas.

KDE

O K Desktop Environment surge junto ao Gnome como uma das opções mais populares. Além disso os dois projetos são bastante parecidos. O projeto KDE porém, só ganhou mais força dentro da comunidade Software Livre depois que uma das suas peças chaves, a biblioteca Qt, passou de proprietária para livre. Atualmente o KDE vem se destacando por ser a interface gráfica nativa da distribuição Kurumin. Também não é recomendável para máquinas obsoletas.

GNU/Linux

Sobre o GNU/Linux:

O Linux é um sistema operacional criado pelo finlandês Linus Torvalds e mantido, atualmente, por vários desenvolvedores pelo mundo. O nome "Linux" veio de uma mistura do primeiro nome do criador: "Linus" + "Unix". O Unix é um sistema operacional proprietário desenvolvido pela Bell Labs no início dos anos 70 e que serviu de base para uma variedade de outros sistemas operacionais.

O desenvolvimento do Linux teve início em 1991, quando Linus, na época estudante de Ciência da Computação na Universidade de Helsinki, Finlândia, resolveu desenvolver um sistema baseado no Minix, porém mais robusto. O Minix é um sistema operacional simples criado por Andrew S. Tanenbaum, com o objetivo de ser usado para estudos acadêmicos.

Então porque chamar de GNU/Linux? Acontece que Linus Torvalds desenvolveu a parte central do sistema operacional, conhecida como kernel, e o nome Linux deve ser atribuída somente a esse núcleo. Como ninguém vai utilizar um sistema operacional que só possui o kernel, era necessário para Linus Torvalds desenvolver os aplicativos a serem executados.

Enquanto Linus estava desenvolvendo o Linux, Richard Stallman objetivava a criação de um sistema computacional totalmente livre e ao contrário de Linus ele começou o desenvolvimento pelos aplicativos, como compiladores, editores de textos, etc.

Foi da união dos aplicativos desenvolvidos pelo projeto GNU (de Richard Stallman) com o kernel do Linux (criado por Linus Torvalds) que surgiu o GNU/Linux, comumente chamado de

Linux pela grande maioria das pessoas e pela mídia.

O bom desempenho aliado a segurança e a possibilidade de personalização do sistema, devido ao código fonte ser aberto, tornou o GNU/Linux ideal para ser utilizado em servidores. A sua utilização em máquinas de usuário comum tem crescido bastante após a melhoria das interfaces gráficas, o que levou o GNU/Linux a ser o alicerce de muitos projetos de inclusão digital.



Distribuições GNU/Linux

Distribuições GNU/Linux são "estruturas" definidas através de uma metodologia que criam uma "personalidade" para esse sistema operacional. São maneiras diferentes de agrupar o software necessário para se usar o sistema.

Algumas distribuições focam o uso do sistema operacional para um fim específico (servidores, super computadores, uso doméstico, etc.).

Outras procuram ser abrangentes, oferecendo tudo o que pode ser necessário (e até o que provavelmente nunca será necessário) em sua distribuição.

Algumas enfocam a estabilidade, outras o moderno, outras a quantidade de opções; algumas querem ser a mais amigável ao usuário, com a intenção de tornar a instalação ou o uso mais fácil, entre tantas outras características.

A distribuição ideal é aquela que satisfaça melhor as necessidades específicas do usuário. Cabe a cada um estabelecer quais são suas necessidades, e a partir daí buscar a distribuição que melhor atenda os seus desejos.

Debian

Debian é pronunciado "débien" que vem do nome de seu criador, Ian Murdock, e sua esposa, Debra. Essa distribuição é conhecida pela sua grande estabilidade conseguida por um rigoroso esquema de testes e também por estar profundamente ligada a filosofia GNU. Isso é conseguido com o trabalho de aproximadamente 1000 desenvolvedores espalhados pelo mundo. Poucos se conhecem pessoalmente, tudo é decidido em listas de discussão[Debian1], por email e reuniões em chat IRC (canal #debian em irc.debian.org).

A página oficial do Projeto Debian é <http://www.debian.org>, podendo ser baixado no link <http://www.debian.org/releases/stable/>.

Kurumin

Kurumin é uma distribuição GNU/Linux nacional, baseada no knoppix e no Debian, que roda a partir do CD (sem instalar no disco rígido), mas também pode ser instalada no HD. Bastante amigável, é ótima como "Meu primeiro GNU/Linux".

A página oficial do Kurumin é <http://www.kurumin.org>.



debian



Kurumin

Conectiva

Conectiva Linux é uma distribuição nacional baseada originalmente no RedHat, mas que já se diferenciou o suficiente para ter sua "personalidade" própria. Essa distribuição foca o mercado latino-americano e para isso correu atrás para ter suporte à maioria do hardware normalmente comercializado por aqui.



Slackware

Slackware é uma criação de Patrick Volkerding, tendo surgido em 1993, o que faz dela uma das primeiras distribuições GNU/Linux. Essa distribuição visa a estabilidade e facilidade de uso, embora nem todos a considerem fácil de se usar.



slackware
linux

Um dos preceitos do Slackware é ser a distribuição GNU/Linux mais parecida com o UNIX, fazendo dela uma distribuição bastante personalizável, uma vez que quase não possui ferramentas automáticas de configuração como as outras distribuições.

Embora a maior parte das pessoas ache que para instalar um programa no Slackware seja necessário compilar seu código fonte (embora isso seja sempre possível) a distribuição vem com vários programas compilados e prontos para usar, incluindo o XFree86 e o KDE.

Interfaces Gráficas ou Gerenciadores de Janelas

O que são "Interfaces Gráficas ou Gerenciadores de Janelas"?

Após a popularização do sistema operacional Microsoft Windows, o conceito de janelas parece ser óbvio num primeiro instante. Mas para aqueles que acompanharam o processo de amadurecimento das interfaces gráficas a importância dessa tecnologia é evidente do que aparenta ser. Antes de continuar, consideremos que interfaces gráficas é o que visualizamos na nossa tela de computador, isto é, a aparência, representada por cores, formatos de botões, janelas e menus, etc. Gerenciadores de janelas são programas que controlam a interação entre esses diversos elementos além de permitir a sua direta manipulação pelos usuários, como fechar, minimizar, arrastar janelas, abrir menus, executar programas, etc. Os dois conceitos portanto possuem uma estreita relação entre si, já que enquanto um controla a aparência o outro controla a manipulação, e muitas vezes eles compõem uma peça só dentro do sistema.

No Windows da Microsoft o gerenciador de janelas não é uma peça separada do próprio sistema operacional, algo que ocorre quando vislumbramos o