



Monografia

O software livre como alternativa ao aprisionamento tecnológico imposto pelo software proprietário

Aluno: Nicolas Michel Bacic
Orientador: Luis Antonio T. Vasconcelos

O software livre como alternativa ao aprisionamento tecnológico imposto pelo software proprietário

Monografia realizada em 2003

Agradecimentos

Ao orientador Prof. Luiz Antonio Teixeira Vasconcelos, pelo acompanhamento, sugestões e discussões.

A professora Maria Carolina de Azevedo Ferreira de Souza pelas sugestões.

Ao Luis Dosso, da Dextra Sistemas, pela entrevista, tempo disponibilizado e material enviado.

Ao Capitão Wildison Jone Jobim de Souza, Chefe da Divisão de Tecnologia da Informação da Escola Preparatória de Cadetes do Exército, por ter contado sua experiência com software livre e pelo tempo que disponibilizou para realizar a entrevista.

Ao meu pai pela sugestões e críticas.

A Adriana Ruggeri pela paciência nestes quase seis anos.

Ao meu irmão pelos raros momentos de silêncio que conseguiu proporcionar.

A minha mãe e a todos que direta ou indiretamente contribuíram para a realização deste trabalho.

Resumo

Hoje qualquer que seja a instituição pública ou privada necessita utilizar programas de computadores para auxiliar os seus processos, gerando uma enorme demanda por softwares. No entanto, a aquisição destes programas não é barata, tornando-se inviável a aquisição ou a atualização do software para diversas organizações. Cabe observar que o software é um componente de peso no custo dos computadores e, por exemplo, a implantação ampla de computadores em toda a rede de escolas públicas no Brasil demandaria um alto investimento em software.

Com o surgimento do movimento de software livre em 1984 com o projeto GNU (Gnu is Not Unix) e o amadurecimento do mesmo, o software livre tornou-se uma opção tanto para o setor público como privado. Sendo assim, este estudo visa identificar e explicar os fatores que condicionam a expansão do software livre, avaliando os fatores que dificultam e os que favorecem sua adoção.

São avaliadas as vantagens da adoção deste tipo de programa para as empresas e governo. Para isso são estudados casos reais da adoção de softwares livres por empresas como o Metrô de São Paulo e as Forças Armadas Brasileira. O trabalho estuda as restrições existentes à adoção do software livre a partir de uma reavaliação da teoria do aprisionamento tecnológico.

Palavras-Chave: aprisionamento tecnológico, software livre, projeto GNU, avaliação de software, feedback positivo, software, software proprietário, migração, custo de mudança.

Sumário

Introdução.....	3
Justificativa.....	5
Capítulo 1 – Software proprietário e Livre.....	7
Conceito de Software Livre e Proprietário.....	7
Sistema Operacional.....	9
Histórico do Software Livre.....	9
Licenças de Software Livre.....	15
Licença GNU.....	16
Licença BSD.....	17
Capítulo 2 – Estratégia Competitiva na Economia da Informação.....	18
Aprisionamento Tecnológico.....	21
Compromissos contratuais.....	22
Compras de bens duráveis.....	23
Treinamento para uma marca específica.....	25
Informações e Banco de Dados.....	26
Fornecedores Especializados.....	28
Custos de busca.....	28
Programas de lealdade.....	29
Incompatibilidade de sistemas.....	30
Comprometimento da cadeia com determinado padrão.....	31
O Ciclo do Aprisionamento.....	31
Feedback Positivo.....	32
Estratégia da Evolução.....	36
Estratégia da Revolução.....	37
Consolidação de padrão: o projeto dominante.....	37
Aprisionamento voltado ao software.....	40
Capítulo 3 – Aprisionamento x Software Livre no estudo da migração.....	43
Pacotes de escritório.....	43
Editor de Texto.....	44
Programa para Apresentação.....	46
Planilha Eletrônica.....	47
Considerações finais sobre os pacotes de escritórios.....	49
Banco de Dados.....	51
Sistema Operacional.....	52
Servidor de WEB.....	54
Custos do Aprisionamento.....	54
Considerações finais.....	57
Capítulo 4 – Análise de casos de adoção e migração do software livre.....	59
Companhia Do Metropolitano De São Paulo - METRÔ.....	59
Forças Armadas.....	62
Exército.....	62

Aeronáutica.....	64
Marinha.....	65
Viação Graciosa	67
Amazon.com	68
Toyota	68
Prefeitura de Rio das Ostras	69
Conclusão	71
Bibliografia.....	73
Anexos	80
Anexo 1 – Lei do Software	80
Anexo 2 – Softwares similares	87

Introdução

Segundo pesquisa realizada pela Sociedade para Promoção da Excelência do Software Brasileiro (SOFTEX) em conjunto com o Massachusetts Institute of Technology (MIT), o Brasil é o sétimo mercado de software no mundo, com vendas de US\$ 7,7 bilhões em 2001, rivalizando com a Índia e a China, com respectivamente US\$ 7,9 e US\$ 8,2 bilhões. Entre 1991 e 2001, a participação do segmento no PIB triplicou, alcançando 0,71%. (MIT- SOFTEX, 2002).

O setor de produção de software tem características diferenciadoras, no que se refere ao comportamento do ciclo de produto¹, em relação a outros setores. Na maioria dos setores, o peso dos investimentos na etapa correspondente ao ciclo operacional (aquisição dos insumos e produção dos produtos) é importante se comparado com o valor dos investimentos realizados nas outras etapas do ciclo do produto (planejamento e desenvolvimento do produto). No setor de software, conforme Vasconcelos (1999) as etapas de planejamento e desenvolvimento são bem mais importantes, em termo de investimento de recursos que a etapa produtiva (entendendo esta etapa, como a geração da cópia do produto a ser vendido). Afirma Vasconcelos, “quando se atinge o auge das vendas de um determinado produto, os gastos de produção característicos - insumos materiais, mão de obra direta, processamento, etc. - são, em geral, residuais em relação ao total de gastos de planejamento e desenvolvimento do produto em questão”. Este fato coloca importantes desafios para as empresas de software, quanto às estratégias a serem seguidas para a gestão de seus negócios, dado o peso importante dos investimentos iniciais em relação aos investimentos da etapa produtiva.

¹ Conforme Vasconcelos (1999) o ciclo de vida do produto, se “inicia com o planejamento e desenvolvimento de novos produtos e se encerra com o seu descarte (retirada de linha), passando pelo projeto técnico de produtos, processos, máquinas, equipamentos, ferramentas instalações, dispositivos de apoio, desenvolvimento e teste do produto, pelo processo de produção e de vendas corrente (ciclo operacional) e pelo apoio ao cliente ou usuário através de eventuais atividades de manutenção ou assistência técnica”.

Outra característica diferenciadora é observada pelo lado da demanda. Os compradores do software incorporam o produto a seus processos e rotinas e passam, em muitos casos, a ter uma situação de forte dependência em relação ao software escolhido. Esta situação se dá por diversos aspectos, por exemplo, em função do aprendizado, ou por problemas de compatibilidade que devem ser mantidos entre distintos softwares que trabalham trocando informações. Neste caso, tendem a ficar “presos” ao fornecedor inicial e tem um custo significativo (comparativamente a outros produtos) para mudar de fornecedor, dado que devem superar um conjunto de custos relativos ao aprendizado do uso do novo software e a superação dos problemas de compatibilidade. Esta situação de aprisionamento, tende a gerar vantagens para as empresas melhor sucedidas, que passam a usufruir de uma situação de quase monopólio, com os usuários sendo induzidos a acompanhar suas modificações de produto, sem terem condições de trocar por outros produtos, oferecidos por outras empresas.

Verifica-se assim, um interessante paradoxo, existindo uma oferta abundante e de qualidade de software de baixo custo ou gratuito - o software livre-empresas, organizações públicas e privadas e usuários finais, têm dificuldade de abandonar modalidades de softwares pagos e migrar para o software livre. Quais são os fatores que dificultam esta migração? Que fatores podem ajudar na migração?

Este estudo visa identificar e explicar os fatores que condicionam a expansão do software livre, avaliando os fatores que dificultam e os que favorecem sua adoção.

Serão estudadas as restrições existentes a adoção do software livre a partir da teoria do aprisionamento tecnológico. Serão avaliadas as vantagens da adoção deste tipo de programa para as empresas e governo. Para isso serão estudados casos reais da adoção de softwares livres por empresas como o Metrô de São Paulo e a Amazon.com.

Justificativa

Hoje qualquer que seja a instituição pública ou privada necessita utilizar programas de computadores para auxiliar os seus processos, gerando uma enorme demanda por softwares. No entanto, a aquisição deste programas não é barata, tornando-se inviável a aquisição ou a atualização do software para diversas organizações. Cabe observar que o software é um componente de peso no custo dos computadores e, por exemplo, a implantação ampla de computadores em toda a rede de escolas públicas no Brasil demandaria um alto investimento em software.

Como o software, freqüentemente, é fundamental e indispensável para o desenvolvimento dos processos – básicos ou não – executados e/ou administrados pelas organizações, estas podem acabar decidindo utilizar o programa mesmo sem obter sua devida licença, incorrendo assim na pirataria de software, que é uma prática ilegal. Com a intensificação do combate a pirataria, esta prática pode sair cara, como mostra a Lei de Software, Lei nº 9.609, de 19 de fevereiro de 1998. Assim o artigo Art. 12º da lei enuncia que quem violar direitos de autor de programa de computador terá pena de detenção de seis meses a dois anos ou multa e se a violação consistir na reprodução, por qualquer meio, de programa de computador, no todo ou em parte, para fins de comércio, sem autorização expressa do autor ou de quem o represente a pena será de reclusão de um a quatro anos e multa².

Para não incorrer nesta prática ilegal, as organizações tem dois caminhos. O primeiro consiste em pagar as devidas licenças de uso. O segundo caminho consiste em buscar no software livre uma alternativa de qualidade e de baixo custo.

A opção pelo software livre se justificaria pelas seguintes e principais razões:

² Ver anexo 1 (Lei de Software – Capítulo V).

- Ser considerado, por especialistas, de qualidade igual ou superior aos softwares pagos.
- Poder ser instalado em quantas máquinas forem necessárias sem ter pagar pela licença.
- Permitir acesso ao código fonte, o que possibilitaria realizar programas de melhor qualidade e realizar alterações que visem adequar melhor o programa as necessidades do usuário.
- Oferecer grande variedade de programas em quase todos os campos de aplicação de Softwares.

Por outro lado, a utilização do software livre pelas instituições poderia permitir uma melhora no Balanço de Pagamentos, já que os gastos com os softwares proprietários, via aquisição de licenças de uso, resultam, segundo o Banco Central (2003), em uma saída na ordem de US\$ 1,12 bilhão, anuais. Ainda, a utilização dessa categoria de software pelas instituições públicas permitiria uma melhora nas contas públicas, possibilitando, por exemplo, nas escolas públicas, que fossem utilizados os recursos não gastos em Software para a ampliação do número de computadores disponíveis aos estudantes e professores, que utilizariam softwares livres. Evidentemente, que se a decisão é tomada num momento em que o parque computacional instalado é pequeno, os eventuais custos de transição serão menores que quando se trate de mudar um parque amplo e já instalado.

Capítulo 1 – Software proprietário e Livre

Conceito de Software Livre e Proprietário

O conceito de software livre foi desenvolvido, inicialmente, por Richard Stallman no início dos anos 80. Tudo começou quando Richard trabalhava no Laboratório de Inteligência Artificial do MIT e a impressora que ele utilizava quebrou, sendo necessário adquirir uma nova impressora.

Para continuar a realizar seu trabalho precisava conhecer o funcionamento desta nova impressora, no entanto suas solicitações ao fabricante para que este liberasse os códigos fontes do equipamento foram negadas. A partir deste momento Stallman passou a pensar em uma forma de tornar acessíveis os programas e códigos, surgindo a idéia de software livre.

A filosofia do software livre está baseada em quatro conceitos básicos:

- A liberdade de executar o programa, para qualquer propósito.
- A liberdade de estudar como o programa funciona, e adaptá-lo para as suas necessidades. O acesso ao código-fonte é um pré-requisito para esta liberdade.
- A liberdade de redistribuir cópias de modo a ampliar as possibilidades de acesso de pessoas e instituições a tais programas.
- A liberdade de aperfeiçoar o programa, e liberar os seus aperfeiçoamentos, de modo que toda a comunidade se beneficie, sem gastos adicionais.

Esses conceitos buscam garantir que o usuário possa executar, copiar, estudar, modificar o software, visando sempre à liberdade de produção (ou

aperfeiçoamento) e utilização. Software livre pressupõe a liberdade de utilização, por isso é melhor evitar termos como “de graça” ou “doador”, pois não se trata de uma questão de preço, mas sim de liberdade de acesso ao código-fonte e de uso. Um usuário pode redistribuir as cópias do software de maneira gratuita ou cobrando uma taxa pela distribuição, as pessoas possuem livre escolha para decidir como distribuir um programa, alterá-lo, etc., sem ter que pedir permissão a ninguém. Cabe observar que esta possibilidade de cobrança é marginal, não havendo obrigação de pagar as licenças de uso, assim sendo ao comprar um programa de um usuário que cobre pelo serviço de distribuição, o comprador passa a ter direitos de realizar o que quiser com o software, podendo alterá-lo e instalá-lo em outras máquinas sem que se tenha que pagar nada para ninguém.

Já o software proprietário é regido por uma série de normas que visam limitar o seu uso ao número de licenças adquiridas. São estabelecidas licenças pelas quais é necessário pagar por cópia instalada, não sendo permitida a alteração do código; ademais, é ilícita a livre distribuição do programa e normalmente não se tem acesso ao código fonte destes programas.

Como a grande maioria dos softwares proprietários não possui o código aberto não é possível personalizar o programa, verificar a qualidade do código, realizar melhorias no programa e corrigir erros.

Os softwares livres mais conhecidos são: o sistema operacional LINUX, o pacote de ferramentas para escritório OpenOffice, o editor de imagem GIMP, o navegador de Internet Mozilla, etc.

O fabricante de softwares proprietários mais conhecido é a Microsoft com programas como o Windows 95/98/NT/2000/XP, o pacote Microsoft Office, o navegador Internet Explorer, etc.

Sistema Operacional

O Sistema Operacional (SO) é o principal programa do computador, sendo de sua responsabilidade gerenciar e monitorar os recursos de um computador. É ele que "diz" ao computador como utilizar os periféricos e rodar os programas.

O SO transforma o computador em uma caixa preta para os usuários, uma vez que este simplifica o acesso aos recursos, exemplo: o usuário não precisa saber como funciona um disco rígido para conseguir gravar um arquivo nele, basta que acione o comando de copiar e o sistema fará isso automaticamente, a partir de instruções e procedimentos pré-estabelecidos e fixos. Além disso, o SO coordena os diversos processos, evitando que eles se sobreponham, exemplo: ao mandar imprimir dois arquivos ao mesmo tempo o SO faz com seja impresso cada um deles por vez e não a linha do primeiro, depois a do segundo, etc.

Sem o Sistema Operacional não seria possível, ou pelo menos seria muito complicado, realizar operações simples. Por esta razão o SO é fundamental e indispensável para a utilização de um computador.

Histórico do Software Livre

Neste item será feito um breve histórico do Software Livre baseados nos autores: ASCENSO e SANTOS (2003), HAUBEN (2003), LOTTERMANN (2001), STALLMAN (1996), TANENBAUM (1994) e UNIXSUPPORT (2003).

Durante a criação da Arpanet³ (*Advanced Research Projects Agency Network*), que foi o ponto inicial para o aparecimento da Internet, os desenvolvedores dos Laboratórios Bell e da American Telephone and Telegraph

³ Rede precursora da Internet criada no final dos anos 60 nos Estados Unidos. Foi desenvolvida no meio acadêmico e financiada pelo governo americano. A Arpanet foi a primeira experiência de uma rede de longa distância e tinha como objetivo continuar funcionando mesmo que parte da rede fosse destruída, por exemplo, por um ataque nuclear.

(AT&T) sentiram a necessidade de um sistema para controlar os diversos processos e recursos “in-house computer” controlados por eles (ASCENSO e SANTOS, 2003).

A Bell então desenvolveu o sistema operacional BESYS (*Bell Operating System* – Sistema Operacional da Bell) para ser utilizado como seu computador. No entanto o sistema foi requisitado por outras pessoas e a Bell optou por fornecer o software de maneira gratuita, mas sem suporte técnico. Caso houvesse necessidade de suporte, este deveria ser contratado.

Em 1964, os Laboratórios Bell, quando da adoção da terceira geração de computadores, tiveram que decidir se utilizariam um sistema operacional criado por terceiros ou se desenvolveriam novo sistema operacional. Após deliberar internamente a empresa optou criar um novo sistema, o Multics (Multiplexed Information and Computing Service), unindo forças com a General Electric e com o Massachusetts Institute of Technology (MIT).

No entanto em 1969 a AT&T optou por retirar-se do projeto, por entender que o desenvolvimento do Multics se daria em um prazo longo demais e a um custo excessivo.

Ken Thompson, Dennis Ritchie e outros colegas que haviam participado do desenvolvimento do Multics, resolveram escrever uma versão simplificada do Multics usando um computador PDP-7 da Digital Equipment Corporation que estava ocioso. Foi implementado um sistema de arquivos e um interpretador de comandos digitados pelos usuários que ganhou o nome de *Shell*. O sistema funcionou bem, suportando os trabalhos de desenvolvimento. A partir daí o sistema ganhou nome de Uniplexed Information and Computing Service (Unics), uma sátira ao sistema operacional Multics e posteriormente teve seu nome mudado para Unix (HAUBEN, 2003).

Em 1970, a equipe de desenvolvimento de Thompson adquiriu um computador PDP-11 da Digital para criar um sistema de editoração de textos. Dennis Ritchie alterou a linguagem B, desenvolvida por Thompson, dando origem à linguagem C, que foi utilizada para reescrever o Unix.

Thompson e Ritchie publicaram um artigo sobre o Unix em 1974, o que incentivou que muitas universidades solicitassem uma cópia do software a AT&T. Como na época a empresa era um monopólio controlado na área de telecomunicações, não podia atuar na área de computação. Sendo assim, a AT&T forneceu sem empecilhos para aquisição do programa para as universidades, que puderam contar com o código fonte completo⁴ para estudar e melhorar o programa.

A partir de então se criaram encontros científicos em torno do Unix, onde surgiram novas idéias e melhoramentos que foram rapidamente absorvidos pelo sistema. Esse grande dinamismo só foi possível por que na época compartilhar programas de computador e códigos fontes dos programas, era uma prática comum, o que permitia que se desenvolvessem estudos e fossem introduzidas melhorias nos programas. Por outro lado, esta prática fez com surgissem diversos sistemas operacionais baseados em Unix.

Na década de 80 foi criado um ambiente gráfico, o X-Windows, que se tornaria padrão em todas as distribuições Unixes e o protocolo livre TCP/IP (Transmission Control Protocol/Internet Protocol – Protocolo de Controle de Transmissão/Protocolo Internet), criado pela DARPA (Defense Advanced Research Projects Agency), foi utilizado pelo departamento de sistemas da universidade de Berkeley para comunicação a distância entre sistemas operacionais Unixes, modelo que viria a se tornar padrão na Internet.

Nos anos 80 o sistema Unix já era amplamente utilizado, pois sua portabilidade de uma máquina para outra era simples. Além disso, muitas empresas e instituições contribuíam para melhorar o sistema operacional. No entanto em 1984 o governo norte-americano dividiu a AT&T em várias companhias independentes, permitindo a ela criar uma subsidiária que operasse no ramo de informática, levando o Unix a ter código fonte fechado e que se passasse a cobrar pelo mesmo.

⁴ Conjunto de comandos e instruções escritos pelo programador para criação do software

Neste mesmo período ocorreu a popularização dos computadores pessoais (PC), onde já não existiam mais sistemas com código aberto, fazendo com que as pessoas se acostumassem com os softwares proprietários.

Ainda no ano de 1984, Richard Stallmann que trabalhava na MIT, decidiu criar um sistema operacional que qualquer pessoa poderia copiar, usar, modificar e distribuir. Stallmann queria ter uma comunidade de desenvolvedores trabalhando juntos novamente, o que tornava imprescindível que o sistema possuísse código fonte aberto (LOTTERMANN, 2001).

Segundo Richard, o desenvolvimento de um software deveria ocorrer de forma evolucionária, no qual um programador pudesse, a partir de um programa já existente, adicionar melhorias, novas funcionalidades, novos recursos e até mesmo criar novos programas sem a necessidade de se começar do zero. A partir desse momento Stallmann inicia o projeto e conceito do software livre.

Optou-se por produzir um sistema operacional compatível com o Unix, pois desta forma a migração de usuários Unix para o novo sistema seria facilitada. O nome dado para o projeto foi GNU (Gnu Is Not Unix – Gnu não é Unix).

Foi criado um novo tipo de licença, na qual qualquer pessoa poderia copiar, executar, alterar, modificar e distribuir o programa, desde que o mesmo permaneça livre. Esta nova licença foi chamada de copyleft, uma paródia com o termo copyright que mantém os direitos autorais. O copyleft permitiria o sistema operacional GNU obter vantagem social sobre o Unix, mesmo se o mesmo não apresentasse vantagens técnicas.

A fim de economizar recursos, os desenvolvedores do GNU optaram por integrar programas que fossem livres, mesmo que não fossem regidos pela licença GNU; exemplos disso são a utilização do sistema de janelas desenvolvido pelo MIT, o X Windows e o protocolo de comunicação TCP/IP, desenvolvido pelo departamento de defesa dos Estados Unidos da América, etc.

No mesmo ano de 1984 Stallmann saiu do seu trabalho, mas conseguiu acesso à estrutura do laboratório de Inteligência Artificial do MIT para o desenvolvimento do sistema GNU. Inicialmente desenvolveu o editor de textos Emacs e em seguida o compilador gcc (Gnu C Compiler), pois para se criar um

programa é necessário um editor onde se possa escrever o código e um compilador para transformar o código em um arquivo executável.

No ano de 1985 ele criou o FSF (Free Software Foundation – Fundação Software Livre) para o desenvolvimento e manutenção do sistema operacional GNU, cuja receita provem de doações e das vendas de cd-rom com software livres.

Inicialmente a fundação desenvolveu o interpretador de comandos bash (Bourne Again Shell), cujo nome surgiu de mais uma brincadeira dos programadores com o interpretador de comandos do Unix (Bourne Shell), desenvolveu diversos programas que poderiam ser rodados em Unixes proprietários, mesmo sem existir o sistema GNU completo. Finalmente em 1990 o sistema GNU estava praticamente completo, no entanto faltava desenvolver o kernel (núcleo), programa que interage diretamente com o hardware do computador, uma parte essencial para qualquer sistema operacional. O núcleo que está sendo desenvolvido pelo GNU foi chamado de HURD e atualmente se encontra na versão 0.2.

Em 1991, Linus Torvalds, do Departamento de Ciência da Computação da Universidade de Helsinki na Finlândia, criou um núcleo compatível com o Unix (utilizando o gcc), baseado no Minix, um sistema operacional desenvolvido pelo professor Andy Tannenbaum para fins educacionais. Após a criar a versão 0.02 do núcleo, que recebeu o nome de Linux, Linus disponibilizou o código fonte na Internet, pedindo que programadores o ajudassem a melhorar o programa.

Em 1992, o kernel Linux foi incorporado aos programas GNU, gerando um sistema operacional livre e completo, que recebeu o nome de GNU/Linux, embora hoje seja mais conhecido como apenas Linux. O GNU/Linux executa diversos programas Unix e continua evoluindo como um sistema operacional.

Em 1996, Richard Stallman escreveu um artigo sobre a relação Linux e o projeto GNU e avaliando a idéia de software livre.

(...)

O projeto GNU não é somente desenvolvimento e distribuição de alguns softwares livres úteis. O coração do projeto GNU é uma idéia: que software deve ser livre, e que a liberdade do usuário vale a pena ser defendida. Se as pessoas têm liberdade, mas não a apreciam conscientemente, não irão mantê-la por muito tempo. Se quisermos que a liberdade dure, precisamos chamar a atenção das pessoas para a liberdade que elas têm em programas livres.

O método do projeto GNU é que programas livres e a idéia da liberdade dos usuários ajudam-se mutuamente. Nós desenvolvemos software GNU, e conforme as pessoas encontrem programas GNU ou o sistema GNU e comecem a usá-los, elas também pensaram sobre a filosofia GNU. O software mostra que a idéia funciona na prática. Algumas destas pessoas acabam concordando com a idéia, e então escrevem mais programas livres. Então, o software carrega a idéia, dissemina a idéia e cresce da idéia.

(...)

Nós devemos continuar a falar sobre a liberdade de compartilhar e modificar software -- e ensinar outros usuários o valor destas liberdades. Se nós nos beneficiamos por ter um sistema operacional livre, faz sentido para nós pensar em preservar estas liberdades por um longo tempo. Se nós nos beneficiamos por ter uma variedade de software livres, faz sentido pensar sobre encorajar outras pessoas a escrever mais software livre, em vez de software proprietário.
(Stallman, 1996)

O processo de criação de um software livre se inicia com o surgimento da primeira versão do código, com sua posterior publicação pelo autor ou coordenador, na Internet. Alguns usuários e também desenvolvedores de software melhoram o código e o retornam ao coordenador. Este absorve as alterações e reinicia o ciclo.

Com o aumento do tamanho dos projetos surgiram sites centralizadores do softwares livres entre eles The Apache Group, Mozilla.org, Openoffice.org, Linux Online. Nestes sites os programadores podem transferir os códigos fontes, comunicar erros e falhas, apresentar correções e novos recursos, trocar experiências e realizar documentação.

Licenças de Software Livre

Para se definir as regras para utilização de um software são utilizadas licenças, por exemplo, caso se deseje proteger os direitos autorais utiliza-se o copyright, assim como se podem colocar diversas restrições ao uso de determinado software.

As licenças de software livre visam garantir a possibilidade de alteração e distribuição do software, definindo as regras de utilização. Ao longo tempo diversas licenças surgiram, apresentando de forma geral as seguintes características:

- Proteger a identidade do autor;
- Distribuir os códigos fontes;
- Qualquer trabalho que inclua parte do software deve citar o autor;
- Instalação do software em um número irrestrito de computadores;
- Não restringir a venda do software;

A seguir serão mostrados os tipos mais relevantes de licenças para softwares livres.

Licença GNU

O projeto GNU procura garantir a liberdade sobre o software, para isso existem alguns tipos de licença criados pelo projeto.

Segundo Richard Stallman, o termo *free software*, pode causar confusão ao ser interpretado como um software gratuito, para ele o termo *free* não está relacionado ao preço, mas sim a liberdade de utilização do software. Ou seja, o programa pode ser utilizado sem restrições, modificado, melhorado, redistribuído gratuitamente ou cobrado.

Para alcançar os objetivos acima, existe a licença GNU GPL (*General Public Licence* – Licença Pública Geral), que garante acesso ao código fonte do programa, sendo assim possível personalizá-lo ou modificá-lo. O software pode ser distribuído com ou sem custo, mas, ao contrário do software comercial, não existem restrições para sua utilização, podendo o comprador utilizá-lo em um número indeterminado de máquinas, redistribuí-lo, modificá-lo ou até mesmo cobrar pela cópia. Entretanto, caso algum pedaço de um software licenciado pelo GNU seja utilizado, obrigatoriamente o novo programa deverá ser licenciado conforme a licença GPL.

No entanto, existe uma alternativa para que bibliotecas⁵ do projeto GNU possam ser utilizadas em um software comercial, é a licença LGPL (*Library General Public License* – Licença Pública Geral de Biblioteca). Através dessa licença, as bibliotecas desenvolvidas pela GNU podem ser livremente utilizadas em aplicações comerciais, tendo como única restrição que o programa seja capaz de aceitar atualizações das bibliotecas livres.

A GNU ainda possui uma licença especial para documentação, a GNU FDL (*Free Documentation License* – Licença para documentação livre), que visa assegurar que documentos, manuais, livros sejam livres, sendo permitido a alteração, cópia e redistribuição de forma gratuita ou comercial.

⁵ Rotinas executáveis dos softwares que podem ser executadas a parte do programa principal. Com isso uma mesma biblioteca pode ser utilizada por diversos softwares, sem a necessidade que a mesma se repita.

Licença BSD

A licença BSD (Berkeley Software Distribution), foi criada pela Universidade de Berkeley que desenvolveu o seu próprio sistema operacional Unix quando o mesmo passou a ser comercializado pela AT&T. Esta licença impõe poucas limitações, tendo como objetivo disponibilizar o desenvolvimento do software para a sociedade e ao mesmo tempo permitir que um financiador privado faça uso da pesquisa pra seus fins proprietários. Qualquer pessoa pode alterar o programa e comercializá-lo como se fosse de sua autoria. Esta licença é bastante utilizada pelos desenvolvedores comerciais como DEC e Sun, no entanto, existem softwares gratuitos que o utilizam esta licença tais como os sistemas operacionais FreeBSD e OpenBSD e os programas BIND, o Apache e o Sendmail.

Capítulo 2 – Estratégia Competitiva na Economia da Informação

Segundo Porter (1980) a rentabilidade de um determinado setor (ao qual denomina indústria) depende da resultante da atuação de cinco forças competitivas. Essas forças atuam aumentando ou contraindo (segundo sejam favoráveis ou desfavoráveis) a rentabilidade que, potencialmente, surgiria do valor que os concorrentes da indústria oferecem aos compradores. As forças configuram uma determinada estrutura e determinam qual parte capta a rentabilidade do valor criado pelas empresas concorrentes para os clientes (as próprias empresas, os compradores ou os fornecedores). Se a estrutura industrial não é atraente, o valor criado pode dissipar-se em forma de preços mais baixos ou custos adicionais de concorrência (mais serviços, mais garantias, mais propaganda, etc).

As cinco forças são: a ameaça de novas entradas, a intensidade da rivalidade entre concorrentes existentes, o poder de negociação dos fornecedores, o poder de negociação dos compradores e a ameaça dos produtos substitutos.

1. **Ameaça de novos competidores:** com a entrada de novos competidores na indústria, a disputa torna-se mais acirrada para os atuais competidores. Barreiras de entrada altas podem desestimular a entrada de novos concorrentes.
2. **Rivalidade entre os competidores da indústria:** o padrão de rivalidade entre competidores afeta o resultado obtido por cada empresa e pela totalidade das empresas concorrentes.
3. **Poder de negociação dos fornecedores:** os fornecedores podem impor preços maiores afetando a estrutura de custos da indústria.

4. **Poder de negociação dos compradores:** os clientes podem absorver a rentabilidade das empresas, forçando preços menores e/ou exigindo mais serviços adicionais.
5. **Ameaça dos produtos substitutos:** produtos substitutos impõem limites à capacidade das empresas da indústria para fixar preços e impor condições

Cada uma das cinco forças é composta por uma série de variáveis (os determinantes estruturais da concorrência) que são elementos de natureza técnica e econômica, os quais compõem a estrutura industrial.

As barreiras de entrada são função de fatores tais como: economias de escala, grau de diferenciação de produtos, custos de mudança de fornecedor, volume de capital, necessidade de penetração nos canais de distribuição, tecnologia, patentes, importância da experiência acumulada, políticas governamentais. A possibilidade de represálias dependerá da percepção que as firmas estabelecidas tiverem da ameaça com a nova entrada e de seu poder de sustentar políticas agressivas, com a finalidade de impedir que o entrante alcance seus objetivos.

Os custos de mudança são uma importante barreira à entrada, pois fazem com que o comprador se defronte com custos ao tentar mudar de fornecedor. Esses custos são formados por: treinamento, custo de novo equipamento, custo para testar e configurar uma nova estrutura, custos psicológicos, etc. Assim sendo, um novo fornecedor deve criar uma série de atrativos e melhorias para que se torne vantajoso ao comprador mudar de fornecedor.

A intensidade da rivalidade entre concorrentes depende de um conjunto de fatores: número, ausência de uma empresa líder, custos fixos altos, ausência de diferenciação de produto e/ou custos de mudanças, necessidade técnica de aumento de capacidade em grandes incrementos, divergências estratégicas e barreiras de saída altas. O aumento de rivalidade leva a diminuições de preços e/ou aumento desnecessário de custos.

O poder de negociação dos compradores é função de fatores tais como: volume comprado, concentração da indústria compradora, produto padronizado ou

diferenciado, existência ou não de custos de mudança, informação dos compradores sobre aspectos estratégicos do setor. Compradores com alto poder de negociação podem diminuir a rentabilidade das empresas do setor.

Os fornecedores podem atuar, pressionando a rentabilidade para baixo. Seu poder de negociação depende da concentração do setor fornecedor, do grau de diferenciação do produto, do desenvolvimento dos custos de mudança, da inexistência de produtos substitutos, da importância relativa da indústria compradora para os fornecedores.

Os produtos substitutos podem pressionar a rentabilidade para baixo, pois, independentemente da ação das outras forças, estes impõem um limite aos preços que as empresas da indústria podem cobrar.

As cinco forças causam impacto na rentabilidade da indústria, uma vez que os preços, os custos e o investimento estão sujeitos à influência dos elementos da estrutura da indústria. As estruturas favoráveis à manutenção do valor criado são aquelas nas quais há um número reduzido de empresas, existe claramente um líder, o produto é diferenciado e/ou há custos de mudanças e barreiras de entrada.

As recomendações de Porter (1980) são que uma empresa individual deve procurar uma posição de domínio e isolamento competitivo para captar a maior parcela possível de rentabilidade. Assim, deve procurar diferenciar produtos, gerar custos de mudança, intensificar os fatores que favorecem a expansão das barreiras de entrada, procurar vantagens tecnológicas e de custo. Ao mesmo tempo deve fugir de estratégias semelhantes que lhe sejam impostas por eventuais fornecedores, ou seja, deve forçar a padronização e evitar ficar à mercê de custos de mudança gerados pelos fornecedores.

A empresa que consegue implantar custos de mudança em seus clientes, usufrui de significativa vantagem sobre suas concorrentes e entrantes potenciais, dado que estas deverão oferecer sensíveis vantagens aos clientes dessa empresa, para que abandonem os produtos/ serviços dos quais são dependentes ou aprisionados. A geradora dos custos de mudança passa a ter assim, um importante controle sobre seus mercados, tendo condição de administrar preços, o ciclo de vida de produto e as condições de comercialização.

Nos setores da tecnologia de informação (TI) a geração de custos de mudança tem um importante papel nas estratégias das empresas, tal como demonstrado por Shapiro e Varian (1999). A empresa presa em custos de mudança de TI, passa a sofrer, segundo esses autores de uma situação de aprisionamento tecnológico.

Aprisionamento Tecnológico

Ao optar por uma determinada tecnologia, uma organização se submete a determinadas restrições que as levam a ficar dependente da tecnologia escolhida, devido à dificuldade de troca da dessa tecnologia por uma outra. Este processo é denominado de aprisionamento tecnológico.

Estas dificuldades de mudança decorrem das incompatibilidades entre as tecnologias, que as levam a altos custos de mudanças. Por exemplo, um usuário de tecnologia de informação (TI) ao escolher um determinado software de computador poderá ter problemas ao tentar migrar para outro programa, devido à incompatibilidade entre os tipos de arquivos, impossibilidade de migrar diretamente os dados, necessidade de treinamento ao novo software, etc.

Assim, antes de mudar de um sistema ou tecnologia, é necessário levar em consideração que, normalmente, os custos de troca são significativos. Isto, muitas vezes, leva aos responsáveis pelo gerenciamento da informação dentro das empresas a optarem por continuar utilizando o sistema vigente, ou seja, leva a um aprisionamento aos sistemas já instalados.

Nem sempre o aprisionamento prevalece, existem determinados momentos nos quais a tecnologia torna-se obsoleta e precisa ser substituída. No entanto, os custos de mudança interferem nas estratégias e opções as empresas.

Para lidar melhor com o aprisionamento é necessário reconhecer seus diversos tipos e os custos de troca nos quais implicam, assim como seu ciclo. Os tipos de aprisionamento e custos de troca associados são mostrados na tabela abaixo:

Tabela 2.1 – Tipos de aprisionamento e custos de troca associados

Tipo de aprisionamento	Custos de mudança
Compromissos contratuais	Indenizações compensatórias ou liquidadas.
Compra de bens duráveis	O custo de substituição de equipamento tende a cair à medida que o bem durável envelhece.
Treinamento em marca específica	Aprender um novo sistema demanda tempo e incorre em custos, que tendem a aumentar com o tempo.
Informação e banco de dados	Conservação de dados para o novo formato. O custo tende a aumentar com o tempo, pois a quantidade de dados aumenta.
Fornecedores especializados	Financiamento de novo fornecedor -tende a ser maior quanto mais difícil for encontrar um novo fornecedor.
Custos de busca	Custos combinados do comprador e fornecedor – incluem o aprendizado sobre a qualidade das alternativas.
Programas de lealdade	Quaisquer benefícios perdidos do fornecedor, mais possível necessidade de reconstruir o uso cumulativo.

Fonte: SHAPIRO e VARIAN (1999)

Reconhecer esses padrões ajuda a identificar e antecipar o aprisionamento, avaliar os custos de troca e planejar uma estratégia para os efeitos deste processo.

Cabe observar que os padrões levantados por Shapiro e Varian, não se aplicam unicamente ao caso da TI e são válidos para um amplo espectro de situações. Porém, no caso de produtos baseados na TI é mais comum o surgimento de custos de mudança que levam a situações de aprisionamento.

Compromissos contratuais

Este tipo de aprisionamento surge do contrato estabelecido entre cliente e fornecedor, ficando o cliente restrito a comprar de um único fornecedor. Esta é a forma mais visível de aprisionamento.

Mesmo quando o contrato é assinado com preço pré-estabelecido é prudente evitar este tipo de compromisso, pois sempre podem existir margens

para aumento de preço, alteração da qualidade do produto/serviço oferecido, problemas com prazo de entrega, etc.

Assim, mesmo se o acordo estabelecido estiver regido por cláusulas que impeçam o aumento do preço do produto oferecido, o fornecedor pode explorar o aprisionamento através da exploração de outros fatores distintos ao preço, que lhe permitam obter maiores ganhos, por exemplo, piora de qualidade do produto, não cumprimento de prazos, etc.

O grau de aprisionamento que acontece através desta prática varia de contrato para contrato. Quanto mais rígido e restritivo for o contrato do fornecedor com o cliente, por exemplo, um contrato de exclusividade de compra, maior é o grau de aprisionamento. Um outro contrato que permita que o cliente possa utilizar outros fornecedores para suprir parte de suas necessidades, resulta em um nível menor de aprisionamento.

O custo de mudança que este aprisionamento impõe está relacionado com as cláusulas indenizatórias que o contrato possui. Assim sendo, o cliente deve procurar realizar contratos que garantam preço, qualidades, prazos e indenizações pequenas no caso de ruptura do contrato. No entanto, no momento em que o contrato finaliza atinge-se um nível de aprisionamento praticamente nulo.

Cabe ressaltar que o aprisionamento pelos compromissos contratuais pode acontecer também pelo lado fornecedor, por exemplo, um fornecedor que se compromete a fornecer unicamente para um único cliente. Por tanto, o fornecedor também deve evitar ou minimizar os possíveis danos que um contrato desse tipo possa ocasionar.

Compras de bens duráveis

Ao optar pela compra de determinado equipamento durável de uma dada marca, o usuário do produto enfrenta um novo tipo de aprisionamento, pois normalmente a compra de equipamentos duráveis exige grande quantidade de capital, o que muitas vezes impossibilita a troca do equipamento durável por outro em um curto período de tempo.

Além da dificuldade de troca do bem durável, o usuário que optou comprar o equipamento de determinada marca pode ficar "amarrado" através de acessórios fornecidos pelo fabricante; tal prática permite aos fornecedores conseguirem boa parte de seus lucros. Exemplo disso são as impressoras a jato de tinta, para as quais os fabricantes costumam cobrar barato pela impressora e caro pelo cartucho da mesma, fazendo com que muitas vezes a troca de dois ou três cartuchos se equipare com o preço da própria impressora. Ou também, tal como no caso da expansão de memórias para os mainframes, essa expansão é extremamente custosa para o cliente, dado que o fornecedor original sabe que representa a única alternativa de fornecimento, pois não existe um mercado de memórias padronizadas para mainframes. Cada fornecedor tem sua própria especificação.

O grau de aprisionamento ocasionado varia de acordo com o período de tempo que o produto leva para se depreciar, com a idade do produto, com a existência ou não de um mercado de produtos usados e com dinâmica da inovação de novos produtos. Assim sendo, um produto que se deprecie rapidamente ocasiona um nível pequeno de aprisionamento, já um equipamento que possua um longo período de depreciação traz um elevado nível de aprisionamento.

A existência de um mercado secundário que negocie o equipamento usado diminui o grau de aprisionamento, pois parte do custo da compra do bem pode ser recuperado, reduzindo-se o custo de troca.

Já a dinâmica da inovação de novos produtos pode diminuir o custo de mudança à medida que torna o equipamento (tanto hardware como software) obsoleto, forçando o cliente a atualizar seus produtos.

O grau de aprisionamento chega ao ponto mais baixo quando a maior parte dos equipamentos duráveis e seus complementos estão depreciados, sendo esse o melhor momento para que o cliente possa optar pela troca de fornecedor. A fim de aumentar este nível mínimo de aprisionamento o fornecedor pode incentivar a troca de produtos mais antigos por novos.

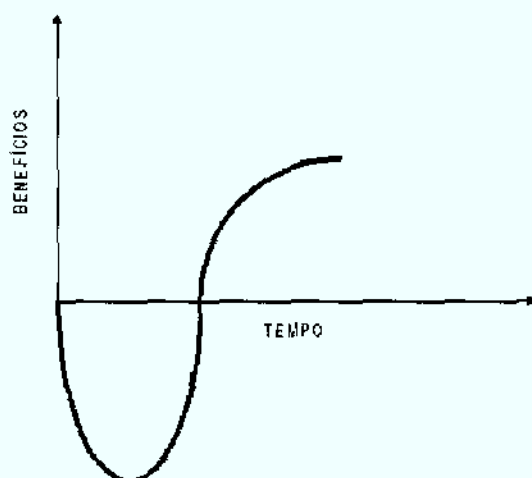
O cliente ao necessitar de um produto durável deve avaliar as necessidades do pós-vendas, como, por exemplo, verificar se existem outros fabricantes de

produtos adicionais. Na existência de diversos fornecedores alternativos o grau de aprisionamento diminui. Voltando ao caso da impressora a jato de tinta, o surgimento de fabricantes alternativos de cartuchos e cartuchos reciclados, diminui o nível de aprisionamento (a questão subjacente aqui é a sustentação do nível de qualidade compatível com a do suprimento original, pois dependendo do uso que se faça do equipamento não é possível alterar este nível sem incorrer em graves riscos competitivos). Além disso, o cliente ainda pode procurar alternativas para a compra do equipamento, tais como o aluguel ou o *leasing*.

Treinamento para uma marca específica

Este aprisionamento surge do treinamento para uma marca específica, o que dificulta a adoção de uma nova marca, pois significaria a necessidade de dispor de tempo e recursos para aprender a lidar eficientemente com o novo produto. A curva da Figura 2.1 ilustra o tempo necessário para que se obtenha algum benefício ao adotar uma nova tecnologia.

Figura 2.1 – Curva de Aprendizagem para novas tecnologias



Fonte: Deghi, 1999

Quanto maior for o tempo que os usuários estiverem acostumados a um produto de uma marca específica, maior será o custo de troca e conseqüentemente o aprisionamento.

Um exemplo claro disso é o aprendizado de um software específico, por exemplo, o Microsoft Windows que para ser aprendido inicialmente demanda uma certa quantidade de recursos, com o passar do tempo maior é familiaridade que o usuário tem com o programa, realizando as tarefas de maneira mais fácil e eficiente. Além disso, o Windows recebe constantemente novos recursos que podem ser facilmente aprendidos por usuários familiarizados com versões anteriores.

Varian e Shapiro colocam que o aprisionamento ocasionado por treinamento para uma marca específica pode durar mais do que o equipamento. Isso pode ser claramente visto quando uma empresa opta por adotar um software de uma marca específica para todo seu parque de computadores, buscando com isso diminuir os custos de aprendizagem e treinamento.

O grau de aprisionamento causado dependerá da similaridade que os produtos dos concorrentes tiverem com o produto específico. Quanto mais parecido for o produto do concorrente menor será o nível de aprisionamento enfrentado, pois será necessário um período menor de adaptação ao novo produto. Um exemplo disso foi à estratégia da Microsoft com a planilha Excel que manteve paralelamente à sua estrutura de menus a estrutura de comandos do Lotus 1-2-3, facilitando a migração da grande base de usuários que utilizavam o programa.

Informações e Banco de Dados

Ao possuir uma determinada quantidade de dados armazenados em um formato específico, torna-se um entrave migrar a estrutura de hardware e software para uma nova estrutura que poderia fornecer vantagens ao usuário.

O nível de aprisionamento será maior quanto maior for a dificuldade e os riscos de perda de dados em caso de transferência de dados de uma arquitetura para outra.

Este tipo de aprisionamento pode ser facilmente notado no passado quando surgiram os primeiros aparelhos de CDs de música, para os quais a migração da tradicional vitrola para o novo formato não ocorreu de maneira instantânea devido em grande parte ao fato que os LPs dos usuários não puderam ser utilizados nesse novo aparelho. Outro exemplo semelhante, um pouco mais recente, foi a transição do videocassete para o DVD, que somente se acelerou com o aumento de filmes em DVDs existente para venda e para alugar nas locadoras.

Uma forma que os fornecedores de novos produtos encontram para reduzir o custo de mudanças para seus clientes consiste em fazer com que a nova tecnologia aceite a tecnologia anterior, exemplo disso são os drivers de DVDs que são capazes de ler os CDs.

Assim o usuário ao adotar uma tecnologia ou determinado produto de um fornecedor deve levar em consideração a facilidade de transferência dos dados para novas tecnologias ou para produtos de outros fornecedores, visando assim minimizar o grau de aprisionamento.

Um exemplo de como um fornecedor pode reduzir o custo de mudança, facilitando que clientes potenciais adotem seu software é tornar o programa capaz de ler e gravar dados no formato dos concorrentes. Foi o que a Microsoft fez quando o mercado de editores de texto era dominado pelo WordPerfect.

Este tipo de aprisionamento tende a aumentar com o tempo, pois a quantidade de dados armazenados tende a crescer com o tempo. Uma forma de minimizar o aprisionamento no longo prazo é adotar formatos de armazenamento que sejam abertos, facilitando assim migrar de uma plataforma para a outra.

Fornecedores Especializados

Ao se adquirir produtos de um único fornecedor, o cliente fica dependente do fornecedor no futuro. Isso permite ao fornecedor usufruir vantagens na próxima rodada de compras.

Assim o cliente deve procurar comprar produtos que possam ser comprados de mais de um fornecedor e dar preferência para produtos que também possuam acessórios e suplementos compatíveis fornecidos por outras marcas. Por exemplo, a Epson não patenteia seus cartuchos havendo diversos fabricantes que produzem cartuchos compatíveis com as impressoras Epson.

O nível de aprisionamento varia de acordo com o número de fornecedores que existem para determinados produtos, o número de fornecedores de suplementos para os produtos e da capacidade do cliente em antever suas necessidades, buscando assim um melhor acordo com o fornecedor.

Custos de busca

Esta categoria envolve os custos que ocorrem ao se buscar um novo fornecedor ou cliente. Os custos que incidem sobre os clientes são fatores psicológicos, o tempo e os recursos necessários gasto na busca do novo fornecedor e riscos de se tratar com um novo fornecedor. Já pelo lado do fornecedor, este deve arcar com gastos promocionais, custo de fechar um novo negócio e riscos de negociar com um cliente desconhecido.

Assim uma empresa que utiliza um determinado software e deseje trocá-lo, deverá gastar tempo na busca de um novo fornecedor que possua um programa que supra suas necessidades, realizar testes para garantir que os dados anteriormente armazenados sejam reaproveitados pelo novo software e incorrer na incerteza de lidar com um fornecedor desconhecido que pode não cumprir determinada parte do acordo.

Já o fornecedor que queira procurar um novo cliente, deverá arcar com gastos de propaganda, criar atrativos que estimulem o futuro cliente, saber lidar

com a lealdade dos clientes com seus atuais fornecedores e a insegurança de lidar com um novo cliente que pode, por exemplo, não realizar os pagamentos.

Assim sendo, o aprisionamento será maior quanto mais dispendioso for a busca de novos clientes ou fornecedores.

Programas de lealdade

Shapiro e Varian classificam essa categoria de aprisionamento como um “aprisionamento artificial”, pois este surge da elaboração de uma estratégia empresarial. Tal estratégia envolve táticas que estimulem a fidelidade dos clientes a determinada marca. Exemplo dessas estratégias são os programas de milhagens áreas, cartões de créditos ofertados pelos fabricantes de automóveis a seus clientes e que acumulam bônus que podem ser trocados por descontos na hora da troca do carro, os valores diferenciados pela quantidade comprada de um determinado produto, valor diferenciado pelo número de licença a ser adquirido de um certo software, brindes após a compra ultrapassar valores estabelecidos, oferecer a possibilidade de concorrer a prêmios após gastar mais que um valor específico, etc.

Os custos de troca criados pelos programas de lealdade ocorrem de variadas maneiras. De um lado, pode fazer com que o cliente perca algum atributo estimulante da escolha da marca (brinde, prêmios, descontos, etc.), por exemplo, um cliente pode deixar de concorrer a um carro se gastar R\$ 90,00 ao invés dos R\$ 100,00 necessários para participar. De outra forma, uma empresa pode oferecer benefícios aos seus clientes de acordo com seus respectivos históricos de compra.

Com o avanço da informática e da Internet, tornou-se mais fácil para os fornecedores registrar o padrão de compra dos consumidores, tornando possível criar preços e propagandas personalizados. Hoje é possível através de cookies⁶ e

⁶ São dados que ficam registrados na máquina local do usuário que normalmente são fornecidas pelo próprio usuário. Esse dados podem ser utilizados, por exemplo, para lembrar o nome do

spywares⁷ observar o comportamento do internauta em seu processo de navegação – observar quanto tempo o usuário ficou em determinada página de Internet, registrar qual site foi visitado após o internauta sair da loja on-line, coletar os períodos nos quais os usuários mais navegam, etc. Torna-se fácil criar programas de lealdade de acordo com as características de cada cliente, assim como enviar propagandas personalizadas para potenciais clientes.

Além desses fatores de aprisionamento mencionados por Shapiro e Varian, cabe observar outros dois tipos de aprisionamento: a incompatibilidade de sistemas e o comprometimento da cadeia em favor de determinado padrão.

Incompatibilidade de sistemas

Na área de TI é necessário que sistemas distintos “conversem”, ou seja, que possa haver transferência e leitura de dados de forma simples e eficiente entre ambos. A mudança de um elemento não pode ser realizada sem estudar seu impacto nos demais componentes do sistema. No caso dos softwares, a escolha de determinado ambiente operacional (p. ex. Windows) leva a comprar e desenvolver programas que são desenvolvidos tendo esse ambiente como pressuposto de programação. Mudar o ambiente significa realizar pesados esforços para readequar todos os programas ao novo ambiente.

No caso de uma mudança de sistema operacional é necessário não apenas garantir o funcionamento de todas as aplicações para o novo ambiente, mas também o funcionamento de todos os hardwares utilizados pelo sistema.

usuário quando o mesmo entrar em um site, ou até mesmo permitir que suas preferências sejam recordadas.

⁷ Spyware é um software que é instalado em um computador, que colecta informações sem o conhecimento do usuário, e as envia para anunciantes ou outras partes interessadas. Ele pode colectar e transmitir informações sobre teclas pressionadas, hábitos de navegação na Web, endereços de e-mail etc.

Comprometimento da cadeia com determinado padrão

Alguns produtos precisam de uma cadeia de apoiadores para seu sucesso. Assim, grande parte do sucesso da Matsuchita com o padrão VHS, deve-se a sua estratégia de licenciar sua tecnologia para outros concorrentes, o que convenceu os produtores de filmes, os distribuidores e os lojistas a apostar na expansão do padrão, deslocando o Betamax da Sony.

Na produção de *softwares*, as *soft-houses* tendem a adquirir os programas (meios de produção) que sejam compatíveis com o padrão dominante, ou seja, se esse padrão é o Windows, tenderão a adquirir programas que trabalhem com esse ambiente e produzirão programas (produtos finais) que vão reproduzir tal necessidade de padronização. Os programadores terão habilidade no ambiente Windows. Assim, a oferta de produtos, passa a ser condicionada pelo padrão dominante, qualquer mudança exige alterações tecnológicas tanto no produtor como no cliente.

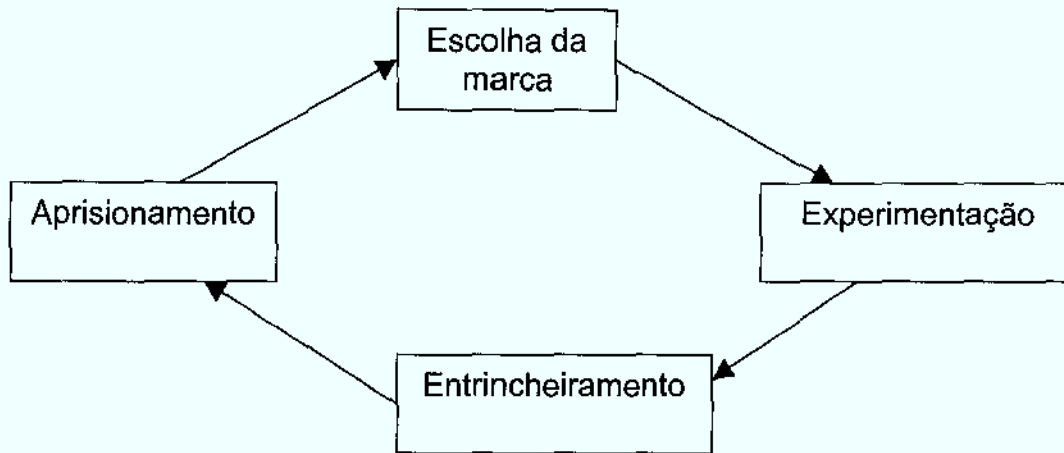
O Ciclo do Aprisionamento

O ciclo do aprisionamento inicia-se no momento da seleção de uma marca. Quando a escolha é a primeira que se realiza, não existe preferência por uma marca, mas após optar pela primeira marca incorre-se no aprisionamento, não ficando tão livre para escolher a próxima marca.

Após a escolha da marca, entra-se na fase de experimentação, na qual o usuário estará testando a marca e usufruindo as vantagens e incentivos que receber da mesma.

Posteriormente entra-se na fase do entrincheiramento, que ocorre quando o usuário acostuma-se a nova marca, dando preferência a ela em relação às outras marcas. Quanto mais tempo durar esta fase, maiores serão os custos de troca, levando a fase do aprisionamento, onde os custos de mudança tornam-se altos demais.

Figura 2.2 – Ciclo do aprisionamento



Fonte: SHAPIRO e VARIAN (1999)

Volta-se então a escolha da marca, que não mais ocorrerá livremente, uma vez que podem existir contratos com os fornecedores, os usuários estão acostumados com a marca utilizada, os arquivos de dados estão em determinado formato, etc. Assim, os custos de troca tornam-se mais altos, dificultando a migração para outras novas tecnologias que possam ter surgido.

Portanto para uma empresa diminuir seu aprisionamento frente a determinado sistema de informação, torna-se necessário que ela consiga antecipar vários ciclos no futuro ao traçar determinada estratégia, procurando ficar o mínimo possível dependente de determinada tecnologia.

Feedback Positivo

Quando um sistema se beneficia do maior número de usuários que o utilizam, fazendo com que os novos usuários também optem por utilizar o mesmo sistema, temos o chamado feedback positivo. Sistemas que apresentam esse padrão alcançam maiores vantagens quanto maior for a base instalada de

usuários de um determinado sistema, pois maior será o número de novos usuários que optaram por utilizar este sistema.

Esse efeito pode ser facilmente visto a partir da observação do uso do Sistema Operacional Microsoft Windows, para o qual, tendo em vista sua enorme base de usuários, acaba se impondo para as pessoas que queiram aprender informática, que procurem um curso que ensine ou ofereça treinamento para o uso deste sistema operacional, assim como, e por decorrência, uma nova empresa tenderá a optar por utilizar este sistema devido a maior facilidade de encontrar pessoas que o saibam operar. Além disso, a popularização do sistema operacional estimulou a produção de um maior número de programas e drivers⁸ para o sistema, fazendo com o Windows ficasse mais atraente para novos clientes.

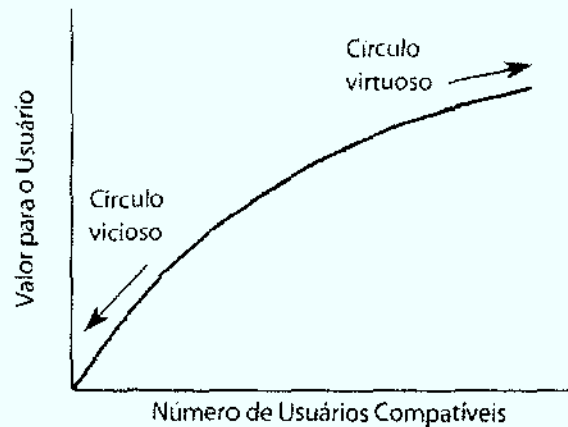
No entanto, o feedback não age somente de forma favorável, podendo acelerar o fracasso de determinado produto, caso o mercado não entenda o produto como promissor. Assim, o produto entra no chamado círculo vicioso do feedback ou feedback negativo.

Como exemplo do círculo vicioso, se pode pegar o caso da Netscape, onde a perda de usuários para o Internet Explorer levou ao fechamento do grupo. Já como exemplo do círculo virtuoso existe o caso dos sistemas operacionais da Microsoft, onde o Windows é valorizado devido ao grande número de usuários que o utilizam, fazendo com que seja o padrão do setor.

Assim o feedback positivo acaba por fortalecer o mais forte e enfraquecer o mais fraco, como mostra a figura 2.4. No entanto, isso não significa que o produto que estiver atrás no mercado irá necessariamente perder, como exemplo disso temos o MS-Word que, apesar de ter permanecido atrás do WordPerfect por um bom tempo, hoje é líder do mercado de editores de textos.

⁸ Pequenos programas que fazem com que o hardware funcione com o Sistema Operacional.

Figura 2.3 – Valor x Popularidade



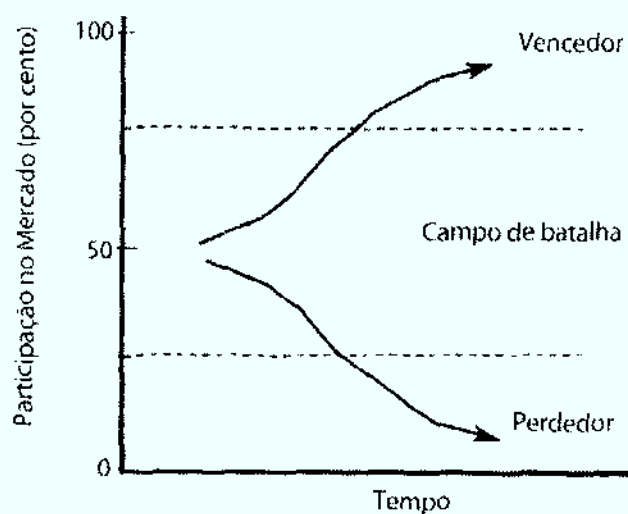
Fonte: SHAPIRO e VARIAN (1999)

Na economia da informação o feedback positivo em seu ciclo virtuoso reforça ainda mais a posição de líder do produto, através das economias de escala do lado da demanda, onde um produto como o MS-Windows torna-se uma razão para ser usado devido a grande quantidade de pessoas que utilizam esse sistema. Isso faz com que qualquer sistema operacional concorrente que surja tenha dificuldade em obter massa crítica para concorrer com o sistema da Microsoft.

As exterioridades de rede positivas, que agem no sentido de conformar situações nas quais um membro de um mercado interfere sobre outros membros sem gerar contrapartidas de receitas ou rendimentos, faz surgir o feedback positivo. Assim, se um usuário começa a utilizar um programa que grava e lê vídeos em um determinado formato, outros usuários ficarão mais propícios a utilizar o mesmo programa para poderem trocar arquivos entre si. Um exemplo bastante recente disso são as redes de troca ponto-a-ponto⁹, tais como o Napster, o Kaaza, e Fastrack onde a rede que obtiver um maior número de usuário tendem a receber mais usuários novos, pois possuem maiores possibilidades para a troca de arquivos entre usuários.

⁹ Redes onde dois usuários podem trocar arquivos diretamente entre si.

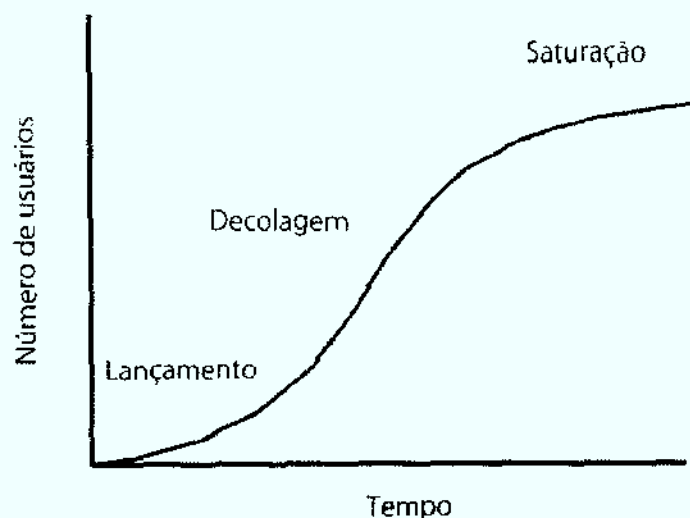
Figura 2.4 – Feedback positivo



Fonte: SHAPIRO e VARIAN (1999)

O feedback positivo segue um padrão de uma curva em S, onde na fase inicial temos o lançamento do produto, a decolagem do produto (acentuada pelo feedback positivo) e por último a saturação do produto. Esse comportamento pode ser visto na figura 2.5:

Figura 2.5 – Curva em S



Fonte: SHAPIRO e VARIAN (1999)

Shapiro e Varian colocam duas abordagens para contornar os problemas inerciais causados pelo feedback positivo e pelo aprisionamento tecnológicos: a estratégia da evolução tecnológica e a estratégia da revolução tecnológica. Estas duas estratégias serão discutidas a seguir.

Estratégia da Evolução

Esta estratégia se baseia em facilitar a migração dos usuários para a nova tecnologia, para isso se devem utilizar táticas que permitam reduzir o aprisionamento tecnológico, diminuindo os custos de troca e proporcionando uma migração sem ruptura.

Esta estratégia foi utilizada diversas vezes pela Microsoft para conseguir que usuários que utilizavam programas de versões antigas ou de concorrente, migrassem para uma nova versão.

No entanto esta estratégia esbarra em obstáculos técnicos e legais. O primeiro refere-se às dificuldades de criar uma nova tecnologia que se mantenha compatível com a tecnologia anterior e ao mesmo tempo possua recursos novos que a tornem atraente para novos consumidores.

Manter a compatibilidade com a tecnologia anterior é essencial para reduzir os custos de trocas para os clientes. No entanto esta estratégia pode ter como principal custo à limitação do desempenho da nova tecnologia, abrindo caminho para que novos produtos revolucionários conquistem o mercado.

O obstáculo legal surge da possibilidade de não conseguir o direito legal de produzir produtos similares aos já instalados. Essa barreira normalmente aparece nas formas das patentes que mantêm o direito a propriedade intelectual de determinado fabricante, podendo este cobrar pelo licenciamento do produto ou até mesmo impossibilitar que o produto seja fabricado por um novo concorrente. Um exemplo disso foi a estratégia da Hewlett-Packard quando patenteou um número muito grande das peças de suas impressoras a jato de tinta, causando sérios transtorno para que a Epson conseguisse lançar estes produtos.

Estratégia da Revolução

Esta estratégia parte do princípio da ruptura, oferecendo uma tecnologia que traga uma grande quantidade de melhorias, tornando-se irresistível para os usuários.

Segundo Shapiro e Varian, esta estratégia primeiramente atrai os clientes mais preocupados com o desempenho, que são os usuários pioneiros e normalmente são especialistas da área e possuem a capacidade de influenciar os clientes de massa. Ao conquistar esses usuários pioneiros, os fabricantes buscam utilizar a influência desses clientes para começar a impulsionar o feedback positivo, fazendo com que os consumidores acreditem que o sucesso do produto é inevitável.

Esta estratégia tende a proporcionar melhores resultados em um mercado em rápido crescimento e com baixo aprisionamento. No entanto, a estratégia da revolução é arriscada, uma vez que é quase impossível saber se a sua tecnologia obterá sucesso ou não.

Consolidação de padrão: o projeto dominante

Utterback (1994) desenvolve um modelo explicativo da inovação industrial após o estudo de diversos casos. O modelo, segundo o autor, é válido para explicar as inovações introduzidas dentro do contexto industrial dos Estados Unidos nos últimos 150 anos, em produtos onde custo e desempenho sejam fatores importantes. Nesse modelo, o autor explica como, a partir de projetos inovadores numa indústria, se consolida um determinado padrão que atende as necessidades dos clientes e como esse padrão torna-se dominante. Segundo Utterback o projeto inovador vai se cristalizando ao longo do tempo, até atingir a forma de um projeto dominante.

Utterback distingue três fases na inovação da indústria que vão ocorrendo ao longo do tempo: fluída, transitória e específica.

Na fase fluida de um setor, o novo conceito é lançado por uma ou mais empresas. Os vários concorrentes operam com projetos diferentes para atender a mesma necessidade dos clientes. Os concorrentes tentam descobrir quais as características mais adequadas do produto do ponto de vista dos clientes. Como um produto é um *"pacote de valor ou feixe de benefícios"* (Kotler, Fox, 1994:257) não é evidente para os produtores, qual é a melhor solução do ponto de vista dos clientes.

Num certo momento há uma escolha por parte dos clientes, os que manifestam preferir uma solução em lugar de outra. O projeto preferido conquista a fidelidade do mercado. Esse projeto é aquele *"que satisfaz o usuário em termos de interação das possibilidades técnicas e opções de mercado"* (Utterback, 1994:27). Além dos aspectos técnicos, vários fatores adicionais contribuem para que um projeto seja dominante: uma forte imagem corporativa, as vantagens que uma empresa estabelecida já possui e que pode alavancar em seu favor, os canais de distribuição, a imagem, os regulamentos setoriais, as manobras estratégicas e a interação com os clientes, forçando uma aprendizagem de mercado que seja favorável.

O projeto dominante reduz a liberdade inicial de requisitos de desempenho a serem atendidos por um produto, e os transforma em implícitos ao próprio projeto. Isto ocorre, pois os clientes "votaram" em favor de uma solução, o que diminui as possibilidades de sobrevivência de outras soluções, salvo em nichos específicos. Assim, quando a IBM lançou o PC, esta solução tornou-se o projeto dominante, no caso de computadores pessoais, levando a saída de mercado a empresas que já operavam (Sinclair, Amiga, etc.) e deslocando a Macintosh para um nicho de mercado.

Tendo ocorrido a escolha do projeto dominante, as outras empresas serão obrigadas a atingir a lógica do projeto dominante para poder continuar concorrendo. O projeto dominante consolida a definição de padrões técnicos e a padronização de partes. A concorrência desloca-se das características inovadoras para dimensões de custo, escala e desempenho (Utterback, 1994:34). A taxa de

inovação se reduz, passando a ser mais de natureza incremental para apoiar melhorias de eficiência e desempenho.

Como que a adequação do próprio produto ao projeto dominante leva tempo, muitas empresas podem não consegui-lo ou podem ficar em desvantagem, frente ao rival mais rápido, que já se apropriou de vantagens de custo e de escala. Isso pode significar sua saída da indústria. O número de empresas concorrentes diminui sensivelmente, uma vez estabelecido o projeto dominante. Se uma empresa agrega a ocupação de mercados, a obtenção de vantagens de custo e de escala à vantagem da detenção do projeto dominante e, ademais, aposta em contínuas inovações incrementais terá, possivelmente assegurada, uma confortável liderança em seu setor, até o aparecimento de alguma inovação radical que ameace sua posição.

Quando uma empresa que obtenho um projeto dominante usufruirá vantagens competitivas alavancadas pelo feedback positivo, conseguindo um número cada vez maior de usuários que irão reforçar sua posição como projeto dominante.

Por tanto, uma empresa na área de software, que se aproprie dos benefícios de um projeto dominante, será alavancada pelos efeitos positivos do *feed-back* e os concorrentes terão sérias dificuldades para desalojá-la. Pelo contrário, essa empresa terá condições de expulsar a maioria (ou totalidade dos concorrentes) tendendo a impor uma situação de monopólio. Este é o caso da Microsoft, que impôs um monopólio no referente aos sistemas operacionais.

Para desalojar essa empresa, serão necessários esforços coordenados de diversos agentes, sejam empresas concorrentes, sejam usuários além do papel do setor público apoiando esses esforços. Esses agentes, além de gerar uma alternativa ao projeto dominante, deverão obter os “votos” do mercado em favor de sua solução.

Aprisionamento voltado ao software

As etapas do processo decisório para compra de um software são:

1. Definição dos requisitos.
2. Escolha do software
3. Aquisição
4. Implantação e integração com outros sistemas
5. Uso inicial
6. Substituição (em função da obsolescência do software)

O custo de aprisionamento cresce a partir da etapa de aquisição e torna-se extremamente elevado com o uso inicial, atingindo valores máximos com a integração que vai naturalmente ocorrendo com outros softwares. A obsolescência do software leva a estudos de substituição, nesse ponto o aprisionamento torna-se mínimo (porém não nulo). Nessa etapa o usuário pode decidir manter o padrão anterior, reforçando novamente o aprisionamento, ou escolher outro padrão, entrando em um novo padrão de aprisionamento, que pode ser mais forte ou mais fraco e com custos maiores ou menores. A figura 2.6 mostra a variação do grau de aprisionamento ao longo do tempo.

Os principais tipos de aprisionamento que afetam mais fortemente o software são:

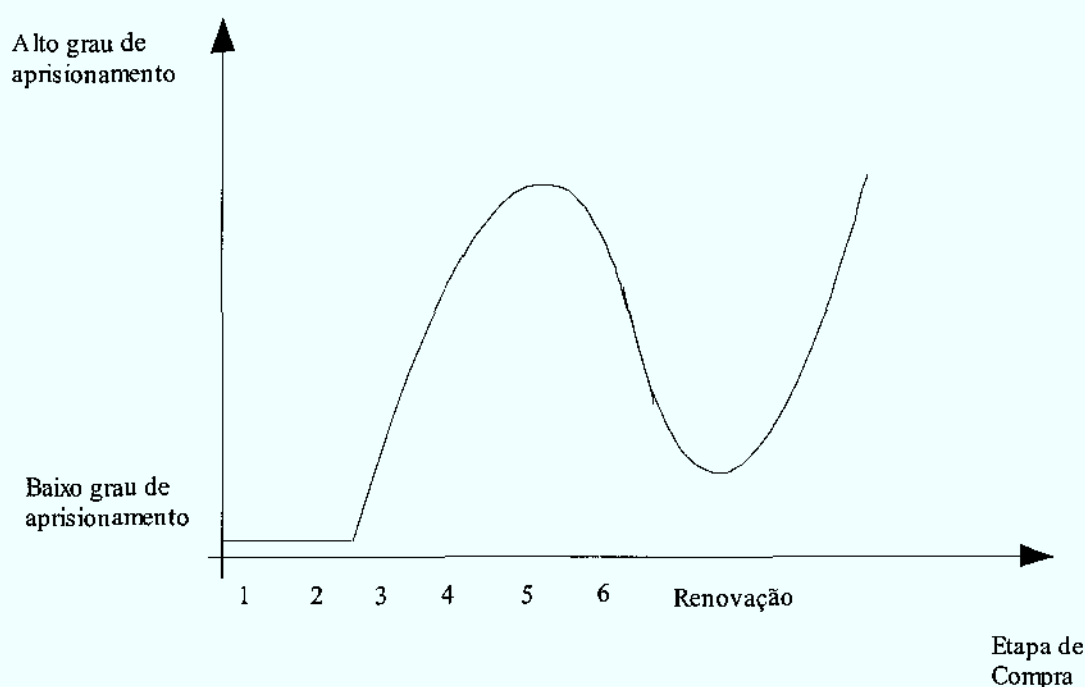
- Informações e Banco de Dados.
- Custos de busca.
- Treinamento em marca específica.
- Incompatibilidade de sistemas.
- Comprometimento da cadeia com determinado padrão.

O mais sério tipo de aprisionamento que dificulta a migração entre softwares está relacionado à preservação e conservação dos dados já existentes. Torna-se praticamente impossível para uma empresa adotar um novo software

caso seus dados não possam ser transferidos, se o processo de transferência é caro ou arriscado demais.

Portanto, é imprescindível que um novo software seja capaz de ler e gravar dados das versões líderes de mercado, diminuindo o aprisionamento e tornando-se uma opção para os usuários.

Figura 2.6 – Grau de aprisionamento nas distintas etapas de compra



Os custos de busca surgem na hora em que a organização (ou o usuário) optar pela migração de software, pois é necessário procurar programas que atendam as necessidades do usuário e realizar diversos testes para garantir a qualidade do software e sua compatibilidade com o resto do sistema do usuário.

Outra forte barreira à migração do software é o treinamento em um software específico. Os usuários que estiverem habituados a utilizar um software poderão, no início, não apresentar o mesmo desempenho com o novo programa. Além disso, existe o fator psicológico para os usuários de terem que aprender a lidar com uma nova ferramenta, fazendo com que muitos deles imponham resistência à mudança do software.

A incompatibilidade de sistema constitui de mais um problema para do software, pois o usuário precisa garantir que os diversos sistemas que possui possam trocar dados. Por exemplo, quando se deseja trocar de banco de dados, é necessário garantir que os programas que acessavam os dados do banco de dados antigo consigam acessar a nova base de dados. Outro exemplo se refere à mudança de sistema operacional, onde programas feitos para o sistema vigente podem não rodar no novo sistema operacional. Além disso, é necessário garantir que existam drivers para todos os hardwares utilizados pelo sistema.

O comprometimento da cadeia com determinado padrão poderá agravar o problema da incompatibilidade de sistemas, pois um sistema operacional que seja líder absoluto do mercado, tenderá a possuir um número maior de programas e drivers sendo feito para o ambiente dominante, limitando assim, as opções de sistemas operacionais concorrentes.

Portanto, o usuário ao estudar a possibilidade de migração deve considerar todos os aprisionamentos com os quais está envolvido. Além disso, para garantir uma melhor chance de sucesso, o usuário deve realizar a migração em um momento no qual o aprisionamento esteja reduzido.

Capítulo 3 – Aprisionamento x Software Livre no estudo da migração

Os programadores de software livre, conscientemente ou não, têm adotado estratégias que visam diminuir o aprisionamento tecnológico e favorecer a adoção desse tipo de software pelos usuários.

As principais estratégias dos programas de software livre que diminuem o aprisionamento tecnológico são:

- Tornar o programa capaz de ler e gravar dados no formato dos principais softwares.
- Criar interfaces gráficas similares aos dos programas líderes, facilitando o aprendizado para um novo usuário.
- Permitir que o programa livre seja executado em diversos sistemas operacionais, inclusive no Microsoft Windows.
- Criar documentação que facilite o aprendizado do usuário em diversos idiomas.
- Criar fóruns de discussões on-line onde se podem resolver problemas, expor experiências vividas e realizar sugestões para melhoras dos softwares.
- Adicionar recursos que os programas proprietários não possuem e que os usuários possam valorizar.

A seguir serão estudados alguns dos principais segmentos de softwares, visando analisar os principais pontos de aprisionamento que dificultam a migração do software proprietário para o software livre.

Pacotes de escritório

O pacote de aplicativos para escritório possui os programas mais utilizados pelos usuários. O pacote é constituído de um programa para edição de texto, uma planilha eletrônica, um programa para apresentação e é desejável que possua um

banco de dados. O mais conhecido e utilizado pacote de escritório é o Microsoft Office, formado pelo editor de texto Word, a planilha Excel, o programa para apresentação PowerPoint, o banco de dados Access e o Outlook.

Assim, é imprescindível para a migração do software proprietário que existam programas compatíveis como o Microsoft Office. O programa aberto que apresenta maior compatibilidade com o MS-Office é o OpenOffice, apesar de existirem diversos aplicativos abertos para escritórios, tais como o KOffice, o AbiWord, GNumeric, etc.

O OpenOffice além do editor de textos (SWriter), a planilha eletrônica (SCalc), o programa para apresentação (SImpres) e o banco de dados, ainda é composto por um programa para desenho vetoriais (SDraw). Outro diferencial desse programa é sua facilidade em exportar os arquivos para o formato pdf e o fato de possuir um formato de gravação que ocupa cerca de um quarto de espaço do formato da Microsoft.

Além disso, o OpenOffice oferece suporte para diversas línguas, o que facilita a adoção do programa em diversos países, e pode ser executado em diversos sistemas operacionais, dando maior flexibilidade aos seus usuários.

Editor de Texto

O Editor de texto do OpenOffice é capaz de abrir documentos do Word com bastante precisão, praticamente sem que se perda qualquer formatação. Além disso, o SWriter consegue gravar os documentos no formato doc, fazendo com que o aprisionamento pelo dados seja praticamente nulo.

A interface do SWriter busca ser parecida com a do Word, tendo ícones semelhantes e uma disposição dos menus similar, fazendo com que o aprendizado do programa se torne mais fácil.

A figura 3.1 mostra um documento aberto no Word e a 3.2 mostra o mesmo documento aberto no SWriter. Pode-se notar que o documento foi aberto corretamente pelo Openoffice e a semelhança dos principais ícones.

Figura 3.1 – Documento dolarização.doc aberto Word

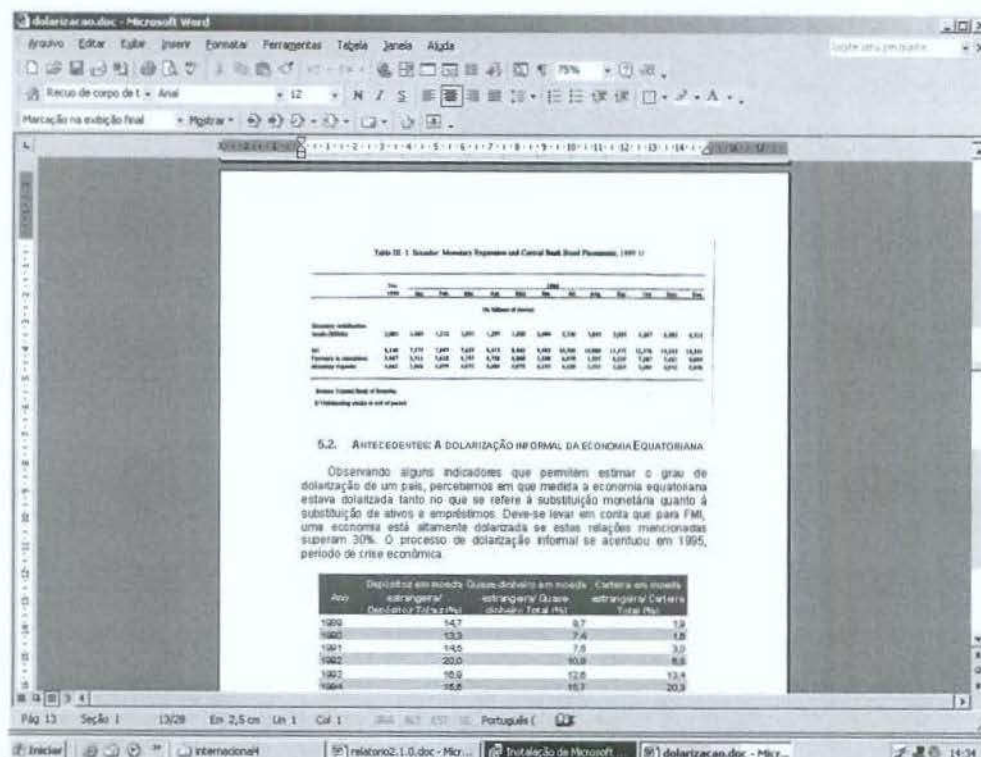
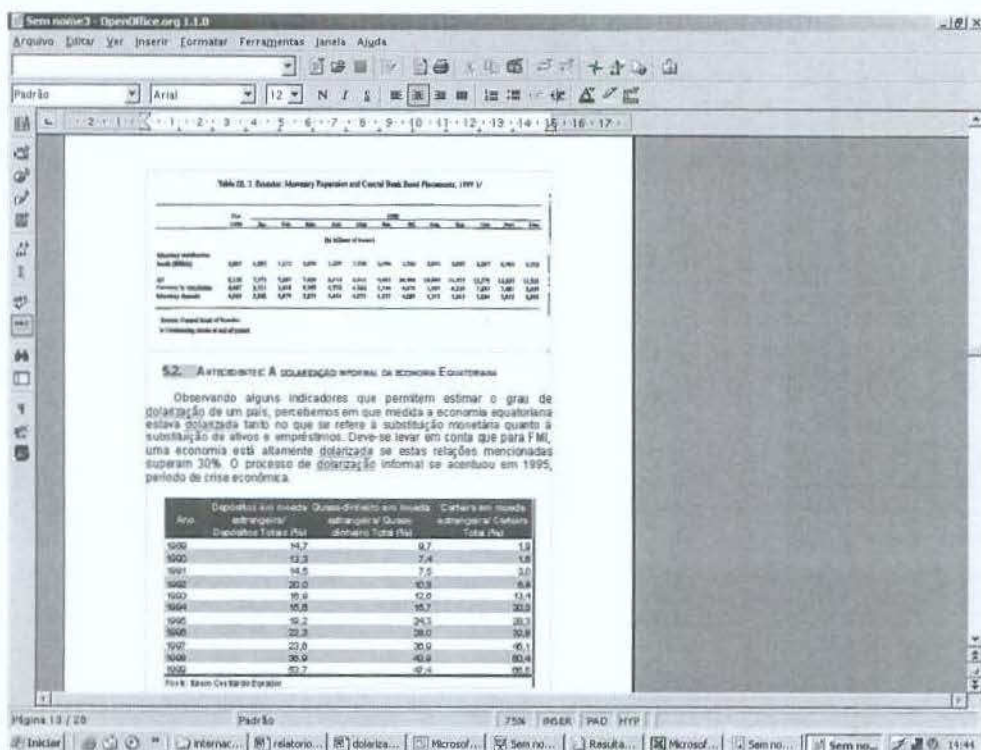


Figura 3.2 – Documento dolarização.doc aberto SWriter



Programa para Apresentação

O programa de apresentação do OpenOffice é capaz de abrir e gravar documentos do PowerPoint de forma bastante precisa.

A disposição dos menus do SImpress é relativamente parecida com a do PowerPoint. Como pode ser comprovado através das figuras 3.3 (mostrando um documento aberto no PowerPoint) e a 3.4 (mostrando o mesmo documento aberto no SImpress).

Figura 3.3 – Documento dolarização.ppt aberto no PowerPoint

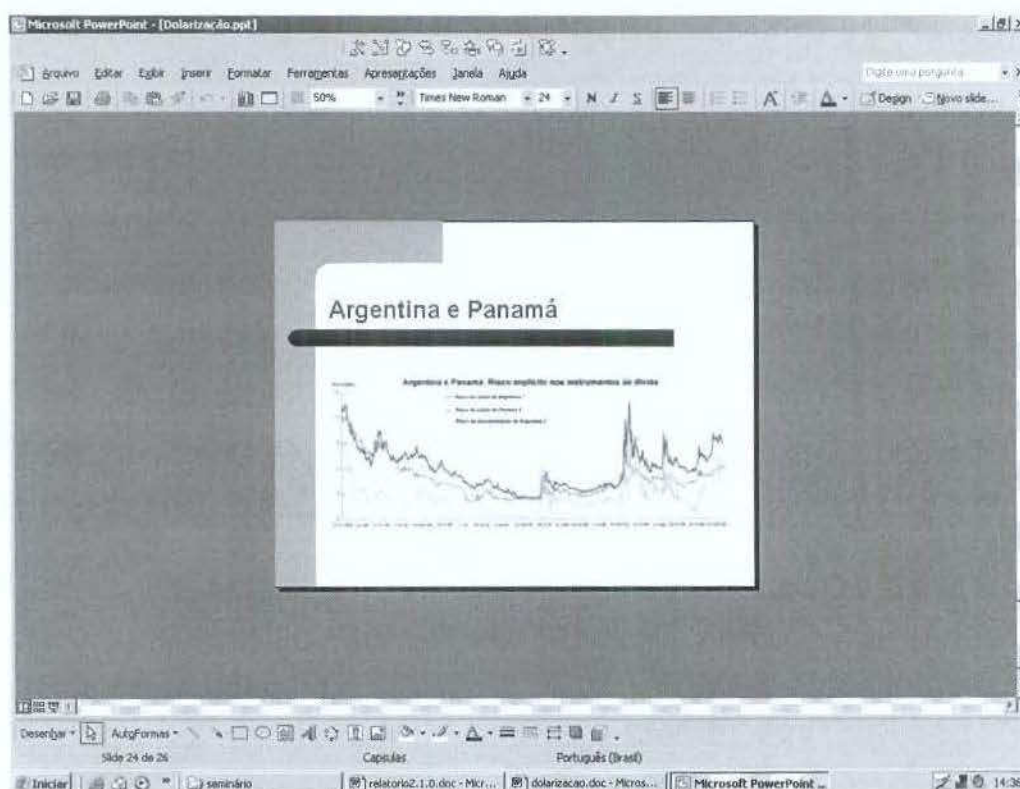
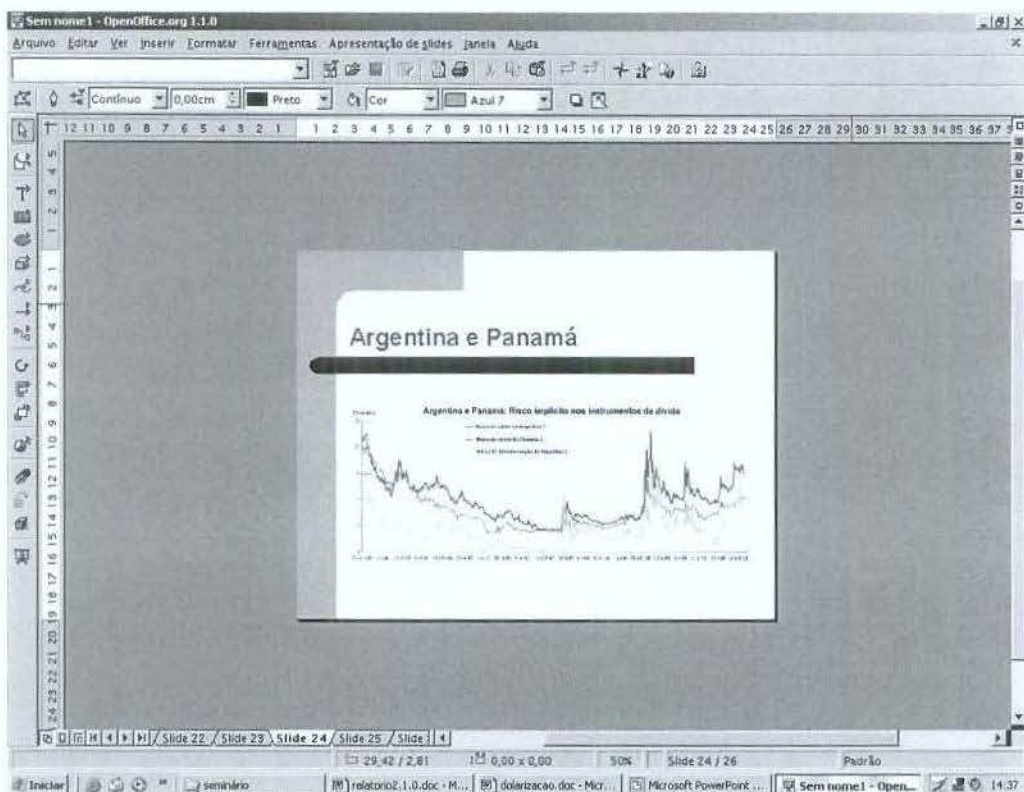


Figura 3.4 – Documento dolorização.ppt aberto no SImpress



Planilha Eletrônica

O OpenOffice possui o programa de planilha eletrônica SCalc capaz de ler e gravar arquivos do formato Excel de maneira bem precisa. Além disso, a planilha livre possui, assim como programa da Microsoft, um grande número de funções que podem ser inseridas pelo usuário. Os desenvolvedores do OpenOffice criaram estas funções de forma tal que os seus parâmetros sejam idênticos aos parâmetros das funções do Excel.

A figura 3.5 mostra um documento aberto no Excel e a 3.6 mostra o mesmo documento aberto no SCalc. Pode-se notar que o documento foi aberto corretamente pelo Openoffice e a semelhança da interface gráfica.

Figura 3.5 – Documento brandao(1).xls aberto no Excel

setor	empresa	liderança de mercado (%)	posição	mercado nacional ¹	mercado total ¹⁺²
alimentos	SADIA	7,4	4	14,9	57,3
	PERDIOÃO AGROINDUSTRIAL	5,8	5		
	ITAMBÉ	1,7	10		
automotivo	EMBRAER	9,5	2	11,3	30,6
	MARCOPOLO	1,1	5		
	RANDON	0,7	10		
bebidas	CBBIAMBEV	49,6	1	62,5	74,2
	COCA-COLA VONPAR	3,2	4		
	REFRIGERANTES MÔ	2,8	5		
	CERVEJARIA ASTRA	2,0	6		
	ANTARCTICA DO SUDESTE	1,6	8		
	AROSUCO	1,7	9		
confecções e têxtil	SCHINCAROL NE	1,6	10	46,2	54,9
	VICUNHA TÊXTIL	11,2	1		
	COTEMINAS	6,7	2		
	SÃO PAULO ALPARGATAS	6,6	3		
	ORENENE SOBRAL	5,8	5		
	AZALEIA	5,4	6		
	TEKA	4,2	7		
setor	HERING	3,9	8		
	DE MILLUS	2,4	10		
setor	NORBERTO ODEBRECHT	14,1	1		
	CAMARGO CORRÊA	9,6	2		

Figura 3.6 – Documento brandao(1).xls aberto no SCalc

setor	empresa	liderança de mercado (%)	posição	mercado nacional ¹	mercado total ¹⁺²
alimentos	SADIA	7,4	4	14,9	57,3
	PERDIOÃO AGROINDUSTRIAL	5,8	5		
	ITAMBÉ	1,7	10		
automotivo	EMBRAER	9,5	2	11,3	30,6
	MARCOPOLO	1,1	5		
	RANDON	0,7	10		
bebidas	CBBIAMBEV	49,6	1	62,5	74,2
	COCA-COLA VONPAR	3,2	4		
	REFRIGERANTES MÔ	2,8	5		
	CERVEJARIA ASTRA	2,0	6		
	ANTARCTICA DO SUDESTE	1,6	8		
	AROSUCO	1,7	9		
confecções e têxtil	SCHINCAROL NE	1,6	10	46,2	54,9
	VICUNHA TÊXTIL	11,2	1		
	COTEMINAS	6,7	2		
	SÃO PAULO ALPARGATAS	6,6	3		
	ORENENE SOBRAL	5,8	5		
	AZALEIA	5,4	6		
	TEKA	4,2	7		
setor	HERING	3,9	8		
	DE MILLUS	2,4	10		
setor	NORBERTO ODEBRECHT	14,1	1		
	CAMARGO CORRÊA	9,6	2		

Considerações finais sobre os pacotes de escritórios

O OpenOffice é capaz de ler e gravar, de maneira bastante precisa, no formato dos documentos do Microsoft Office, reduzindo muito o aprisionamento tecnológico pelo lado de Informações e banco de dados. Além disso, por possuir uma interface gráfica semelhante ao MS-Office, o OpenOffice reduz tempo de aprendizagem para o novo usuário, facilitando a sua adoção.

No entanto, o OpenOffice ainda não é capaz de ler e executar macros no formato da Microsoft, sendo necessário recriar estas macros para o formato utilizado por este software livre. Apesar da linguagem ser razoavelmente semelhante, isso pode ocasionar dificuldades na hora da migração por usuários mais avançados que utilizam este recurso.

Visando facilitar a adoção do OpenOffice foram criados diversos documentos e fóruns de discussões onde se pode aprender e tirar dúvidas sobre este programa de forma rápida e gratuita. Exemplos destas iniciativas no Brasil são o Rau-tu da Unicamp (www.rau-tu.unicamp.br/openoffice) e o OpenOffice.org.br (www.openoffice.org.br).

Apesar do OpenOffice não possuir nenhum programa similar ao Outlook da Microsoft é possível utilizar programas livres similares como o Ximian-Evolution. A similaridade entre estes programas podem ser vistas através das figuras 3.7 mostrando o Outlook e 3.8 mostrando o Evolution.

Figura 3.7 – Outlook

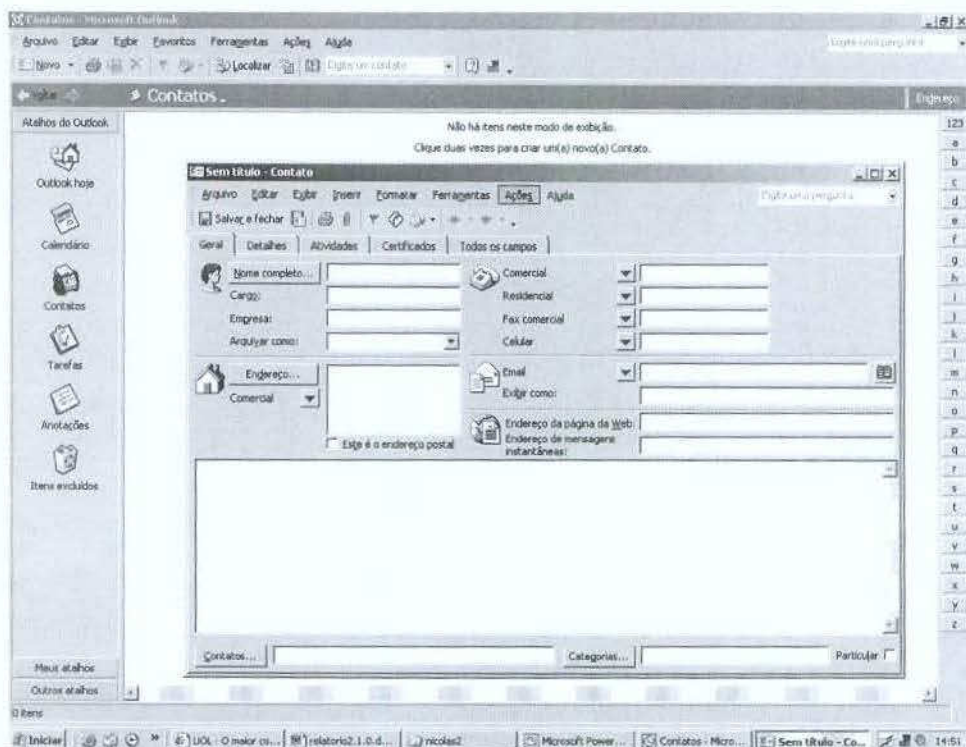
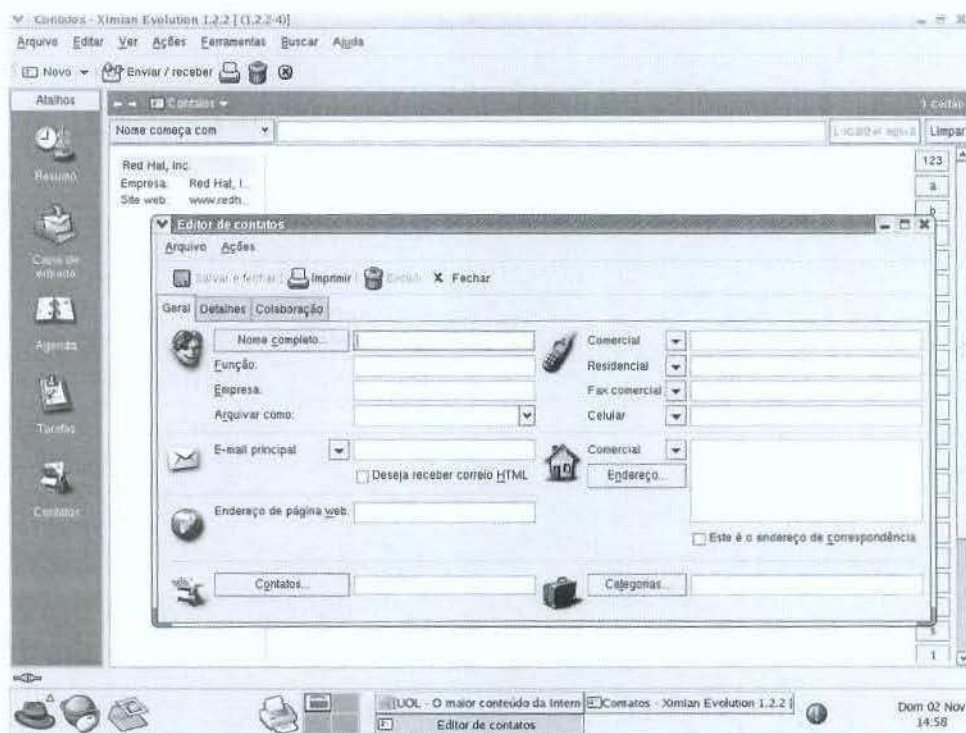


Figura 3.8 – Evolution



Banco de Dados

Os banco de dados são um aspecto crítico ao se estudar a migração de um software para outro, pois normalmente estes guardam uma série de dados críticos para o bom funcionamento das empresas.

Alguns dos principais bancos de dados proprietários são: Oracle, Access, Microsof SQL Server, Informix, etc. A conversão de dados destes para banco de dados livres, tais como o Mysql e Postgres, pode ser feita de forma relativamente simples, sem incorrer em grandes riscos. No entanto, o principal obstáculo da migração dos bancos de dados são as triggers ¹⁰, que devem ser refeitas de acordo com as peculiaridades da linguagem de cada banco de dados. Isso pode tornar extremamente custosa a adoção de um novo banco de dados.

O Mysql tem como principal vantagem ser um banco de dados simples e bastante rápido. No entanto, este banco de dados ainda não suporta triggers e não é recomendado para grandes volumes de dados.

O Postgres é um banco de dados bastante completo e estável, suportando triggers e um grande volume de dados. Conforme Dosso (2003) (em entrevista realizada para o autor), muitas empresas que utilizam o Oracle (um banco de dados muito caro que tem como principal vantagem suportar uma grande quantidade de dados), sub-utilizam os seus recursos e poderiam estar usando o Postgres como banco de dados, sem comprometer as atividades da empresa.

O principal fator de aprisionamento deste segmento é a incompatibilidade entre as linguagens utilizadas pelos diversos bancos de dados, fazendo com que empresas que utilizem muitas triggers somente retomem seus investimentos na migração no longo prazo.

¹⁰ Conjunto de comandos que são criados dentro do próprio banco de dados, visando simplificar os programas que utilizam o banco de dados. Pode-se definir que os comandos serão executados quando o banco de dados realizar alguma operação, por exemplo, inserção de dados, alteração de dados, exclusão de dados.

Sistema Operacional

O GNU/Linux é o mais conhecido sistema operacional livre, já possuindo um grande apoio na comunidade de software livre e entre os fabricantes de hardware. Isso faz com que o Linux usufrua algumas exterioridades de redes positivas, conseguindo suporte para que um grande número de hardware funcione com este sistema operacional.

Outros fatores que reduzem o aprisionamento que resulta da incompatibilidade de sistemas estão relacionados com o fato do Linux ser capaz de ler e gravar em sistemas de arquivos do padrão da Microsoft. Além disso, existe um programa chamado SAMBA que permite que sistemas Linux troquem dados com sistemas rodando o Microsoft Windows, o que torna viável uma rede mista.

As interfaces gráficas de maior sucesso para Linux têm adotado um visual parecido com o MS-Windows, o que facilita o aprendizado de um usuário que esteja habituado com o sistema da Microsoft. A figura 3.7 mostra a interface gráfica do MS-Windows e a figura 3.8 mostra a interface GNOME do Linux.

Outra incitativa para facilitar a adoção do GNU/Linux é o programa WINE (*Wine Is Not a Emulator* – Wine não é um emulador) que permite que alguns softwares para Windows sejam executados no Linux. Com isso é possível suprir a falta de alguns programas para Linux.

Figura 3.9 – Windows XP com o navegador Internet Explorer ao fundo

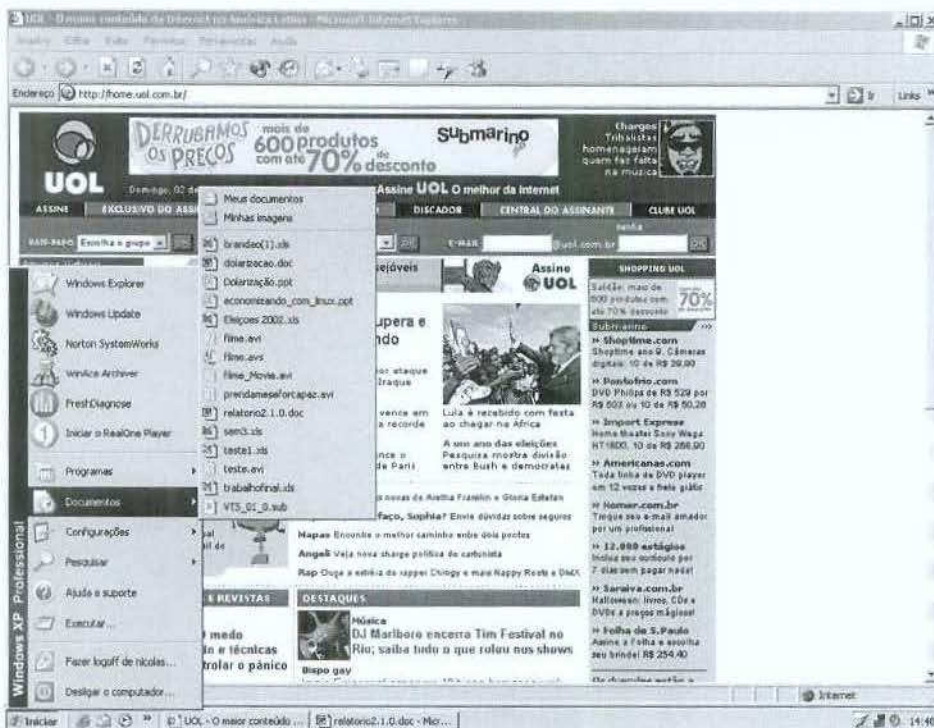
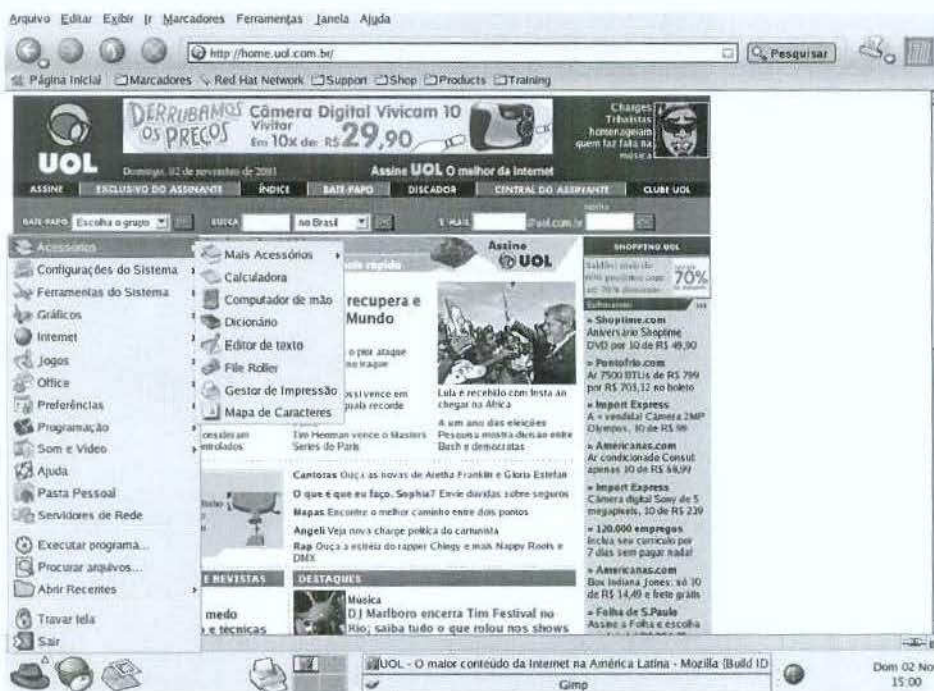


Figura 3.10 – Linux rodando interface gráfica GNOME com o navegador Mozilla ao fundo



Servidor de WEB

O servidor livre para web dominante é o Apache. Este tem sido amplamente adotado devido a sua robustez e por poder ser executado em diversos sistemas operacionais. Além disso, é possível instalar módulos que fazem com que seja possível utilizar diversas linguagens de programação conjuntamente com o apache. Através desses módulos o Apache suporta trabalhar com PHP, Perl e até mesmo o ASP da Microsoft.

Custos do Aprisionamento

O custo do uso do software (C) envolve além dos custos de compra das licenças (l), o valor do treinamento para o uso (t) e o valor atual da renovação periódica das licenças (r) (dentro de um determinado horizonte de planejamento). Também devem ser considerados os custos de adequação do software novo ao ambiente computacional existente (a) (adequação às necessidades dos usuários, compatibilidade de programas, demanda de hardware).

Ou seja, $C = l + t + r + a$ (1)

Uma empresa nova, que necessite de 16 estações de trabalho, possuindo cada uma um sistema operacional e um aplicativo de escritório, instalados, além de servidor de e-mail, um servidor de web, um servidor de banco de dados e duas estações para desenvolvimento contendo cada uma, um sistema operacional, um aplicativo Office, um programa para editoração de imagens e aplicativo para desenvolvimento de software, gastará cerca de R\$73.915,00 caso opte por uma solução baseada em softwares proprietários, como mostra a tabela 3.1.

Tabela 3.1 – Gastos para aquisição de softwares proprietários

Software	Preço Unitário	Quantidade	Total
Microsoft Windows XP professional Edition	R\$ 989,00	18	R\$ 17.802,00
Microsoft Office XP full professional	R\$ 1.850,00	18	R\$ 33.300,00
Antivirus	R\$ 60,00	20	R\$ 1.200,00
Microsoft SQL Server 2000	R\$ 5.013,00	1	R\$ 5.013,00
Microsoft Windows 2000 Server	R\$ 2.800,00	2	R\$ 5.600,00
Adobe PhotoShop	R\$ 2.000,00	2	R\$ 4.000,00
Microsoft VisualStudio (programa para desenvolvimento de software)	R\$ 3.500,00	2	R\$ 7.000,00
Total			R\$ 73.915,00

Valores retirados de www.buscape.com.br em 01/11/2003

Além disso, a empresa terá que voltar a gastar com software a aproximadamente a cada três anos, caso deseje manter seus computadores com softwares atualizados.

Já um sistema baseado em softwares livres com a mesma funcionalidade terá custo zero, como pode ser visto na tabela 3.2. Isso é possível, pois os programas podem ser baixados de forma gratuita na Internet. Além disso, a empresa não incorrerá em gastos caso deseje manter seu parque de computadores com softwares atualizados.

Tabela 3.2 – Gastos para aquisição de softwares livres

Software	Preço Unitário	Quantidade	Total
Linux RedHat 9	R\$ 0,00	20	R\$ 0,00
OpenOffice	R\$ 0,00	18	R\$ 0,00
Sendmail (servidor de e-mail)	R\$ 0,00	1	R\$ 0,00
PostgreSQL	R\$ 0,00	1	R\$ 0,00
Apache (servidor de web)	R\$ 0,00	1	R\$ 0,00
Eclipse (programa para desenvolvimento de software)	R\$ 0,00	0	R\$ 0,00
GIMP (programa equivalente ao PhotoShop)	R\$ 0,00	2	R\$ 0,00
Total			R\$ 0,00

Portanto, no caso de uma empresa nova, os valores de l e r serão nulos, caso opte pelo software livre. Provavelmente terá custos semelhantes de adequação (a) entre ambas soluções. Poderá ter algum custo adicional, que deve ser entendido como marginal em treinamento, t , dado que a maioria das pessoas, conhece Windows, o que é uma externalidade que favorece os softwares proprietários. Mais os valores mostrados, são bem claros quanto à vantagem da adoção dos softwares livres.

Uma empresa que já tem instalado um parque computacional com softwares proprietários deverá comparar, para um determinado período de planejamento, o valor atual do custo esperado de ficar pagando a renovação das licenças para os softwares atuais (r) e de adquirir as licenças dos novos softwares (l) e o custo do treinamento para usar os novos softwares proprietários (tp) com o valor atual do custo dos treinamentos para uso do software livre (tl) e adequações necessárias (alterar programas incompatíveis com o sistema livre e gasto para conversão de dados).

Sendo VAp , o valor atual do custo de posse do software proprietário e VAI o valor atual do custo da migração e posse do software livre, pode-se definir:

$$VAp = l+r+tp \quad (2)$$

$$VAI = tl + a \quad (3)$$

Assim, uma empresa que considere um determinado horizonte de planejamento (por exemplo 5 anos), deverá considerar a migração uma opção válida caso o custo de se fazer um novo treinamento¹¹, de alterar programas incompatíveis com o sistema livre e gasto para conversão de dados for menor que o custo atual de treinamento da empresa com os softwares proprietários, aquisições de novos softwares e atualizações dos programas existente no período considerado no período de planejamento (ou seja deverá migrar quando $VAp > VAI$).

A migração obterá vantagens quando os softwares feitos sob medida para a empresa e que só rodem em uma plataforma proprietária estiverem obsoletos (ponto mínimo de aprisionamento), sendo necessária a confecção de novos programas.

Mesmo que a empresa não se encontre em um momento de aprisionamento mínimo, esta pode considerar a migração de forma gradual de seus sistemas, uma vez que é possível manter a troca de dados os programas livres e proprietário. Isto diminuiria o grau de aprisionamento da empresa com os softwares proprietários.

¹¹ Segundo Terra (2003), os organizadores do IT Conference realizada em Salvador (BA), o custo de treinamento para capacitação na tecnologia do software livre é equivalente a 30% do gasto de um ano com licença de software proprietário.

Considerações finais

Os desenvolvedores de software livre têm reduzido o aprisionamento tecnológico tornando os programas criados capazes de importar e exportar dados com os programas líderes. Além disso, estes programadores têm buscado criar interfaces gráficas similares com os programas mais usados. Isso reduz o aprisionamento uma vez que torna mais simples o aprendizado do novo software.

Outro fator que reduz o aprisionamento é que hoje existem diversos softwares similares aos principais programas proprietário. A tabela a seguir mostra alguns programas similares entre o Windows e o Linux¹².

Tabela 3.3 – Programas similares entre Windows e Linux

Tipo	Windows	Linux
Navegadores de Internet	Internet Explorer, Netscape / Mozilla, Opera [Prop], Firebird,	1) Netscape / Mozilla. 2) Galeon. 3) Konqueror. 4) Opera [Prop] 5) Firebird. 6) Nautilus. 7) Epiphany. 8) Links (with "-g" key). 9) Dillo (Russian language patches - here). 10) Encompass.
Clientes de e-mail	Outlook Express, Netscape / Mozilla, The Bat, Eudora, Becky, Datula, Sylpheed / Sylpheed-claws, Opera	1) Evolution. 2) Netscape / Mozilla messenger. 3) Sylpheed / Sylpheed-claws. 4) Kmail. 5) Gnu. 6) Balsa. 7) Bynari Insight GroupWare Suite [Prop] 8) Arrow. 9) Gnumail. 10) Althea. 11) Lianail. 12) Aethera. 13) MailWarrior. 14) Opera
Mensagens Instantaneas	ICQ Lite, ICQ Corp, MSN, AIM, Yahoo, ...	1) Licq (ICQ). 2) Centericq (Nearly all IM protocols, console). 3) Alicq (ICQ). 4) Micq (ICQ). 5) GnomelCU (ICQ). 6) Gaim. (Nearly all IM protocols). 7) Aytim. (Nearly all IM protocols). 8) Kopete. 9) Everybuddy. 10) Simple Instant Messenger. 11) Imici Messenger. 12) Ickle (ICQ). 13) aMSN (MSN). 14) Kmerlin (MSN). 15) Kicq (ICQ). 16) YSM. (ICQ, console). 17) kxicq. 18) Yahoo Messenger for Unix. 19) Kmess (MSN). 20) AIM. 21) MSNre. (console).

Fonte: LinuxShop (2003)

¹² A tabela completa pode ser vista no Anexo 2 – Softwares similares

Apesar de tudo, o aprisionamento existente através dos custos de busca e treinamento em marcas específicas, conversões de dados e incompatibilidade de sistemas ainda são relevantes.

No entanto, uma empresa que busque encarar esses custos de mudança pode obter ganhos no longo e curto prazo, como será mostrado no próximo capítulo.

Capítulo 4 – Análise de casos de adoção e migração do software livre

Segundo Dosso (2003), em uma fase inicial as empresas procuravam o software livre como uma opção devido ao seu baixo custo. No entanto, com a popularização e aumento do conhecimento deste tipo de software, muitas empresas passaram a considerar este uma opção não somente pelo seu baixo custo, mas também devido a sua qualidade, segurança e robustez.

O Linux, devido a sua grande flexibilidade, possibilita que seus usuários revitalizem máquinas antigas. Com isso é possível rodar sistemas de forma satisfatória em máquinas que o usuário já possua, sem a necessidade de troca de hardware como a qual seria necessária para rodar um sistema de mesmo desempenho com softwares proprietários. Isto significa uma economia em hardware, o que reduz os custos de mudança e incentiva a adoção do sistema livre.

A seguir serão mostrados alguns casos extraídos de revistas especializadas sobre a experiência que algumas empresas tiveram no processo de migração para o software livre.

Companhia Do Metropolitano De São Paulo - METRÔ

O Metrô de São Paulo é uma empresa vinculada Secretária do Estado de Transportes Metropolitanos (STM), com mais de 8.000 funcionários regidos pela CLT e obtém sua receita de maneira própria.

O Metrô de São Paulo vem adotado softwares livres desde 1999. Este processo iniciou-se quando o Metrô não conseguiu chegar em um acordo com a Microsoft para atualizações de seus softwares, devido ao alto preço pedido por este fornecedor.(Mazzarical, 2003c).

Dosso (2003), frisa que ao contrário de muitas empresas o Metrô iniciou seu processo de migração através dos Desktops, com a substituição do pacote MS-Office pelo StarOffice da Sun, um produto comercial, mais gratuito até 2002.

Gustavo Mazzarical em entrevista para Portal Popular (2002), coloca que chegou um momento em que as vantagens de adoção do StarOffice, no Metrô eram tão sensíveis que a transição se tornou inevitável. No entanto, foram necessários a realizações de diversos testes para garantir a compatibilidade e viabilidade do programa alternativo. Além disso, foi necessário convencer os funcionários a aceitarem o StartOffice, para isso, foram realizadas diversas palestras sobre o programa, foram criadas apostilas para cada programa que compõe o StarOffice, foi oferecido treinamento específico e por último, a empresa contratou três estudantes da PUC de São Paulo com a função de ajudar e familiarizar os empregados com esse novo programa.

Com fim da estratégia da SUN de oferecer o StarOffice de maneira gratuita, o Metrô migrou mais uma vez de aplicativo de escritório, adotando desta vez o OpenOffice, que é um projeto de software livre baseado no código e tecnologia do StarOffice liberado pela Sun Microsystem, que participa ativamente da comunidade OpenOffice.org.

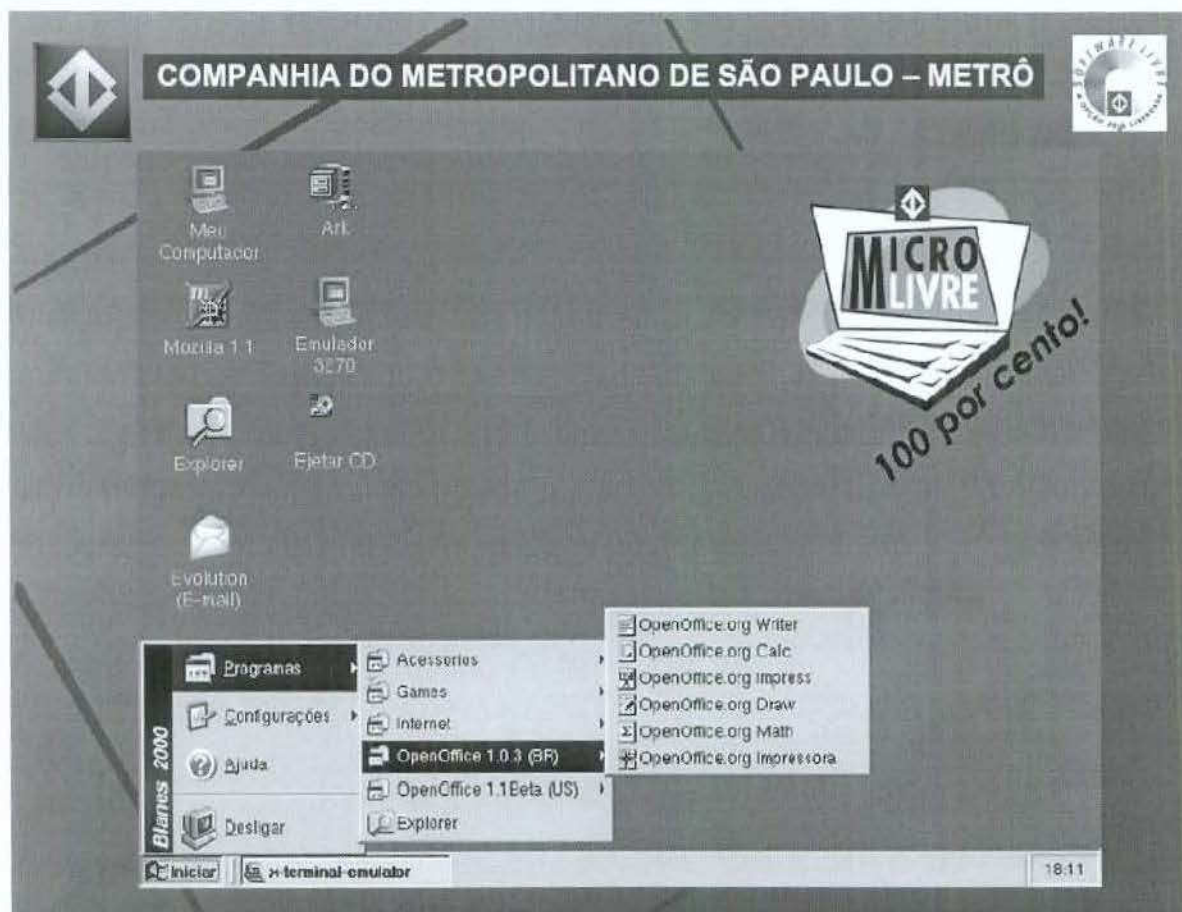
Com essas medias foi possível vencer a resistência inicial dos funcionários. Para tanto o Metrô já realizou até 2003, 1200 treinamentos para a migração do MS-Office para o StarOffice e OpenOffice, 400 treinamentos completos para o OpenOffice. Além disso, existem mais 700 pessoas em treinamento para o OpenOffice. (Mazzarical, 2003b).

Ainda em entrevista para o Portal Popular (2002), Mazzarical ressalta que a redução de custos conseguida pela adoção do software livre, foi possível investir em compra de novos equipamentos, treinamento e outros produtos. Isto só foi possível devido aos R\$ 700 mil anuais economizados pela opção de não continuar pagando pelo MS-Office.

O sucesso obtido com a migração dos aplicativos para escritório facilitou que novas iniciativas de substituições de software proprietário por livre fossem adotadas. Mazzarical (2003b) coloca que hoje o Metrô conta com um serviço de e-

mail baseado em software livre, o Metromail, e conta com alguns servidores rodando Linux. Além disso, existem os projetos Micro Livre e PostgreSQL, sendo que o primeiro busca dar treinamento e suporte para adoção do Linux em desktops. Para minimizar o impacto para os seus usuários, o Metrô optou utilizar uma interface gráfica muito parecida com o Windows como pode ser visto na figura 4.1:

Figura 4.1 – Sistema Linux rodando a interface gráfica Blanes 2000 utilizada no projeto Micro Livre do Metrô-SP



Fonte: Mazzarical (2003b)

Já o projeto PostgreSQL está sendo utilizado para aplicações de banco de dados entre os departamentos e estuda a viabilidade de utilização do Postgres como alternativa ao Oracle. Segundo Dosso (2003) já existem no Metrô diversas aplicações utilizando banco de dados livres como o Postgres e o Mysql.

Todas as iniciativas de utilização de software alternativos no Metrô de São Paulo rendem, segundo Mazzarical (2003a) para a empresa uma economia de aproximadamente três milhões de reais. Caso a empresa não obtivesse esta economia seria necessário um orçamento 30% maior que o atual para o setor de informática.

Devido ao seu pioneirismo o Metrô tornou-se modelo na comunidade software livre no Brasil, conseguindo grande destaque na mídia e eventos. Além disso, a empresa colabora com documentação gratuitas em português dos softwares livre que utiliza.

Por último Mazzarical (2003c) coloca que:

"Software livre, além de ser um novo estado de espírito, está alinhado com a sobrevivência das empresas, sejam elas públicas ou privadas. (...) É a forma de minimizar custos, otimizar soluções e garantir a qualidade."

Forças Armadas

Existem iniciativas para adoção do software livres no Exército, Marinha e Aeronáutica. A seguir serão descritas algumas experiências de cada corporação.

Exército

Em artigo para Revista do Linux nº 23 é discutido o caso do 19º Grupo de Artilharia de Campanha (19º GAC), que foi uma das unidades pioneiras a adotar o software livre no exército a partir de maio 2000.

Inicialmente foi elaborado um Plano Diretor de Informática pelo Escritório de Qualidade Total do 19º GAC para criação de uma Intranet, instalação de um Firewall, monitoramento dos usuários, FTP, correio eletrônico e acesso a Internet em todos os computadores da rede e instalação do pacote StarOffice. Pretendia-se com isso a economia de material de expediente no âmbito da Unidade.

A utilização de software livre pesou devido ao seu baixo custo de aquisição e por ser capaz de reaproveitar máquinas antigas do 19º GAC. Após diversos

testes o programa que mais se adequou com a estrutura da Unidade foi o Conectiva 6.0, pois se mostrou mais estável para os servidores e estações de trabalho.

O retorno do investimento foi obtido no curto prazo devido a economia em aquisições de softwares proprietários e pela redução de 60% com gastos de materiais para expediente que foram informatizadas.

O principal problema enfrentado pelos idealizadores do plano foi à resistência dos usuários, que estavam familiarizados com os softwares proprietários e em muitos casos já tinham realizado cursos para os programas fora da corporação. A solução foi dar treinamento e suporte gratuitos dentro da Unidade, com módulos do StarOffice, correio eletrônico, digitação e também diversas palestras de conscientização dos motivos pelos quais a Unidade estava utilizando os novos sistemas e aplicativos.

A experiência da 19ª GAV serviu para capacitar os militares temporários para um novo futuro emprego civil, além de servir como modelo de outras Unidades do Exército, mostrando as principais dificuldades e vantagens na adoção do software livre.

O capitão Wildison Jone Jobim de Souza (2003) (em entrevista realizada para o autor), Chefe da Divisão de Tecnologia da Informação da Escola Preparatória de Cadetes do Exército (Especeex), existe uma tendência de migração do software proprietário para o livre em todo o Exército devido em grande parte aos altos custos dos softwares proprietários.

Segundo a mesma fonte, a Escola Preparatória de Cadetes do Exército já possui parte dos seus servidores rodando Linux e há uma tendência para migração das estações de trabalho para sistemas livres. A principal dificuldade para que a migração se concretize é a necessidade de treinamento do pessoal da área de informática com os novos programas e a falta de tempo do pessoal especializado para que a migração seja realizada.

Aeronáutica

Segundo a Revista do Linux número 31, o Instituto de Proteção ao Voo (IPV), organização do comando da Aeronáutica, devido a recursos limitados, deixou de adquirir novas licenças de software proprietário e passou a utilizar software livre em seu sistema. Esta estratégia economizou duzentos mil reais até o final de 2001 para o IPV.

Além da falta de recursos para aquisições de novos softwares, o Instituto enfrentava problemas de falta robustez dos servidores de rede, que necessitavam de manutenção e, freqüentemente, de reinstalação de todos os serviços da máquina.

O Major-Aviador Ricardo Rangel coloca para a Revista do Linux que a fragilidade dos sistemas operacionais Windows NT instalados nos servidores, não protegia a rede de acessos indevidos, o que tornava o site do IPV, os serviços de redes e os acessos às bases de dados extremamente vulneráveis.

A migração iniciou-se pelos servidores que receberam o Conectiva Linux 6.0/7.0 e ferramentas para serviços de e-mail (sendmail), serviços de web (Apache), serviços para controle de acesso (ipchains e iptables), banco de dados (Mysql) e para compartilhamento de arquivos com estações Windows (SAMBA).

Segundo Rangel, a migração dos servidores para Linux foi um sucesso, sendo que os serviços oferecidos tornaram-se mais estáveis, houve aumento do nível de segurança da rede e do acesso ao banco de dados. Além disso, os danos causados por ataques de vírus se reduziram a zero e a taxa de tempo de manutenção caiu significativamente.

Com o sucesso da migração dos servidores e após serem realizados testes durante um ano, o IPV começou a mudar suas estações de trabalho para Linux, instalando ferramentas como GIMP, StarOffice, SAMBA, etc.

De acordo com Rangel nas estações de trabalho houve redução da taxa de manutenção, diminuição de acessos indevidos e não ocorreram mais problemas causados por vírus.

Para apoiar a migração foram realizados treinamentos com empresas especializadas, foram realizadas palestras e mini-cursos para familiarizar os usuários e minimizar as rejeições com o novo sistema. O treinamento durou 120 horas, tendo treinado técnicos e usuários e teve custo de quinze mil reais.

Estas não foram às únicas iniciativas da Aeronáutica na adoção do Linux. No início de agosto de 2003 o Centro de Computação da Aeronáutica substituiu o servidor Microsoft Exchange pelo software nacional e livre Direito, desenvolvido pela Companhia de Processamento de Dados do Rio Grande do Sul (Procergs).

Marinha

Em palestra proferida no Ministério de Ciência e Tecnologia, o Comandante José Carlos Amado (Amado, 2003), informou que a Marinha possui o Departamento de Administração da Marinha (DADM), responsável pela implantação do software livre nesta instituição e o Departamento de Telecomunicação da Marinha (DTM), responsável por toda estrutura de redes (segurança, comunicação e criptografia) para comunicação desta instituição. Estes departamentos estão desenvolvendo o Comando de Operações Navais para Linux (ComOpNav-Linux).

Amado ressalta que os fatores que motivaram a implementação de software livre no Comando de Operações Navais foram:

- Expansão do número dos distritos navais
- Controle tecnológico
- Auditabilidade do código fonte
- Apoio da comunidade acadêmica

A expansão de programas para troca de dados entre os diversos distritos navais e o comando no Rio de Janeiro implica em gastos com o aumento do número de aquisições de softwares. Assim sendo, a adoção do software livre permitiria a expansão sem custos de aquisição de software.

O Capitão frisa que as Forças Armadas não podem depender de firmas, não podem existir sistemas militares aprisionados a empresas. Isso faz com que o

domínio, o controle da tecnologia e a editabilidade do código tenham fundamental importância.

Outra vantagem do software livre é o acesso ao seu código fonte, sendo possível auditá-lo. José Carlos coloca que nada garante que uma solução proprietária não tenha uma backdoor¹³ que permita que dados enviados pela Marinha sejam espionados.

Para conseguir apoio para a migração, Amado, coloca que é preciso conseguir credibilidade. Para isso, foi criada toda uma metodologia própria para a migração constituída por: montagem um projeto de migração, levantamento de requisitos, realizações de treinamento para suporte e usuários, levantamento das necessidades dos usuários, realizações de testes, verificação de todos os softwares utilizados e elaboração de um cronograma de trabalho.

Foram identificados dois problemas que poderiam inviabilizar a migração: utilização do Lotus Notes e a criptografia de arquivos. O primeiro problema foi solucionado através da utilização do WINE, onde foi possível alcançar uma funcionalidade de todos os recursos críticos. Já o segundo problema foi resolvido com a criação de um programa específico para Linux que escolhe e criptografa um arquivo.

Para diminuir o impacto inicial da substituição do software proprietário por livre estão sendo realizados treinamentos do pessoal de suporte e usuários, explicações sobre o Linux e suas características.

Os resultados alçados até o outubro de 2003 haviam sido: credibilidade do processo de migração, motivação dos usuários, ampla documentação do projeto, cd para instalação e estação de trabalho "piloto" (Pentium 166 com 64MB de memória RAM).

Espera-se alcançar a elaboração da documentação completa para instalação, treinamento para toda a equipe de suporte e usuários e instalação do Linux Conectiva 9 em todas estações de trabalho.

Por último Amado coloca que:

¹³ Portas do fundo – brechas em programas que permitem aos conhecedores da brecha terem acesso indevido aos dados.

“O trinômio auditabilidade + domínio tecnológico + preço influi drasticamente na adoção do Software Livre como plataforma de software para o Setor Público. Vale a pena investir.”

Viação Graciosa

Na Revista do Linux número 31 é estudado o caso da Viação Graciosa, empresa paranaense de ônibus interestaduais, onde a adoção Linux permitiu um maior desempenho da transmissão de dados entre filiais e as estações de trabalho antigas puderam ser aproveitadas.

A empresa necessitava resolver problemas de comunicação entre as filiais e as garagens. Isso somente seria possível mudando a estrutura de informática da empresa, que contava com rede Novell e banco de dado em ambiente DOS.

O GNU/Linux não foi a iniciativa inicial da empresa para atualização de sua infra-estrutura. Inicialmente uma solução tendo como Windows NT foi analisada, mas teria demandado um alto investimento devido à compra de licenças, atualizações de hardware das estações de trabalho e compra de servidores caros.

Ainda antes da adoção do Linux a empresa chegou a testar por dois meses outra alternativa, a utilização de um software pronto especializado para empresas de transporte. No entanto, o software mostrou-se inflexível e a empresa precisaria um servidor de preço médio de R\$ 18 mil, além de um servidor de R\$ 27 mil que já havia comprado. Devido ao alto custo das outras opções, o Linux foi estudado e adotado. O sistema livre atendeu as necessidades da empresa, tendo conseguido alta performance com um servidor de apenas R\$ 3 mil, as estações de trabalho puderam ser reaproveitadas e a migração do banco de dados Dataflex em ambiente DOS para o Dataflex em Linux foi feita de maneira simples e segura.

Além de resolver os problemas de comunicação entre as filiais e as garagens, a adoção do Linux permitiu a criação de um sistema de vendas on-line. Este sistema agilizou a empresa uma vez que houve uma redução nos erros de emissão de passagens, e ainda permitiu a empresa saber o número de

passageiros em tempo real, permitindo a criação de horários extra e adequação do tamanho do ônibus a ser utilizado.

Matheus Schaedler Uhlmann, diretor adjunto de sistemas da Graciosa, para a Revista do Linux, o menor tempo para se processar o relatório estatístico, que com o sistema antigo levava quatro horas e meia com Linux passou para menos de cinco minutos, mudou a interação dos gerentes, possibilitando que as decisões possam ser tomadas de maneira mais dinâmica.

A Graciosa também criou um sistema de correio eletrônico interno baseado em software livre que permitiu a empresa uma melhor integração entre os setores e uma redução significativa no custo com ligações telefônicas.

Amazon.com

Segundo Planetarium (2003), a Amazon.com, umas das maiores lojas virtuais do mundo, migrou o seu sistema para Linux em junho de 2001.

Com esta medida a empresa conseguiu, segundo IDG Now (2003), economizar aproximadamente 25% dos gastos com tecnologia. A empresa que gastou US\$ 71 milhões durante o terceiro trimestre de 2000, teve que investir apenas US\$ 54 milhões no mesmo período de 2001.

Planetarium (2003), ressalta que esta economia foi possível devido à redução dos gastos com aquisição de softwares e pelo fato do Linux conseguir alcançar um desempenho satisfatório com uma infra-estrutura mais barata do que a necessária para rodar o mesmo serviço com softwares proprietários.

Toyota

Em artigo publicado na Revista do Linux número 29, a Toyota no Brasil utiliza Linux desde 1998. Hoje o Linux já está presente nas três unidades existentes no Brasil: a comercial, na capital paulista, a matriz, em São Bernardo do Campo e a montadora em Indaiatuba, no estado de São Paulo. O Linux é utilizado

na empresa para servidor de Internet, servidor de e-mail, servidor DNS¹⁴ e Firewall¹⁵.

A implementação do Linux recebeu apoio da diretoria, não encontrando resistência alguma por parte desta. Isso porque a diretoria entende o Linux como confiável e ideal para operações críticas que não podem parar sob pena de causar grandes prejuízos.

Outro fator que pesou na escolha do Linux foi a economia obtida com a redução de gastos com aquisições de licenças e o sistema livre diminuiu a necessidade de atualizações de hardware. Além disso, por ser um sistema mais estável a necessidade de manutenção foi bastante reduzida, o que significa a que o sistema pode ser mantido por um número menor de pessoal.

O processo de migração não exigiu gastos com treinamento, pois os funcionários técnicos buscaram aprender o Linux junto a comunidade Livre e através de tutoriais disponíveis na Internet.

Prefeitura de Rio das Ostras

Segundo a Revista do Linux número 33, a prefeitura de Rio das Ostras está utilizando Software Livre desde de agosto de 2001, quando foi criada a Assessoria de Informática para estudar a viabilidade da instalação de programas livres nos servidores e estações de trabalho da prefeitura.

As vantagens observadas pela equipe da prefeitura foram: melhor performance e segurança, códigos fontes abertos e livres para alteração, maior integração de dados, redução de custo em aquisição de licenças de softwares proprietários e diminuição do tempo com manutenção.

Pesou positivamente para o software livre o fato de ser possível reaproveitar máquinas consideradas obsoletas. Foram reaproveitados 10 computadores que são utilizados em projetos para inclusão digital pela prefeitura.

¹⁴ Servidores que transformam o nome de uma máquina em seu endereço. Por exemplo, ao digitar www.unicamp.br, este endereço escrito é transformado no endereço numérico da máquina.

¹⁵ Sistema que visa controlar tudo que entre e sai pela rede, barrando acessos indevidos.

A prefeitura migrou todos os servidores e várias estações de trabalho foram migrados para o GNU/Linux com StarOffice, existindo pelo menos um computador rodando somente software livre em cada secretaria. A secretária de Guarda e Trânsito esta utilizando somente software livre.

Segundo a Revista do Linux, a prefeitura possuía até o final de 2002 todos os seus servidores rodando em Linux, 20% das estações de trabalho rodando somente software livre e 40% dos equipamentos rodando StarOffice.

As principais dificuldades para adoção do software livre, segundo a Prefeitura de Rios das Ostras (2003), foram: a mudança de culturas dos usuários e gerência, adaptação à nova tecnologia, migração de dados antigos e suporte.

Para enfrentar estas dificuldades a equipe da prefeitura criou uma nova interface gráfica para o Linux que obtivesse uma melhor adaptação por parte dos usuários. Além disso, treinou boa parte dos seus funcionários para habilitá-los a utilizarem o Linux e o StartOffice.

A economia gerada até 2002, segundo a Prefeitura de Rio das Ostras (2003) foi de aproximadamente R\$ 422,4 mil. Com esta economia o município esta disponibilizando um laboratório para treinamento de funcionários da Prefeitura e outro para combate da exclusão digital na comunidade.

Segundo Marcos Vinicius P. Marini, assessor de Informática da prefeitura de Rio das Ostras, para Revista do Linux:

"Para um munícipe é melhor que seja investido R\$ 1.600 em livros ou remédios do que em um software. Afinal, os gestores públicos devem aplicar naquilo que é mais econômico e eficiente".

Conclusão

A questão inicial que motivou este trabalho era entender a razão pela qual, mesmo existindo softwares livres, empresas, organizações e pessoas continuavam usando softwares proprietários. Também interessava entender as condições que seriam necessárias para que houvesse migração para o software livre. Para estudar o assunto foi necessário formular um enfoque interdisciplinar. Parte das explicações veio do campo da economia, de onde foi entendido o conceito de aprisionamento tecnológico, enunciado por Shapiro e Varian, que mostra as razões das dificuldades de sair da situação de aprisionamento.

Outra parte das explicações veio do campo da computação, onde foram estudados os aspectos técnicos e de qualidade de software, que permitiram entender, tanto a natureza técnica dos aprisionamentos, como os esforços que os desenvolvedores de software livre estão fazendo para diminuir o custo que as empresas, organizações e pessoas, tem para sair da situação de aprisionamento.

Também foi verificado que o software livre nada tem a dever quanto a padrão de qualidade ao software proprietário. Ao contrário, muitas vezes tem sido considerado mais seguro e robusto que o software proprietário.

Finalmente os estudos de casos, mostraram as dificuldades que empresas e organizações de diversos tipos tiveram na migração, mas ficou claro, em todos os casos, as vantagens obtidas, não somente quanto a redução de custos, como também quanto a qualidade, estabilidade, menor necessidade de manutenção, eliminação dos ataques por vírus e possibilidade de reutilizar equipamentos sucateados que não acompanhavam a demanda de hardware dos softwares proprietário. Além disso, foram observadas vantagens de interesse específico (caso de segurança nacional para a Marinha).

Os estudos de casos nos levaram, inclusive, a revisar a equação (2) do capítulo 3, que indica o valor atual da posse do software proprietário, acrescentando mais uma variável de custo (m), o custo de manutenção do

software proprietário em relação aos problemas de instabilidade, manutenção, ataque de vírus.

Assim a equação (2) deve ser expressa como:

$$VAp = l+r+tp + m \quad (4)$$

Pode-se também, repensar a equação (3) diminuindo do custo da migração para o software livre a necessidade de compra de novos equipamentos (h), dada a possibilidade de reaproveitar hardware relativamente obsoleto. Portanto a equação 3 pode ser enunciada como:

$$VAI = tl + a - h \quad (5)$$

Ou seja, a análise dos casos mostra, que além das vantagens de qualidade, o custo de migração pode ser menor que aquele esperado a partir do enunciado original das variáveis das equações 2 e 3.

No entanto, o software livre enfrenta difivuldade para ser adota em parte devido a resistência a mudanças dos usuários e gerentes, ao feedback positivo que o software proriitário usufrui e outros tipos de aprisionamento que foram discutidos ao longo do trabalho.

Apesar disso, com o passar do tempo e a maior participação do software livre nas instituições, se espera o surgimento de um número maior de usuários capacitados a utilizar software livre, que farão com que este usufrua os efeitos positivos do feedback.

Bibliografia

AMADO, José Carlos. **Primeira Oficina de Software Livre – Apresentação Experiência da Marinha Brasileira.** Disponível em: <<http://www.mct.gov.br/temas/sl/wmv/softwarelivre-8.wmv>>. Acesso em 03 de novembro de 2003.

AMADO, José Carlos. **Primeira Oficina de Software Livre – Apresentação Experiência da Marinha Brasileira em PDF.** Disponível em: <http://www.mct.gov.br/temas/sl/pdf/sl_marinha.pdf>. Acesso em 03 de novembro de 2003.

ASCENSO, Mariana F., SANTOS, Pedro Miguel de S. **História e desenvolvimento do Sistema Operativo Unix.** Disponível em: <<http://www.eq.uc.pt/~pmg3/Unix1.htm>>. Acesso em 03 de setembro de 2003.

AUSTRIN, Rodrigo. **Crônica de uma experiência bem sucedida:** O metro de São Paulo foi uma das primeiras grandes empresas públicas a adotar Software Livre e é um exemplo genuíno de convivência pacífica e gradual entre soluções proprietárias e livre. **Revista do Linux**, nº 34, outubro de 2002. Disponível em <<http://www.revistadolinux.com.br/ed/034/assinantes/especial.php3>>. Acesso em 26 de março de 2003.

BANCO CENTRAL. **Balanco de Pagamento (série histórica).** Disponível em: <http://www.bcb.gov.br/ftp/notaecon/balpagm.zip>. Acesso em 01 de junho de 2003.

CORREIO BRASILIENSE. **Para economizar nosso dinheiro.** Disponível em:
http://www2.correioweb.com.br/cw/edicao_20030819/sup_info_190803_40.htm.
Acesso em 01 de novembro de 2003.

DEGHI, Gilmar J. **Os Custos Ocultos da Informática.** IV SEME AD, 1999
(deghe.pdf).

Dosso, Luís. **Software Livre e Metro SP.** Entrevista realizada em 16/10/2003.

GHEMAWAT, Pankaj **A Estratégia e o Cenário dos Negócios.** Porto Alegre:
Bookman, 2000.

GRAEMIL, Alexandre R. **Sistemas de Informação: O Alinhamento da Estratégia de TI com a Estratégia Corporativa.** São Paulo: Atlas, 2003.

HAUBEN, Ronda. **Unix History.** Disponível em
<<http://www.unixsup.com/unixlinux/historiaunixin.html>>. Acesso em 14 de julho de 2003.

HEWLETT-PACKARD COMPANY. **Empresa decidiu aumentar seu data warehouse para atender o crescimento atual e futuro.** Disponível em:
<http://www.hp.com.br/casos/caso.php?caso_id=26&pagina=descricao>.
Acesso em 7 de setembro de 2003.

IDGNow. **Amazon adere ao Linux e economiza 25%.** Disponível em:
<<http://idgnow.terra.com.br/idgnow/corporate/2001/10/0083>>. Acesso em 12 de setembro de 2003.

KOTLER, Philip; FOX, Karen F.A. **Marketing estratégico para instituições de ensino.** São Paulo: Atlas, 1994.

LINUXSHOP. **Amazon The table of equivalents / replacements / analogs of Windows software in Linux.** Disponível em: <<http://linuxshop.ru/linuxbegin/win-lin-soft-en/table.shtml>>. Acesso em 7 de setembro de 2003.

LOTTERMANN, Jair. **Viabilidade da Utilização de Software Livre em um Ambiente Empresarial.** Dissertação de Mestrado. Puc-Campinas, 2001.

MANÇAN, Eduardo. **Opinião: Com licença? Obrigado! Revista do Linux**, nº 34, outubro de 2002. Disponível em <<http://www.revistadolinux.com.br/ed/034/assinantes/especial.php3>>. Acesso em 11 de setembro de 2003.

MARINI, Marcos V. P. **Cases de sucesso - Prefeitura Municipal de Rio das Ostras - RJ.** Disponível em: <http://www.linuxbr.org.br/cases_pre_riostras.htm>. Acesso em 02 de novembro de 2003.

MAZZARICAL, Gustavo C. de Q. **Experiência do Metrô de São Paul.** Disponível em: <http://www.openoffice.unicamp.br/apresentacao_metro.html>. Acesso em 13 de outubro de 2003c.

MAZZARICAL, Gustavo C. de Q. **Primeira Oficina de Software Livre – Apresentação Experiência do Metrô de São Paulo.** Disponível em: <<http://www.mct.gov.br/Temas/sl/wmv/SoftwareLivre-11.wmv>>. Acesso em 03 de novembro de 2003a.

MAZZARICAL, Gustavo C. de Q. **Primeira Oficina de Software Livre – Apresentação Experiência do Metrô de São Paulo em PDF**. Disponível em: <http://www.mct.gov.br/Temas/sl/pdf/sl_metro.pdf>. Acesso em 03 de novembro de 2003b.

MEDEIROS, Elizabet M. S., SAUVÉ, Jacques P. **Sistemas Avaliação do Impacto de Tecnologias da Informação Emergentes nas Empresas**. Rio de Janeiro: Qualitymark, 2003.

MIT- SOFTEX (2002). **A indústria de Software no Brasil -2002**. Campinas: SOFTEX, 2002.

PLANETARIUM. **Amazon economiza US\$ 17 mi com adoção do Linux**. Disponível em: <<http://www.planetarium.com.br/planetarium/noticias/2001/10/1004532041>>. Acesso em 12 de setembro de 2003.

PORTAL POPULAR. **A experiência do Metrô com softwares livres**. Disponível em: <<http://WWW.PORTALPOPULAR.ORG/POLITICA/BRASIL/BRASIL-48.HTM>>. Acesso em 12 de setembro de 2003.

PORTER, Michael E. **Estratégia Competitiva: Técnicas para Análise de Indústrias e da Concorrência**. Rio de Janeiro: Campus, 1980.

PREFEITURA DE RIO DAS OSTRAS. **Projeto Software Livre**. Disponível em: <<http://livre.pmro.rj.gov.br/modules.php?name=Content&pa=showpage&pid=1>>. Acesso em 4 de setembro de 2003.

REVISTA DO LINUX. **Com uma série de projetos de lei engatilhados para este ano, o Rio Grande do Sul se confirma como a Meca do software livre no país,** nº 20, agosto de 2001. Disponível em: <<http://www.revistadolinux.com.br/ed/020/assinantes/especial.php3>>. Acesso em 01 de setembro de 2003.

REVISTA DO LINUX. **ESTUDO DE CASO – Braço forte, mão amiga: Eliminar barreiras no aprendizado de ferramentas livres foi o principal obstáculo para o Exército,** nº 23, novembro de 2001. Disponível em: <<http://www.revistadolinux.com.br/ed/023/assinantes/caso.php3>>. Acesso em 15 de outubro de 2003.

REVISTA DO LINUX. **ESTUDO DE CASO – A Toyota do Brasil adota (e adora) o pingüim. O motivo? O sistema é altamente confiável nas missões críticas,** nº 29, maio de 2002. Disponível em: <http://www.revistadolinux.com.br/ed/029/assinantes/caso_toyota.php3>. Acesso em 12 de outubro de 2003.

REVISTA DO LINUX. **SERVIÇO PÚBLICO – A Prefeitura de Rio das Ostras, no Rio de Janeiro, economizará R\$ 422 mil com Linux e investe em programas sociais de capacitação e inclusão digital,** nº 33, setembro de 2002. Disponível em: <http://www.revistadolinux.com.br/ed/033/assinantes/servico_publico.php3>. Acesso em 9 de outubro de 2003.

REVISTA DO LINUX. **SERVIÇO PÚBLICO – Veja como o Instituto de Proteção ao Voo (IPV), do Comando da Aeronáutica, conseguiu reduzir custos operacionais e aumentar a segurança de seus dados,** nº 31, julho de 2002. Disponível em: <http://www.revistadolinux.com.br/ed/031/assinantes/servico_publico.php3>. Acesso em 16 de outubro de 2003.

SANTOS, Enarni M. **Aprisionamento Tecnológico: Novos desafios da gestão das estratégias organizacionais na era da informação.** Disponível em: <<http://www.ead.fea.usp.br/cad-pesq/arquivos/v08-1art06.pdf>>. Acesso em 21 de abril de 2003.

SHANKLAND, Stephen, KANE, Margaret, LEMOS, Robert. **How Linux saved Amazon millions.** Disponível em: <<http://news.com.com/2100-1001-275155.html?legacy=cnet>>. Acesso em 03 de novembro de 2003.

SHAPIRO, Carl, VARIAN, Hal. **A economia da informação: Como os princípios econômicos se aplicam à era da Internet.** Rio de Janeiro: Campus, 1999.

SOUZA, Wildison J. J. **Software Livre e Especex.** Entrevista realizada pelo autor em 10/11/2003.

STALLMAN, Richard. **Linux e o sistema GNU.** Disponível em <<http://www.geocities.com/CollegePark/Union/3590/linuxgnu.html>>. Acesso em 16 de agosto de 2003.

STALLMAN, Richard. **Projeto Gnu.** CIPSGA – Comitê de Incentivo a Produção do Software Gratuito e Alternativo. Disponível em: <<http://www.cipsga.org.br>>. Acesso em 27 de abril de 2003.

TANENBAUM, Andrew S. **Sistemas Operacionais Modernos.** Rio de Janeiro: Livros técnicos e Científicos Editora S.A., 1999.

TERRA. **Linux é 74% mais estável que Windows, aponta teste.** Disponível em: <<http://informatica.terra.com.br/interna/0,,OI210362-EI553,00.html>>. Acesso em 9 de novembro de 2003.

UNIXSUPPORT. **Historias de Unix.** Disponível em:
<<http://www.unixsup.com/unixlinux/historiaunix.html>>. Acesso em 4 de junho de 2003.

UNIXSUPPORT. **Línea Del Tiempo de Linux.** Disponível em:
<<http://www.unixsup.com/unixlinux/historialinux.html>>. Acesso em 4 de junho de 2003.

UTTERBACK, James. **Dominando a Dinâmica da Inovação.** Rio de Janeiro: Qualitymark Editora Ltda., 1996.

VARGAS, Nilson. Travou? Ele Resolve. **Veja On-line**, edição 1668, 27 de dezembro de 2000. Disponível em: <http://veja.abril.com.br/270900/p_088.html>. Acesso em 01 de abril de 2003.

VASCONCELOS, L. A. **Os ciclos relevantes do processo de produção e a gestão estratégica de custos: breve análise do processo de custeio nos setores de produção de Software e de Serviços de Informática.** Trabalho apresentado ao: VI Congresso Internacional de Custos, 1999, Braga.

Anexos

Anexo 1 – Lei do Software

Lei de Software

Lei nº 9.609, de 19 de fevereiro de 1998

Dispõe sobre a proteção da propriedade intelectual de programa de computador, sua comercialização no País, e dá outras providências.

O PRESIDENTE DA REPÚBLICA

Faço saber que o Congresso Nacional decreta e eu sanciono a seguinte Lei:

CAPÍTULO I

DISPOSIÇÕES PRELIMINARES

Art. 1º Programa de computador é a expressão de um conjunto organizado de instruções em linguagem natural ou codificada, contida em suporte físico de qualquer natureza, de emprego necessário em máquinas automáticas de tratamento da informação, dispositivos, instrumentos ou equipamentos periféricos, baseados em técnica digital ou análoga, para fazê-los funcionar de modo e para fins determinados.

CAPÍTULO II

DA PROTEÇÃO AOS DIREITOS DE AUTOR E DO REGISTRO

Art. 2º O regime de proteção à propriedade intelectual de programa de computador é o conferido às obras literárias pela legislação de direitos autorais e conexos vigentes no País, observado o disposto nesta Lei.

§ 1º Não se aplicam ao programa de computador as disposições relativas aos direitos morais, ressalvado, a qualquer tempo, o direito do autor de reivindicar a paternidade do programa de computador e o direito do autor de opor-se a alterações não-autorizadas, quando estas impliquem deformação, mutilação ou

outra modificação do programa de computador, que prejudiquem a sua honra ou a sua reputação.

§ 2º Fica assegurada a tutela dos direitos relativos a programa de computador pelo prazo de cinquenta anos, contados a partir de 1º de janeiro do ano subsequente ao da sua publicação ou, na ausência desta, da sua criação.

§ 3º A proteção aos direitos de que trata esta Lei independe de registro.

§ 4º Os direitos atribuídos por esta Lei ficam assegurados aos estrangeiros domiciliados no exterior, desde que o país de origem do programa conceda, aos brasileiros e estrangeiros domiciliados no Brasil, direitos equivalentes.

§ 5º Inclui-se dentre os direitos assegurados por esta Lei e pela legislação de direitos autorais e conexos vigentes no País aquele direito exclusivo de autorizar ou proibir o aluguel comercial, não sendo esse direito exaurível pela venda, licença ou outra forma de transferência da cópia do programa.

§ 6º O disposto no parágrafo anterior não se aplica aos casos em que o programa em si não seja objeto essencial do aluguel.

Art. 3º Os programas de computador poderão, a critério do titular, ser registrados em órgão ou entidade a ser designado por ato do Poder Executivo, por iniciativa do Ministério responsável pela política de ciência e tecnologia.

§ 1º O pedido de registro estabelecido neste artigo deverá conter, pelo menos, as seguintes informações:

I - os dados referentes ao autor do programa de computador e ao titular, se distinto do autor, sejam pessoas físicas ou jurídicas;

II - a identificação e descrição funcional do programa de computador; e

III - os trechos do programa e outros dados que se considerar suficientes para identificá-lo e caracterizar sua originalidade, ressalvando-se os direitos de terceiros e a responsabilidade do Governo.

§ 2º As informações referidas no inciso III do parágrafo anterior são de caráter sigiloso, não podendo ser reveladas, salvo por ordem judicial ou a requerimento do próprio titular.

Art. 4º Salvo estipulação em contrário, pertencerão exclusivamente ao empregador, contratante de serviços ou órgão público, os direitos relativos ao programa de computador, desenvolvido e elaborado durante a vigência de contrato ou de vínculo estatutário, expressamente destinado à pesquisa e desenvolvimento, ou em que a atividade do empregado, contratado de serviço ou servidor seja prevista, ou ainda, que decorra da própria natureza dos encargos concernentes a esses vínculos.

§ 1º Ressalvado ajuste em contrário, a compensação do trabalho ou serviço prestado limitar-se-á à remuneração ou ao salário convencionado.

§ 2º Pertencerão, com exclusividade, ao empregado, contratado de serviço ou servidor os direitos concernentes a programa de computador gerado sem relação com o contrato de trabalho, prestação de serviços ou vínculo estatutário, e sem a utilização de recursos, informações tecnológicas, segredos industriais e de negócios, materiais, instalações ou equipamentos do empregador, da empresa ou entidade com a qual o empregador mantenha contrato de prestação de serviços ou assemelhados, do contratante de serviços ou órgão público.

§ 3º O tratamento previsto neste artigo será aplicado nos casos em que o programa de computador for desenvolvido por bolsistas, estagiários e assemelhados.

Art. 5º Os direitos sobre as derivações autorizadas pelo titular dos direitos de programa de computador, inclusive sua exploração econômica, pertencerão à pessoa autorizada que as fizer, salvo estipulação contratual em contrário.

Art. 6º Não constituem ofensa aos direitos do titular de programa de computador:

I - a reprodução, em um só exemplar, de cópia legitimamente adquirida, desde que se destine à cópia de salvaguarda ou armazenamento eletrônico, hipótese em que o exemplar original servirá de salvaguarda;

II - a citação parcial do programa, para fins didáticos, desde que identificados o programa e o titular dos direitos respectivos;

III - a ocorrência de semelhança de programa a outro, preexistente, quando se der por força das características funcionais de sua aplicação, da observância de preceitos normativos e técnicos, ou de limitação de forma alternativa para a sua expressão;

IV - a integração de um programa, mantendo-se suas características essenciais, a um sistema aplicativo ou operacional, tecnicamente indispensável às necessidades do usuário, desde que para o uso exclusivo de quem a promoveu.

CAPÍTULO III

DAS GARANTIAS AOS USUÁRIOS DE PROGRAMA DE COMPUTADOR

Art. 7º O contrato de licença de uso de programa de computador, o documento fiscal correspondente, os suportes físicos do programa ou as respectivas embalagens deverão consignar, de forma facilmente legível pelo usuário, o prazo de validade técnica da versão comercializada.

Art. 8º Aquele que comercializar programa de computador, quer seja titular dos direitos do programa, quer seja titular dos direitos de comercialização, fica obrigado, no território nacional, durante o prazo de validade técnica da respectiva versão, a assegurar aos respectivos usuários a prestação de serviços técnicos complementares relativos ao adequado funcionamento do programa, consideradas as suas especificações.

Parágrafo único. A obrigação persistirá no caso de retirada de circulação comercial do programa de computador durante o prazo de validade, salvo justa indenização de eventuais prejuízos causados a terceiros.

CAPÍTULO IV

DOS CONTRATOS DE LICENÇA DE USO, DE COMERCIALIZAÇÃO E DE TRANSFERÊNCIA DE TECNOLOGIA

Art. 9º O uso de programa de computador no País será objeto de contrato de licença.

Parágrafo único. Na hipótese de eventual inexistência do contrato referido no *caput* deste artigo, o documento fiscal relativo à aquisição ou licenciamento de cópia servirá para comprovação da regularidade do seu uso.

Art. 10º Os atos e contratos de licença de direitos de comercialização referentes a programas de computador de origem externa deverão fixar, quanto aos tributos e encargos exigíveis, a responsabilidade pelos respectivos pagamentos e estabelecerão a remuneração do titular dos direitos de programa de computador residente ou domiciliado no exterior.

§ 1º Serão nulas as cláusulas que:

I - limitem a produção, a distribuição ou a comercialização, em violação às disposições normativas em vigor;

II - eximam qualquer dos contratantes das responsabilidades por eventuais ações de terceiros, decorrentes de vícios, defeitos ou violação de direitos de autor.

§ 2º O remetente do correspondente valor em moeda estrangeira, em pagamento da remuneração da que se trata, conservará em seu poder, pelo prazo de cinco anos, todos os documentos necessários à comprovação da licitude das remessas e da sua conformidade ao *caput* deste artigo.

Art. 11º Nos casos de transferência de tecnologia de programa de computador, o Instituto Nacional da Propriedade Industrial fará o registro dos respectivos contratos, para que produzam efeitos em relação a terceiros.

Parágrafo único. Para o registro de que trata este artigo, é obrigatória a entrega, por parte do fornecedor ao receptor de tecnologia, da documentação completa, em especial do código-fonte comentado, memorial descritivo, especificações funcionais internas, diagramas, fluxogramas e outros dados técnicos necessários à absorção da tecnologia.

CAPÍTULO V

DAS INFRAÇÕES E DAS PENALIDADES

Art. 12º Violar direitos de autor de programa de computador:

Pena - Detenção de seis meses a dois anos ou multa.

§ 1º Se a violação consistir na reprodução, por qualquer meio, de programa de computador, no todo ou em parte, para fins de comércio, sem autorização expressa do autor ou de quem o represente:

Pena - Reclusão de um a quatro anos e multa.

§ 2º Na mesma pena do parágrafo anterior incorre quem vende, expõe à venda, introduz no País, adquire, oculta ou tem em depósito, para fins de comércio, original ou cópia de programa de computador, produzido com violação de direito autoral.

§ 3º Nos crimes previstos neste artigo, somente se procede mediante queixa, salvo:

I - quando praticados em prejuízo de entidade de direito público, autarquia, empresa pública, sociedade de economia mista ou fundação instituída pelo poder público;

II - quando, em decorrência de ato delituoso, resultar sonegação fiscal, perda de arrecadação tributária ou prática de quaisquer dos crimes contra a ordem tributária ou contra as relações de consumo.

§ 4º No caso do inciso II do parágrafo anterior, a exigibilidade do tributo, ou contribuição social e qualquer acessório, processar-se-á independentemente de representação.

Art. 13º A ação penal e as diligências preliminares de busca e apreensão, nos casos de violação de direito de autor de programa de computador, serão precedidas de vistoria, podendo o juiz ordenar a apreensão das cópias produzidas ou comercializadas com violação de direito de autor, suas versões e derivações, em poder do infrator ou de quem as esteja expondo, mantendo em depósito, reproduzindo ou comercializando.

Art. 14º Independentemente da ação penal, o prejudicado poderá intentar ação para proibir ao infrator a prática do ato incriminado, com cominação de pena pecuniária para o caso de transgressão do preceito.

§ 1º A ação de abstenção de prática de ato poderá ser cumulada com a de perdas e danos pelos prejuízos decorrentes da infração.

§ 2º Independentemente de ação cautelar preparatória, o juiz poderá conceder medida liminar proibindo ao infrator a prática do ato incriminado, nos termos deste artigo.

§ 3º Nos procedimentos cíveis, as medidas cautelares de busca e apreensão observarão o disposto no artigo anterior.

§ 4º Na hipótese de serem apresentadas, em juízo, para a defesa dos interesses de qualquer das partes, informações que se caracterizem como confidenciais, deverá o juiz determinar que o processo prossiga em segredo de justiça, vedado o uso de tais informações também à outra parte para outras finalidades.

§ 5º Será responsabilizado por perdas e danos aquele que requerer e promover as medidas previstas neste e nos arts. 12 e 13, agindo de má-fé ou por espírito de emulação, capricho ou erro grosseiro, nos termos dos arts. 16, 17 e 18 do Código de Processo Civil.

CAPÍTULO VI

DISPOSIÇÕES FINAIS

Art. 15º Esta Lei entra em vigor na data de sua publicação.

Art. 16º Fica revogada a Lei nº 7.646, de 18 de dezembro de 1987.

Brasília, 16 de fevereiro de 1998; 177º da Independência e 110º da República

FERNANDO HENRIQUE CARDOSO

Anexo 2 – Softwares similares

Retirado de: <http://linuxshop.ru/linuxbegin/win-lin-soft-en/table.shtml>

Description of the program, executed task	Windows	Linux
1) Networking.		
Web browser	Internet Explorer, Netscape / Mozilla, Opera [Prop], Firebird, ...	1) Netscape / Mozilla . 2) Galeon . 3) Konqueror . 4) Opera . [Prop] 5) Firebird . 6) Nautilus . 7) Epiphany . 8) Links . (with "-g" key). 9) Dillo . (Russian language patches - here). 10) Encompass .
Console web browser	1) Links 2) Lynx 3) Xemacs + w3.	1) Links . 2) ELinks . 3) Lynx . 4) w3m . 5) Xemacs + w3.
Email client	Outlook Express, Netscape / Mozilla , The Bat, Eudora, Becky, Datula, Sylpheed / Sylpheed-claws, Opera	1) Evolution . 2) Netscape / Mozilla messenger. 3) Sylpheed / Sylpheed-claws . 4) Kmail . 5) Gnus . 6) Balsa . 7) Bynari Insight GroupWare Suite . [Prop] 8) Arrow . 9) Gnumail . 10) Althea . 11) Liamail . 12) Aethera . 13) MailWarrior . 14) Opera .
Email client / PIM in	Outlook	1) Evolution .

MS Outlook style		2) Bynari Insight GroupWare Suite . [Prop] 3) Aethera . 4) Sylpheed . 5) Sylpheed-claws .
Email client in The Bat style	The Bat	1) Sylpheed . 2) Sylpheed-claws . 3) Kmail . 4) Gnus . 5) Balsa .
Console email client	Mutt [de], Pine , Pegasus , Emacs	1) Pine . [NF] 2) Mutt . 3) Gnus . 4) Elm . 5) Emacs .
News reader	1) Agent [Prop] 2) Free Agent 3) Xnews 4) Outlook 5) Netscape / Mozilla 6) Opera [Prop] 7) Sylpheed / Sylpheed-claws 8) Dialog 9) Gravity	1) Knode . 2) Pan . 3) NewsReader . 4) Netscape / Mozilla . 5) Opera [Prop] 6) Sylpheed / Sylpheed-claws . Console: 7) Pine . [NF] 8) Mutt . 9) Gnus . 10) tin . 11) slrn . 12) Xemacs .
Address book	Outlook	1) Rubrica
Files downloading	Flashget , Golzilla , Reget , Getright , DAP , Wget , WackGet , Mass Downloader , ...	1) Downloader for X . 2) Caitoo (former Kget). 3) Prozilla . 4) Wget (console, standard). 5) GUI for Wget : Kmagoo , Gnome Transfer Manager , QTget , Xget , ... 6) Aria . 7) Axel . 8) Download Accelerator Plus . 9) GetLeft . 10) Lftp .
Sites downloading	Teleport Pro , Htrack , Wget , ...	1) Htrack . 2) WWW Offline Explorer . 3) Wget (console. standard). GUI: Kmagoo .

		QTget, Xget, ... 4) <u>Downloader for X</u> . 5) <u>Pavuk</u> . 6) <u>XSiteCopy</u> . 7) <u>GetLeft</u> . 8) <u>Curl</u> (console). 9) <u>Khttrack</u> .
FTP-clients	Bullet Proof FTP, CuteFTP, WSFTP, SmartFTP, ...	1) <u>Gftp</u> . 2) <u>Konqueror</u> . 3) <u>KBear</u> . 4) <u>IglooFTP</u> . [Prop] 5) <u>Nftp</u> . 6) <u>Wxftp</u> . 7) <u>AxyFTP</u> . 8) <u>mc</u> . (cd ftp://...) 9) <u>tkFTP</u> . 10) <u>Yafc</u> . 11) <u>Dpsftp</u> . (dead project)
Console FTP-clients	FTP in Far, ftp.exe, <u>Ncftp</u>	1) <u>Ncftp</u> . 2) <u>Lftp</u> . 3) <u>Avfs</u> . (From any program: /#ftp:...)
IRC-clients	Mirc, Klient, VIRC, Xircon, Pirch, <u>XChat</u>	1) <u>Xchat</u> . 2) <u>KVirc</u> . 3) <u>Irssi</u> . 4) <u>BitchX</u> . 5) <u>Ksirc</u> . 6) <u>Epic</u> . 7) <u>Sirc</u> . 8) <u>PJIRC</u> .
Local network chat clients without server	1) QuickChat 2) <u>Akeni</u> 3) <u>PonyChat</u> 4) iChat	1) talk (console), ktalk. 2) <u>Akeni</u> . 3) <u>Echat</u> . 4) write, wall (chat between users of one machine)
Local messaging with Windows machines	WinPopUp	smbclient (console). GUI: 1) <u>LinPopUp 2</u> . 2) <u>Kpopup</u> . 3) <u>Kopete</u> .
Instant Messaging clients	ICQ Lite, ICQ Corp, MSN, <u>AIM</u> , <u>Yahoo</u> , ... Trillian ICQ (freeware. can	1) <u>Licq</u> (ICQ). 2) <u>Centericq</u> (Nearly all IM protocols, console). 3) <u>Alicq</u> (ICQ). 4) <u>Mica</u> (ICQ).

	replace all common IM clients), <u>Miranda</u> , <u>Gaim</u>	5) <u>GnomeICU</u> (ICQ). 6) <u>Gaim</u> . (Nearly all IM protocols) 7) <u>Aytm</u> . (Nearly all IM protocols) 8) <u>Kopete</u> . 9) <u>Everybuddy</u> . 10) <u>Simple Instant Messenger</u> . 11) <u>Imici Messenger</u> . 12) <u>Ickle</u> (ICQ). 13) <u>aMSN</u> (MSN). 14) <u>Kmerlin</u> (MSN). 15) <u>Kicq</u> (ICQ). 16) <u>YSM</u> . (ICQ, console). 17) <u>kxicq</u> . 18) <u>Yahoo Messenger for Unix</u> . 19) <u>Kmess</u> (MSN). 20) <u>AIM</u> . 21) <u>MSNre</u> . (console)
Jabber IM clients	<u>JAJC</u> , <u>Tkabber</u> (+activestate tcl), <u>Psi</u> , <u>Exodus</u> , <u>WinJab</u> , <u>myJabber</u> , <u>RhymBox</u> , <u>Rival</u> , <u>Skabber</u> , <u>TipicIM</u> , <u>Vista</u> , <u>Yabber</u> , <u>Miranda</u> , <u>Gaim</u> , <u>Akeni Messenger</u> <u>Jabber Edition</u>	1) <u>Tkabber</u> . 2) <u>Gabber</u> . 3) <u>Psi</u> . 4) <u>Gaim</u> . 5) <u>Centericq</u> (console). 6) <u>Aytm</u> . 7) <u>Akeni Messenger Jabber Edition</u> .
Monitoring sites and mailboxes, displaying news headlines or full articles	<u>WatzNew</u>	1) <u>Web Secretary</u> . 2) <u>Knewsticker & korn</u> . 3) <u>Mozilla</u> (???). 4) <u>watch -n seconds lynx -dump</u>
Video/audio conference	<u>NetMeeting</u>	1) <u>GnomeMeeting</u> . 2) <u>vat/vic/wb</u> . 3) <u>rat/wbd/nte</u> . 4) <u>NeVoT</u> . 5) <u>IVS</u> .
Voice communication	<u>Speak Freely</u>	1) <u>Speak Freely for Unix</u> . 2) <u>TeamSpeak</u> .
Firewall (packet filtering)	<u>BlackICE</u> , <u>ATGuard</u> , <u>ZoneAlarm</u> , <u>Agnitum</u> <u>Outpost Firewall</u> , <u>WinRoute Pro</u> , <u>Norton Internet Security</u> , <u>Svagate</u>	iptables or more outdated ipchains (console, standard). Front ends: 1) <u>Kmyfirewall</u> . 2) <u>Easy Firewall Generator</u> . 3) <u>Firewall Builder</u> . 4) <u>Shorewall</u> .

	<u>Personal Firewall</u> <u>PRO</u> , Kerio Personal Firewall, ...	5) <u>Guarddog</u> . 6) <u>FireStarter</u> . 7) <u>Smoothwall</u> . [Prop] 8) <u>IPCop</u> . 9) <u>Zorp</u> .
IDS (Intrusion Detection System)	1) BlackICE 2) Agnitum Outpost Firewall 3) Tripwire [prop] 4) Kerio Personal Firewall	1) <u>Snort</u> . 2) <u>Portsentry</u> / <u>Hostsentry</u> / <u>Logsentry</u> . 3) <u>Tripwire</u> [GPL]. 4) <u>Tripwall</u> . 5) <u>AIDE</u> . 6) <u>ViperDB</u> . 7) <u>Integrit</u> . 8) <u>Cerberus Intrusion Detection System</u> . 9) <u>MIDAS NMS</u> .
Port scanning detection	???	1) <u>Pkdump</u> .
Making the system more securely	???	1) <u>Bastille</u> . 2) <u>Linux Security Auditing Tool</u> .
Visual route	VisualRoute [Prop]	1) <u>Xtraceroute</u> . 2) <u>VisualRoute</u> . [Prop] 3) <u>Mtr</u> . 4) <u>Geotrace</u> .
Content (ad / popup) filtering	Proxomitron, ATGuard, Agnitum Outpost Firewall, <u>Privoxy</u> , MS ISA server, <u>Guidescope</u> , ...	1) <u>DansGuardian</u> . 2) <u>Squid</u> . 3) <u>Squidguard</u> . 4) <u>Privoxy</u> . 5) <u>JunkBuster</u> . 6) <u>Zorp</u> . 7) <u>Fork</u> . 8) <u>Redirector</u> .
Traffic control / shaping	<u>WinRoute Pro</u> , ...	1) <u>IP Relay</u> . 2) <u>CBQ</u> (from iproute2 package). 3) <u>tc</u> (from iproute2 package). 4) <u>LARTC</u> .
Traffic accounting	Netstat, <u>Tmeter</u> , ...	1) <u>Tcp4me</u> . 2) <u>Getstatd</u> . 3) <u>lpacct</u> . 4) <u>lpac-ng</u> . 5) <u>lpaudit</u> . 6) <u>Lanbilling</u> . 7) <u>SARG</u> (full Squid traffic). 8) <u>Talinux</u> . 9) <u>NetUP UserTrafManager</u> .

		10) <u>MRTG</u> . 11) <u>NetTop</u> .
Peer-to-peer clients / servers, file sharing (p2p)	Morpheus (Gnutella), WinMX, Napster, KaZaA (Fasttrack), <u>eDonkey</u> [Prop], <u>eMule</u> , <u>TheCircle</u> , <u>Bittorrent</u> , <u>SoulSeek</u> , <u>Direct Connect</u>	1) <u>Mldonkey</u> . (eDonkey, Souleseek, Fasttrack, Gnutella, Open Napster, Direct Connect, BitTorrent) 2) <u>LimeWire</u> . (Gnutella) 3) <u>Lopster</u> . (OpenNAP) 4) <u>Gnapster</u> . (OpenNAP) 5) <u>eDonkey</u> . (eDonkey) [Prop] 6) <u>cDonkey</u> . (eDonkey) 7) <u>Gift client</u> / <u>server</u> / <u>fasttrack plugin</u> (Fasttrack) 8) <u>ed2k_gui</u> . 9) <u>Gtk-Gnutella</u> . (Gnutella) 10) <u>Qtella</u> . (Gnutella) 11) <u>Mutella</u> . (Gnutella, console) 12) <u>TheCircle</u> . 13) <u>Freenet</u> . (fully anonymous p2p) 14) <u>GNUnet</u> . 15) <u>Lmule</u> . (eDonkey) 16) <u>Xmule</u> . (eDonkey) 17) <u>Bittorrent</u> . 18) <u>PySoulSeek</u> (Souleseek). 19) <u>Loophole</u> . (WinMX) [Prop] 20) <u>Direct Connect</u> . 21) <u>QuickDC</u> . (Direct Connect). 22) <u>OverNet</u> . 23) <u>Apollon</u> . 24) <u>GrapeWine</u> . (fully anonymous p2p) 25) <u>Snark</u> . (Bittorrent)
"Hotline" p2p protocol clients / servers	???	Clients: 1) <u>Gtkhx</u> . 2) <u>Fidelio</u> . 3) <u>GHX</u> . [Prop] Servers: 1) <u>Synapse</u> . 2) <u>HSX</u> .
Program for working with sound modem with many functions - from answerback and AON to fax device	Venta Fax, <u>PrimaFax</u>	1) <u>Gfax</u> . (???) 2) <u>PrimaFax</u> . [Prop, 99\$] 3) <u>mgetty</u> + voice + <u>sendfax</u> . 4) <u>vgetty</u> .
Work with faxes	WinFax	1) <u>HylaFax</u> . 2) <u>Fax2Send</u> . [Prop]

		3) <u>Efax</u> . 4) <u>VSI-FAX</u> . [Prop]
Dialup	Vdialer, etc	1) <u>Kppp</u> . 2) <u>X-isp</u> . 3) <u>wvdial</u> . (Front ends: <u>X-wvdial</u> , <u>kvdial</u> , <u>gtkdialog</u>). 4) <u>Gppp</u> . 5) <u>Kinternet</u> . 6) <u>Rp3</u> . 7) <u>pppconfig</u> + <u>pon</u> + <u>poff</u> . 8) <u>Modem Lights</u> . 9) <u>Netcount</u> . (console)
FTN editor	Golded	1) <u>Golded</u> . 2) <u>Golded+</u> . 3) <u>Msged</u> . 4) <u>Qded</u> .
FTN tosser	FastEcho, hpt	1) <u>hpt</u> . 2) <u>CrashEcho</u> . 3) <u>Qecho</u> . 4) <u>CrashMail II</u> . 5) <u>Fidogate</u> - gate to news. 6) <u>ifmail</u> - gate to news.
FTN mailer	SF-Mail, T-Mail	1) <u>ifcico</u> . 2) <u>gico</u> . 3) <u>Bforce</u> . 4) <u>Binkd</u> .
Remote management	1) <u>VNC, Virtual Network Computing</u> 2) <u>Remote Administrator (Radmin)</u> [Prop] 3) <u>Remote Assistance</u> [Prop] 4) <u>Symantec pcAnywhere</u> [Prop] 5) <u>Windows Terminal Server</u> [Prop] 6) <u>Rdesktop</u> [Prop] 7) <u>Radmin</u> [Prop] 8) <u>PC-Duo</u> [Prop] 9) <u>Huey PC Remote Control</u> (only for NT) [Prop]	1) <u>VNC, Virtual Network Computing</u> . 2) <u>ssh</u> . 3) Remote management is built-in in <u>XFree86</u> . 4) Remote management is built-in in <u>KDE 3.1</u> . ("desktop sharing"). 5) <u>Rdesktop Client</u> . 6) <u>rsh</u> / <u>rlogin</u> . 7) <u>telnet</u> . 8) <u>Gtelnet</u> . 9) <u>x0rfbserver</u> . 10) <u>KDE Universal Remote Desktop</u> .

	10) <u>Timbuktu Pro</u> [Prop] 11) <u>LapLink</u> [Prop] 12) <u>GoToMyPC</u> [Prop] 13) <u>Bo2k</u> and other trojans	
Transmission of the files on modem	HyperTerminal, Terminate, etc	1) <u>Minicom</u> + <u>lrzsz</u> + <u>Kermit</u> . 2) <u>Mstern</u> . 3) <u>Xtel</u> . 4) <u>uucp</u> . 5) <u>lrzsz</u> + <u>cu</u> from <u>uucp</u> .
Work with ssh	Putty, Irlex, cygwin + <u>ssh</u>	1) <u>Kssh</u> . 2) <u>ssh</u> / <u>openssh</u> . 3) <u>GTelnet</u> . (Telnet, SSH, Rlogin)
Network monitoring tool	Dumeter, Netmedic	1) <u>Gkrellm</u> . 2) <u>Big Brother</u> . 3) <u>Etherape</u> . 4) <u>Nagios</u> . 5) <u>Tkined</u> . 6) <u>MRTG</u> . 7) <u>Rrdtool</u> . 8) <u>PIKT</u> . 9) <u>Autostatus</u> . 10) <u>bcnu</u> . 11) <u>mon</u> . 12) <u>Sysmon</u> . 13) <u>Spong</u> . 14) <u>SNIPS</u> . 15) <u>iptraf</u> (console). 16) <u>Ksysguard</u> . 17) <u>OpenNMS</u> . 18) <u>tcpdump</u> .
Network maintenance tool	HP OpenView, MS SMS, Tivoli	1) HP OpenView agents. [Prop] 2) <u>Big Brother</u> . 3) <u>Cheops</u> . 4) <u>Tkined</u> . 5) <u>OpenNMS</u> .
Protocols analysing, sniffing	<u>Sniffer Pro</u> , <u>EtherPeek</u> (TokenPeek, AiroPeek), <u>Windump</u> , <u>Ethereal</u> , MS Network Monitor, Iris, <u>Lan</u> Explorer, NetSniffer.	1) <u>Ethereal</u> . 2) <u>Tcpdump</u> . 3) <u>Etherape</u> . 4) <u>Ntop</u> . 5) <u>ipxdump</u> . (+ipxparse for parsing) 6) <u>Snort</u> .

	Snort, ...	
Security scanner	ShadowScan, ISS, Retina, Internet Scanner	1) <u>Nessus</u> . 2) <u>Nmap</u> .
Routing	MS RRAS	1) <u>iproute2 + iptables</u> . 2) <u>GateD</u> . [Prop] 3) <u>GNU Zebra</u> .
Utilites and libraries for Ethernet/IP testing	<u>lcrzo</u>	<u>lcrzo</u>
IP-telephony	Buddyphone [Prop], Cisco SoftPhone	1) <u>GNU Bayonne</u> . 2) <u>Openh323</u> . 3) <u>OpenPhone</u> . 4) <u>tkPhone</u> . 5) <u>NauPhone</u> .
Sharing data/files	Windows shares	1) <u>NFS</u> . 2) <u>Samba</u> . 3) <u>Samba-TNG</u> . 4) <u>FTP</u> .
???	Windows Domain, Active Directory	1) <u>Samba</u> . 2) <u>Ldap</u> . 3) <u>yp</u> .
Viewing Windows-network	Network neighborhood	1) <u>Samba</u> . 2) <u>KDE Lan Browser</u> , <u>lisa</u> 3) <u>LinNeighborhood</u> . 4) <u>xSMBrowser</u> . 5) <u>Komba2</u> . 6) <u>Konqueror</u> .
ADSL	1) <u>RASPPPOE</u>	1) <u>rp-pppoe</u> . 2) <u>Pptp client</u> .
Distributed computing	<u>All projects</u> . 1) <u>United Devices</u> . 2) <u>Seti @ Home</u> . 3) <u>Folding @ Home</u> . 4) <u>Genome @ Home</u> .	<u>All projects</u> . 1) <u>Distributed.net</u> . 2) <u>Seti @ Home</u> . 3) <u>Folding @ Home</u> . 4) <u>Genome @ Home</u> . 5) <u>D2ol</u> . * Some versions for Linux run only in console mode.
IPSEC protocol	-	<u>FreeSWAN</u> .
VRML viewer and editor	???	1) <u>White Dune</u> .

Work with Ebay	???	1) <u>Bidwatcher</u> .
2) Work with files.		
File manager in FAR and NC style	FAR, Norton Commander, Disco Commander, Volcov Commander, etc	1) <u>Midnight Commander</u> . 2) <u>X Northern Captain</u> . 3) <u>Deco (Demos Commander)</u> . 4) <u>Portos Commander</u> . 5) <u>Konqueror in MC style</u> . 6) <u>Gentoo</u> . 7) <u>VFU</u> . 8) <u>Ytree</u> .
File manager in Windows Commander style	Total Commander (former Windows Commander)	1) <u>Krusader</u> . 2) <u>Kcommander</u> . 3) <u>FileRunner (TCL/TK)</u> . 4) <u>Linux Commander</u> . 5) <u>LinCommander</u> . 6) <u>Rox, Rox-Filer</u> . 7) <u>Emelfm</u> . 8) <u>Midnight Commander</u> . 9) <u>Worker</u> .
File manager in Windows style	Windows Explorer / Internet Explorer	1) <u>Konqueror</u> . 2) <u>Gnome-Commander</u> . 3) <u>Nautilus</u> . 4) <u>Endeavour Mark II</u> . 5) <u>XWC</u> .
Visual Shell	???	1) <u>vshnu: the New Visual Shell</u> .
Quick viewing of local HTML documents	1) Internet Explorer 2) Microsoft Document Explorer	1) <u>Dillo</u> . (Russian language patches - here). 2) <u>Konqueror</u> . 3) <u>Nautilus</u> . 4) <u>Lynx / Links</u> .
Viewing all operations with files	Filemon	<u>Filemon</u> .
Work with UDF	Roxio (former Adaptec) UDF Reader, Roxio Direct CD	<u>Linux-UDF</u> .
Work with multi session CD + recovering information from faulty multi session disks	<u>IsoBuster</u>	<u>CDFS</u> .

Work with compressed files	1) WinZip 2) WinRar 3) 7-Zip 4) WinACE 5) UltimateZip	1) Ark (kdeutils). 2) Gnozip. 3) KArchiveur. 4) Gnohive. 5) FileRoller. 6) Unace. 7) LinZip. 8) TkZip.
Console archivers	arj, rar, zip, tar, gzip, bzip2, lha...	1) tar, gzip, bzip2. 2) Unarc. 3) Lha. (on Japanese) 4) Infozip. 5) Arj. 6) Avfs. (The support of any archive from any program - file.zip#/...). 7) Zoo. 8) RAR. 9) CAB Extract.
Program for files and directories comparison	1) Beyond Compare. 2) Araxis Merge. 3) Minimalist GNU For Windows (diff.exe) 4) Unison	1) WinMerge. 2) Mgdiff. 3) diff, patch. 4) Xemacs. 5) Xdelta. (make patches for binary files) 6) Meld. 7) Xxdiff. 8) Unison.

3) Desktop / System software.

Text editor	Notepad, WordPad, TextPad, <u>Vim</u> , <u>Xemacs</u> , ...	1) Kedit (KDE). 2) Gedit (Gnome). 3) Gnotepad. 4) Kate (KDE). 5) KWrite (KDE). 6) Nedit. 7) <u>Vim</u> . 8) Xemacs. 9) Xcoral. 10) <u>Nvi</u> . 11) <u>Ozeditor</u> .
Console text editor	1) <u>Vim</u> 2) <u>Emacs</u> 3) Editor from FAR, editor from DN, ...	1) <u>Vim</u> . 2) <u>Emacs</u> . 3) <u>Nano</u> . (It's a free implementation of <u>pico</u>) 4) <u>joe</u> . 5) <u>Fte</u> . 6) <u>Jed</u> .

		7) <u>CoolEdit</u> .
Multi-purpose text and source code editor	<u>SciTE</u> , <u>UltraEdit</u> , <u>MultiEdit</u> , <u>Vim</u> , <u>Xemacs</u> , ...	1) <u>Kate</u> (KDE). 2) <u>Nedit</u> . 3) <u>CodeCommander</u> . 4) <u>SciTE</u> . 5) <u>Quanta Plus</u> . 6) <u>Vim</u> . 7) <u>Xemacs</u> . 8) <u>Mcedit</u> (comes with <u>mc</u>). 9) <u>Jed</u> . 10) <u>Setedit</u> . (Turbo Vision looking style) 11) <u>HT editor</u> .
Text editor with cyrillic encodings support	<u>Bred</u> , <u>Rpad32</u> , <u>Aditor</u> , <u>Vim</u> , <u>Xemacs</u>	1) <u>Kate</u> (KDE). 2) <u>Nedit</u> . 3) <u>Fte</u> (console). 4) Patched version of Midnight Commander. 5) <u>Vim</u> . 6) <u>Xemacs</u> .
Viewing PostScript	1) <u>RoPS</u> 2) <u>GhostView</u>	1) <u>GhostView</u> . 2) <u>Kghostview</u> . 3) <u>GV</u> . 4) <u>GGV</u> .
Viewing PDF	1) Adobe Acrobat Distiller 2) <u>GhostView</u>	1) <u>Acrobat Reader</u> . [Prop] 2) <u>Xpdf</u> . 3) <u>GV</u> . 4) <u>GGV</u> . 5) <u>GhostView</u> . 6) <u>Kghostview</u> .
Creating PDF	1) Adobe Acrobat Distiller 2) <u>GhostView</u> 3) <u>Ghostsript</u>	1) Any Linux WYSIWYG program -> print to file -> ps2pdf. (Here's an article about this). 2) <u>Adobe Acrobat Distiller</u> . [Prop] 3) <u>PStill</u> . [Shareware] 4) <u>PDFLatex</u> . 5) <u>Xfig</u> . 6) <u>Ghostsript</u> . 7) <u>Tex2Pdf</u> . 8) <u>Reportlab</u> . 9) <u>GV</u> . 10) <u>GGV</u> . 11) <u>GhostView</u> . 12) <u>Kghostview</u> . 13) <u>Panda PDF Generator</u> .

Crypto	PGP , GnuPG + Windows Privacy Tools	1) GnuPG (console) + GPA , KGpg , and other frontends. 2) PGP . [Prop]
Disc volume encryption	EFS (standard), PGP-Disk , BestCrypt , Private Disk Light	1) Loop-aes . 2) CFS . 3) TCFS . 4) BestCrypt . 5) CryptFS .
Task scheduler	mstask, nnCron	cron, at (standard, console). GUI: Kcron.
Outlook scheduler	Outlook scheduler	1) KOrganizer.
Virtual CD	VirtualDrive , VirtualCD , Daemon Tools , ...	1) Virtual CD Kernel Modul . 2) "cp /dev/cdrom mycd.iso" + "mount -o loop mycd.iso /mnt/cdrom/".
Text recognition (OCR)	Recognita , FineReader	1) ClaraOcr . 2) Gocr . 3) Kooka .
Translators (cyrillic)	Promt , Socrat	Ksocrat (???)
Eng-rus dictionaries (cyrillic)	1) ABBY Lingvo 2) Socrat 3) JaLingvo 4) phpMyLingvo	1) Mueller . 2) Ksocrat . 3) JaLingvo . 4) phpMyLingvo . 5) dict+Kdict . 6) DictX . 7) Groan . 8) Mova . 9) Slowo . 10) Stardict .
Work with scanner	Programs on CD with scanner, VueScan	1) Xsane . 2) Kooka . 3) Xvscan . [Prop] 4) VueScan . [Prop]
Antivirus	AVG AntiVirus , NAV , Dr. Web , TrendMicro , F-Prot , Kaspersky , ...	1) Dr. Web . [Prop] 2) Trend ServerProtect . [Prop] 3) RAV Antivirus . [Prop] (Bought by Microsoft?) 4) OpenAntivirus + AMaViS / VirusHammer . 5) F-Prot . [Prop] 6) Sophie / Trophie . 7) Clam Antivirus . 8) Kaspersky . [Prop] 9) YAVR .

System configuration	Control Panel, Msconfig, RegEdit, WinBoost, TweakXP, Customizer XP, X-Setup, PowerToys, Config NT, ...	1) setup (Red Hat). 2) chkconfig (Red Hat). 3) redhat-config-<feature>. (Red Hat 8.0). 4) xf86config, xf86cfg. 5) <u>Linuxconf</u>. 6) Drakeconf. 7) Webmin. 8) yast and yast2 (SuSE). 9) sysinstall (FreeBSD). 10) /dev/hands :).
Boot managers	System Commander, PowerQuest Boot Magic, <u>GAG</u> , ...	1) <u>Grub</u>. 2) Lilo. 3) ASPLoader. 4) <u>Acronis OS Selector</u> [Prop]. 5) <u>Ranish Partition Manager</u>. 6) <u>osbs</u>. 7) <u>Symon</u>. 8) <u>Smart Boot Manager</u>. 9) <u>Xosl</u>. 10) <u>GAG</u>.
Hard disk partitions manager	1) <u>PowerQuest Partition Magic</u> [Prop] 2) <u>Acronis PartitionExpert</u> [Prop] 3) <u>Paragon Partition Manager</u> [Prop] 4) <u>Partition Commander</u> [Prop]	1) PartGUI. (GUI for partimage and parted) 2) <u>GNU Parted</u>. (GUI - QTParted). 3) <u>Partition Image</u>. 4) fips. 5) Diskdrake (Mandrake). 6) <u>Paragon Partition Manager</u> [Prop]. 7) <u>Acronis PartitionExpert</u> [Prop]. (review)
LVM + soft-RAID + parted + ...	-	<u>EVMS</u> .
Backup software	ntbackup (standard), Legato Networker [Prop]	1) <u>Legato Networker</u>. [Prop] 2) <u>Lonetar</u>. [Prop] 3) <u>Disk Archive</u>. 4) <u>Bacula</u>. 5) <u>Taper</u>. 6) dump / restore. (console, standard) 7) <u>Amanda</u> (console). 8) <u>Mondo Rescue</u>. (backups that boot and auto restore themselves to disk)
Making images of disk partitions	1) Drivelmage 2) Ghostpe	1) <u>PartitionImage</u> (GUI - PartGUI). 2) dd (console, standard). 3) <u>Mondo Rescue</u>.

Machine mirroring over network	1) <u>ImageCast</u> 2) <u>Norton Ghost</u>	1) <u>UDP Cast</u> . 2) <u>Techteam's UDP Cast Disks</u> . 3) <u>Ghost for Unix (g4u)</u> .
Task manager	TaskMan (standard), TaskInfo, ProcessExplorer NT.	1) top (console, standard). 2) Gtop, Ktop. 3) Ksysguard. 4) "ps aux more", "kill <pid>"
Automatic switch between English and Russian when you type	<u>Punto Switcher</u> , <u>Keyboard Ninja</u> , <u>Snoop</u>	No analogs :(.
Mouse gestures	Sensiva	1) Kgesture. 2) wayV. 3) <u>Optimoz</u> .
TV program	???	<u>Home Portal</u> .
Text to speech	MS text to speech	1) <u>KDE Voice Plugins</u> . 2) <u>Festival</u> . 3) <u>Emacspeak</u> . 4) <u>VoiceText</u> .
Speech recognition	ViaVoice, DragonNaturally Speaking	Yes, there are no usable speech recognition packages. But: 1) <u>Sphinx</u> . 2) ViaVoice. (almost dead...)
Stream text processing	1) <u>Minimalist GNU For Windows (sed.exe)</u> 2) <u>perl</u>	1) sed, awk. 2) <u>perl</u> .
PIM / DB / hierarchical notebook with tree view	<u>TreePad [Prop]</u> , <u>Leo</u> , CueCards	1) <u>TreePad Lite</u> . [Prop] 2) <u>Yank</u> . 3) <u>TreeLine</u> . 4) <u>Gjots</u> . 5) <u>Leo</u> . 6) <u>Hnb - hierarchical notebook</u> . 7) <u>TuxCards</u> .
Program for quick switching between resolutions and frequencies	Integrated with system, comes on CD with video card	1) <u>Multires</u> . 2) Ctrl+Alt+"-", Ctrl+Alt+"+". 3) <u>Fbset</u> . (when using framebuffer)
Search and replace text in files	1) Integrated with svstem	1) find (console, standard). 2) <u>slocate</u> (console. standard).

	2) Indexing service 3) HTMLChanger 4) Any file manager	GUI: 1) Gsearchtool. 2) Kfind. 3) Any file manager
Local search engine with SGBD for indexing	1) Indexing service (???) 2) mnoGoSearch . [prop for Windows]	1) mnoGoSearch . 2) ASPSeek .
System monitoring	System monitor (built-in)	1) top (console, standard). 2) Gkrellm . 3) Ksysguard. 4) Survivor . 5) "Hot-babe". (If you can be shocked by nudity, don't use it :).
Program for logs viewing	Event Viewer (built-in)	1) Xlogmaster . 2) Analog . 3) Fwlogview . (firewall)
Data Recovery Tools	R-Studio (supports Linux partitions)	1) e2undel . 2) myrescue . 3) TestDisk . 4) unrm . 5) Channel 16 .
Text files encoder with automatic detection of character set	???	1) Enca . 2) Jconv . 3) Xcode . (cyrillic) 4) Asreco . (cyrillic)
Alarm clock	MindIt!	1) KAlarm .
Work with Palm	Palm Desktop	1) Kpilot . 2) Jpilot .
Allows to carry on Palm .html files	iSilo	1) iSilo . 2) Plucker .
Low-level optimization (chipset, pci-bus)	Powertweak	1) Powertweak-Linux .
Bible	On-Line Bible, The SWORD	1) BibleTime (KDE) 2) Gnomesword (Gnome)
Convenient mouse scrolling	Mouse Imp	???
Automatic change of desktop background	???	1) Background Buddy .
Convenient switching	???	1) GSwitchIt .

of the keyboard language		
Windows registry editor	RegEdit	Kregedit :).
4.1) Multimedia (audio / CD).		
Useful links:	-	Linux MIDI & Sound Applications - many links and resources. Sound Software - lots of it.
Music / mp3 / ogg players	1) Winamp 2) Zinf 3) SnackAmp 4) Soritong 5) Apollo 6) K-jofol 2000 7) Sonique 8) C-4 9) Media Box Audio / Video Workstation 5 10) Blaze Media Pro 11) NEX 3 12) Real Jukebox 13) Windows Media Player	1) XMMS (X multimedia system). 2) Noatun . 3) Zinf . (former Freeamp) 4) Winamp . 5) Xamp . 6) GQmpeg . 7) SnackAmp . 8) Mplayer . (Frontend: Kplayer). 9) Xine . (Frontends: Sinek , Totem)
Console music / mp3 / ogg players	mpg123 , dosamp , Mplayer	1) Cplayer . 2) mpg123 . 3) ogg123 . 4) mpg321 . 5) Orpheus . 6) Mp3blaster . 7) Madplay . 8) Console utils for xmms . 9) Mplayer .
Programs for CD burning with GUI	Nero, Roxio Easy CD Creator, ...	1) K3b . (KDE) 2) XCDRoast . 3) KOnCd . 4) Eclipt Roaster . 5) Gnome Toaster . 6) CD Bake Oven . 7) KcreateCD . 8) SimpleCDR-X . 9) GCombust . 10) WebCDWriter . (CD burn server, usable from any remote browser with

		Java support) 11) <u>CDR Toaster</u> . 12) <u>Arson</u> . 13) <u>CD-Me</u> (Creation of audio-CD).
CD player	CD player, Winamp, Windows Media Player, ...	1) <u>KsCD</u> . 2) <u>Gtcd</u> (Gnome) + <u>tcd</u> (console). 3) <u>Orpheus</u> . (console) 4) <u>Sadp</u> . 5) <u>WorkMan</u> . 6) <u>Xmcd</u> . 7) <u>Grip</u> . 8) <u>XPlayCD</u> . 9) <u>ccd</u> / <u>cccd</u> . (console) 10) <u>cdp</u> . (console) 11) <u>BeboCD</u> .
Slowing the rotation of CD	Slowcd, Cdslow	1) <u>mount -o speed=<speed></u> 2) <u>hdparm -E <speed></u> 3) <u>eject -x <speed></u> 4) <u>cdspeed</u> .
CD ripping / grabbing	1) <u>Cdex</u> 2) <u>MusicMatch</u> 3) <u>Streambox Ripper</u> 4) <u>Audiocatalyst</u> 5) <u>WinDac</u> 6) <u>Audiograbber</u> 7) <u>Media Box Audio / Video Workstation</u> 8) <u>CD-Copy</u> 9) <u>Blaze Media Pro</u> 10) <u>Real Jukebox</u> 11) <u>Windows Media Player</u> 12) <u>Nero</u> 13) <u>VirtualDrive</u> 14) <u>VirtualCD</u> 15) <u>Audacity</u>	1) <u>Grip</u> . 2) <u>Audacity</u> . 3) <u>RipperX</u> . 4) <u>tkcOggRipper</u> . 5) <u>A Better CD Encoder</u> . 6) <u>cdda2wav</u> . 7) <u>Gnome Toaster</u> . 8) <u>Cdparanoia</u> . 9) <u>Cd2mp3</u> . 10) <u>Dagrab</u> . 11) <u>SimpleCDR-X</u> . 12) <u>RatRip</u> . 13) <u>AutoRip</u> . 14) <u>Sound Juicer</u> .
Tracker music player	Winamp, Windows Media Player, ...	1) <u>xmms + MikMod-plugin</u> . Comes with <u>xmms</u> . 2) <u>xmms + xmp-plugin</u> . Using <u>xmp</u> . 3) <u>MikMod</u> . (console) 4) <u>xmp</u> . Can play tracker music with Midi devices. 5) <u>TiMidity++</u> . Only mod files.

Midi player	Winamp, Windows Media Player, ...	1) xmms + midi-plugin . (Using TiMidity) 2) xmms + awemidi-plugin . (Using drvmidi) 3) xmms + playmidi-plugin . (Using playmidi) 4) TiMidity++ . Supports gus pathes and sf2 banks, backend to another software. 4) timidity-eawpatches . Gus-patches for TiMidity++. 5) Kmid . 6) drvmidi / awemidi . 7) pmidi (console, for ALSA). 8) playmidi . 9) atmidi .
Midi + karaoke player	VanBasco	1) Kmid . 2) Gkaraoke . 3) TiMidity++ .
Mp3 encoders	1) Lame 2) Cdex 3) MusicMatch 4) Streambox Ripper 5) Audiocatalyst 6) Blaze Media Pro 7) Media Box Audio / Video Workstation 8) AudioSlimmer 9) Real Jukebox	1) Lame . 2) Bladeenc . 3) NotLame . 4) L3enc . [Prop] 5) gogo .
OGG encoders	oggenc	oggenc .
Work with Real protocol	1) RealPlayer . [Prop] 2) Mplayer + libraries.	1) RealPlayer . [Prop] 2) Mplayer + libraries. 3) ReMedial .
Radio	VC Radio, FMRadio, Digband Radio	1) xradio . 2) cRadio . 3) Xmradio . 4) RDJ . 5) RadioActive . 6) XMMS-FMRadio . 7) Gqradio . 8) Qtradio .
Audio editors	SoundForge, Cooledit, Audacity , ...	1) Glame . 2) Rezound . 3) Sween .

		4) WaveForge . 5) Sox . 6) Audacity . 7) GNUSound . 8) Ecasound . 9) SoundStudio . 10) mhWaveEdit .
Multitrack audio processor	Cubase	1) Ecasound . 2) Ardour .
Sound tracker	Fasttracker, ImpulseTracker	1) Soundtracker . 2) Insotracker . 3) CheeseTracker .
Sound mixer	sndvol32	1) Opmixer . 2) aumix . 3) mix2000 . 4) KMix . 5) Alsamixer . 6) GMix . 7) wmix (for WindowMaker) 8) Mixer_app (for WindowMaker) 9) Many applets for AfterStep / WindowMaker / FluxBox .
Software for music notation	Finale , Sibelius , SmartScore	1) LilyPond . 2) Notedit . 3) MuX2d .
Midi-sequencer	Cakewalk	1) RoseGarden . 2) Brahms . 3) Anthem . 4) Melys . 5) MuSE . 6) MidiMountain . (KDE) More information: Midi-Howto .
Music creation	Cakewalk, FruityLoops	1) RoseGarden . 2) Ardour .
Synthesizer	Virtual waves, Csound	1) Csound . 2) FluidSynth . 3) Arts Builder (???) .
ID3-Tag Editors	Mp3tag	1) EasyTAG . 2) Cantus . 3) id3tool (console). 4) id3ed (console). 5) id3edit (console).

Guitar/Instrument Tuning software	1) <u>In-Tune Multi-Instrument Tuner</u> 2) <u>Digital Guitar Tuner</u>	???
Record streaming audio into .MP3 files	<u>Twins Stream Ripper</u>	<u>Stream Ripper.</u>

4.2) Multimedia (graphics).

Graphic files viewer	1) ACDSee 2) IrfanView 3) <u>Xnview</u> 4) <u>CompuPic</u> [Prop] 5) Windows Fax and Image viewer 5) Any web browser	1) <u>Xnview.</u> 2) <u>GQview.</u> 3) <u>Qiv.</u> 4) <u>CompuPic</u> [Prop] 5) <u>Kuickshow.</u> 6) <u>Kview.</u> 7) <u>GTKSee.</u> 8) <u>xv.</u> [Prop] 9) <u>pornview.</u> 10) <u>imgv.</u> 11) <u>Gwenview.</u> 12) <u>Gliv.</u> 13) <u>Showimg.</u> 14) <u>Fbi.</u> 15) <u>Gthumb.</u> 16) <u>PixiePlus.</u> 17) <u>Electric Eyes</u> (Gnome). 18) <u>Eye of Gnome.</u> 19) <u>GImageView.</u> 20) <u>Hugues Image Viewer.</u> 21) Any web browser.
Viewing the graphic files in console	QPEG	1) <u>zgv.</u> 2) <u>aalib.</u>
Simple graphic editor	Paint	1) <u>Kpaint.</u> 2) <u>Tuxpaint.</u> 3) <u>Xpaint.</u> 4) <u>Gpaint.</u> 5) <u>Killustrator.</u> 6) <u>Graphtool.</u>
Powerful graphic editor in PhotoShop style	1) Adobe Photoshop [Prop] 2) <u>Gimp</u> 3) Paint Shop Pro [Prop] 4) <u>Pixel32</u> [Prop] 5) Corel PhotoPaint	1) <u>Gimp.</u> 2) <u>ImageMagick.</u> 3) <u>Pixel32.</u> [Prop] 4) <u>CinePaint.</u> 5) <u>RubyMagick.</u> 6) <u>Corel PhotoPaint 9.</u> [Prop]

	[Prop] 6) Macromedia Fireworks	
Programs for work with vector graphics	Adobe Illustrator, Corel Draw, Freehand, AutoSketch, <u>OpenOffice Draw</u>	1) <u>Sodipodi</u> . 2) <u>xfig</u> . 3) <u>Sketch</u> . 4) <u>Karbon14 and Kontour</u> . 5) <u>OpenOffice Draw</u> . 6) <u>Dia</u> . 7) <u>Tgif</u> . 8) <u>Gestalter</u> . 9) <u>ImPress</u> . 10) <u>Tkpaint</u> . 11) <u>Tgif</u> . 12) Corel Draw 9. [Prop]
SVG editor	WebDraw [Prop]	1) <u>Sodipodi</u> . 2) <u>Dia</u> . 3) <u>Sketch</u> .
SVG viewer	1) Squiggle 2) X-Smiles 3) <u>SVG support in Mozilla</u>	1) Squiggle. 2) X-Smiles. 3) <u>SVG support in Mozilla</u> . 4) <u>SVG support in Konqueror</u> .
Program for text decoration	Wordart, <u>OpenOffice Draw</u>	<u>OpenOffice Draw</u> .
Program for ASCII- drawing	ANSI DRAW, <u>Mazaika</u>	1) <u>CanvASCII</u> . 2) <u>Jave</u> . 3) ANSI Draw. 4) <u>EDASCII</u> .
Converting the graphic files in pseudographics	???	aalib.
Flash playing	Flash Player	1) <u>SWF Macromedia Flash Player</u> . 2) <u>Flash Player</u> . [Prop] 3) Plugin for Netscape/Mozilla (download here or here).
Flash creation	Macromedia Flash	1) <u>DrawSWF</u> . 2) <u>Ming</u> . (Creating flash swf output from programming languages)
3D-graphics	3D Studio MAX, <u>Maya</u> [Prop], <u>Povray</u> , ...	1) <u>Blender</u> . 2) <u>Maya</u> . [Prop] 3) <u>KPovModeler</u> . 4) K3Studio.

		5) Moonlight . 6) GIG3DGO . 7) Povray . 8) MegaPov . 9) K3D . 10) Wings 3D . 11) Softimage XSI . [Prop] 12) Kludge3d .
The instrument for making the photographic quality scene based of ArchiCAD 3D-models	ArtLantis Render	???
Icon editor	Microangelo	1) Gnome-iconedit . 2) Kiconedit .
Small program for making screenshots	Integrated with system (PrintScreen), Snagit, ...	1) Ksnapshot . 2) Xwpick . 3) Xwd , xgrabsc . 4) Motv (xawtv) 5) Streamer (video) 6) Integrated with window manager (various hotkeys)
Drawing structure diagrams of the database	Access	1) Dia . 2) Toolkit for Conceptual Modelling .
Create printed calendars, greeting cards, etc., with clip art collection	Broderbund Print Shop	???
Download pictures from digital camera	Polaroid Drivers	1) Camera Tool (gtkam) 2) Gphoto2 .
4.3) Multimedia (video and other)		
Video / mpeg4 players	1) BSplayer 2) Zoomplayer 3) Windows Media Player 4) VideoLAN 5) Winamp3 6) Mplayer 7) RealPlayer 8) Xing 9) Simplayer	1) Mplayer . (Frontend: Kplayer). (LiveCD distribution of Mplayer - MoviX). 2) Xine . (Frontends: Sinek , Totem) 3) VideoLAN . 4) Aviplay . 5) Winamp3 . 6) Noatun . 7) KDE Media Player . 8) XMovie . 9) Kaboodle .

		10) MpegTV . 11) Avifile . 12) Xmps . 13) Ogg Tarkin. (???) 14) Theora. (???)
Console video / mpeg4 players	QuickView	1) Mplayer . 2) QuickView .
DVD players	PowerDVD, WinDVD, MicroDVD, Windows Media Player, VideoLAN	1) Ogle . 2) Mplayer . 3) Xine . 4) Aviplay . 5) VideoLAN . 6) OMS .
DVD rippers / encoders	Gordian Knot, Mencoder (from Mplayer)	1) Drip . 2) Transcode . 3) Mencoder . (from Mplayer) 4) Ffmpeg . 5) DVD::Rip .
Simple video creation and editing	Windows Movie Maker	1) iMira Editing . [Prop] 2) MainActor . [Prop] 3) Broadcast 2000. 4) Avidemux .
Professional video creation and editing	Adobe Premiere, Media Studio Pro	1) iMira Editing . 2) Cinelerra . 3) MainActor . 4) Broadcast 2000. 5) Lives . 6) CinePaint .
Cutting video	Virtual Dub	1) Avidemux . 2) Kino .
Converting video	Virtual Dub, Mencoder (from Mplayer)	1) Transcode . 2) Mencoder . (from Mplayer) 3) Ffmpeg .
Work with TV-tuner / watch TV	AVerTV, PowerVCR 3.0, CinePlayer DVR, Mplayer , ...	1) Tvtime . 2) Kwintv . 3) Xawtv . 4) Zapping . 5) GnomeTV . 6) Mplayer . 7) Xawdecode .
Work with TV-tuner in console	-	1) Mplayer . 2) fbtv .

		3) <u>aatv</u> .
Work with QuickTime format	QuickTime Player	1) <u>QuickTime</u> . 2) <u>Mplayer</u> + Sorenson codec. 3) <u>OpenQuicktime</u> . 4) <u>Xanim</u> .
Creation of 2D and 3D effects	Adobe After Effects	1) <u>Shake</u> . [Proprietary, \$129.95] 2) Plugins for Gimp.
Animation	Animation Shop, ...	1) <u>CinePaint</u> . 2) Plugins for Gimp.
Landscape / terrain generation	Bryce	<u>Terraform</u> .
World construction	World Construction Set, Animatek World Builder	???
Framework for developing video applications	???	1) <u>Gstreamer</u> . 2) <u>Live</u> .
5) Office/business.		
Office suite	MS Office, <u>StarOffice</u> / <u>OpenOffice</u> , 602Software	1) <u>OpenOffice</u> . 2) <u>StarOffice</u> . [Prop] 3) <u>Koffice</u> . 4) <u>HancomOffice</u> . [Prop] 5) <u>Gnome Office</u> . 6) <u>Applixware Office</u> . [Prop] 7) <u>Siag Office</u> . 8) <u>TeX</u> , <u>LaTeX</u> , ...
Office suite	WordPerfect Office 2000	WordPerfect Office 2000 for Linux. (No longer available at Corel website. It was Windows version, running under <u>Wine</u> :).
Word processor	Word, <u>StarOffice</u> / <u>OpenOffice</u> Writer, 602Text, <u>Abiword</u>	1) <u>Abiword</u> . 2) <u>TextMaker</u> [Prop] 3) <u>WordPerfect</u> . 4) <u>Ted</u> . 5) <u>StarOffice</u> / <u>OpenOffice</u> Writer. 6) <u>Kword</u> . 7) <u>LyX</u> . 8) <u>Kile</u> (KDE Integrated LaTeX Environment).
Spreadsheets	Excel, <u>StarOffice</u> / <u>OpenOffice</u> Calc, 602Tab	1) <u>Gnumeric</u> . 2) <u>Abacus</u> . 3) <u>StarOffice</u> / <u>OpenOffice</u> Calc.

		4) <u>Kspread</u> .
Graphing / charting data	Excel, ...	1) <u>Kivio</u> . 2) <u>Dia</u> . 3) <u>KChart</u> . 4) <u>xfig</u> . 5) <u>Gnuplot</u> . 6) <u>GtkGraph</u> . 7) <u>GNU Plotutils</u> . 8) <u>Ploticus</u> .
Creating presentations	MS PowerPoint, <u>StarOffice Presentation</u> , <u>OpenOffice Impress</u>	1) <u>StarOffice Presentation</u> . 2) <u>OpenOffice Impress</u> . 3) <u>Kpresenter</u> . 4) <u>MagicPoint</u> . 5) Kuickshow & gimp :).
Local database	MS Access, <u>InterBase6</u> , <u>OpenOffice + MySQL</u> .	1) <u>KNoda</u> . 2) <u>Gnome DB Manager</u> . 3) <u>OpenOffice + MySQL</u> . 4) <u>InterBase7</u> . [Prop] 5) <u>InterBase6</u> . 6) <u>Berkley DB</u> . 7) <u>Rekall</u> . [Prop] 8) <u>StarOffice Adabase</u> .
Receiving the quotations, news, building graphs and analysing of the financial market.	Omega Research Trade Station 2000	1) <u>The Market Analysis System (MAS)</u>
Software for e-commerce and web business	1) <u>Weblogic</u> [Prop] 2) <u>IBM WebSphere Application Server</u> [Prop] 3) <u>iPlanet</u> 4) <u>osCommerce</u> 5) <u>JOnAS</u> 6) <u>COCOON</u>	1) <u>Weblogic</u> . [Prop] 2) <u>JBoss</u> . 3) <u>IBM WebSphere Application Server</u> . [Prop] 4) <u>osCommerce</u> . 5) <u>JOnAS</u> . 6) <u>COCOON</u> .
Personal finances manager	1) MS Money 2) Quicken 3) <u>Moneydance</u> [Prop]	1) <u>GNUcash</u> . 2) <u>GnoFin</u> . 3) <u>Kmymoney</u> . 4) <u>Grisbi</u> . 5) <u>Moneydance</u> . [Prop]
Project management	MS Project, Project Expert 7	1) <u>Mr Project</u> . 2) <u>Outreach</u> .

Financial accounting package (global)	???	1) Hansa Business Solutions . [Prop] 2) Quickbooks .
Financial accounting package (russian)	"1C: Accounting"	1) Hansa Business Solutions . [Prop] 2) IceB . 3) "Finances without problems" . 4) Ananas . 5) E/AS . 6) 1L: Project .
Financial accounting package (India & Asia)	???	Kalculate . [Prop]
Automation of the enterprise (russian)	"1C: Enterprise"	1) Keeper . [Prop] 2) Oblik . [Prop] 3) IceB . 4) Compiere .
ERP\CRM (english)	???	1) Dolibarr .
ERP\CRM (russian)	"BOSS-Corporation"	1) NauRP . 2) Compiere . 3) Dolibarr .
Corporate docflow system (russian).	1) "Boss-Referent" 2) Documentum 3) "Delo" 4) Lanit:LanDoc	1) NauDoc . 2) Documentum. 3) "Boss-Referent" (without the client part)
6) Games.		
Where to get	Anywhere you want :).	The Linux Game Tome (happypenguin.org) LinuxGames.com Kde Games Linux Game Publishing
The Linux Game List	-	http://www.icculus.org/lqfaq/gamelist.php
Games for Windows, that can be run under WineX	-	This is a search form , and this is a full list of games (very big).
-	Tetris	1) LTris . 2) XWelltris . 3) Emacs + "Meta-X tetris" . 4) Ksirtet .
-	Standard Windows games	1) Kdegames . 2) Gnome-games .
-	Mines	1) KMines . 2) Perlmines . 3) Dmines .

-	Civilization	<u>FreeCiv.</u>
-	Civilization: Call to Power	Civilization: Call to Power.
-	Sid Meyer Alpha Centauri	Sid Meyer Alpha Centauri.
-	Sim City 3000	Sim City 3000.
-	Command&Conquer	<u>FreeCNC.</u>
-	Warcraft 2, Starcraft (?)	<u>FreeCraft.</u>
-	(Win)Digger	1) <u>Digger.</u> 2) <u>XDigger.</u>
-	Arkanoid, Zball, ...	<u>Lbreakout2.</u>
Quake 1, 2, 3	1) <u>Quake 1, 2, 3.</u> 2) <u>QuakeForge.</u> 3) <u>DarkPlaces.</u>	1) <u>Quake 1, 2, 3.</u> 2) <u>QuakeForge.</u> 3) <u>DarkPlaces.</u>
-	CounterStrike	CounterStrike under <u>WineX.</u>
-	Urban Terror	Urban Terror.
DOOM	1) <u>iDoom / Doomsday.</u> 2) <u>Zdoom.</u> 3) <u>DOOM Legacy.</u> 4) <u>LxDoom.</u> 5) <u>PrBoom.</u> 6) <u>EDGE.</u> 7) <u>Vavoom.</u> 8) <u>Original Doom.</u>	1) <u>iDoom / Doomsday.</u> 2) <u>Zdoom.</u> 3) <u>DOOM Legacy.</u> 4) <u>LxDoom.</u> 5) <u>PrBoom.</u> 6) <u>EDGE.</u> 7) <u>Vavoom.</u> 8) <u>Original Linux Doom (X11/svgalib).</u>
Heretic	1) <u>DOOM Legacy.</u> 2) <u>iHeretic / Doomsday.</u> 3) <u>Vavoom.</u> 4) <u>Original Heretic.</u>	1) <u>DOOM Legacy.</u> 2) <u>Vavoom.</u> 3) <u>Heretic, GL Heretic.</u>
Hexen	1) <u>iHexen / Doomsday.</u> 2) <u>Vavoom.</u> 3) <u>Original Hexen.</u>	1) <u>Hexen/SDL.</u> 2) <u>Vavoom.</u>
-	Heretic 2	<u>Heretic 2.</u>
-	Return to Castle Wolfenstein	<u>Return to Castle Wolfenstein. [Prop]</u>

-	Descent	Descent.
-	Never Winter Nights	Never Winter Nights.
-	Unreal Tournament / Unreal Tournament 2003	Unreal Tournament / Unreal Tournament 2003.
-	Soldier Of Fortune	Soldier Of Fortune.
-	Tribes 2	Tribes 2.
-	Blood 1	<u>Qblood</u> .
-	Worms	Nil.
-	Lines	1) <u>GtkBalls</u> . 2) <u>gLines</u> .
-	MS Flight Simulator	<u>FlightGear</u> .
-	Lemmings	<u>Pingus</u> .
Racing	Need For Speed	1) Tux Racer :). 2) <u>KartlingRace</u> .
Chess	ChessMaster, ...	1) <u>Glchess</u> . 2) <u>Xboard</u> . 3) <u>Eboard</u> .

7) Programming and development.

IDE	1) Microsoft VisualStudio .net 2) <u>Emacs</u> , <u>XEmacs</u> 3) <u>Vim</u> + <u>ctags</u> + <u>scripts from vim.sf.net</u> 4) <u>Boa Constructor</u> 5) <u>PythonCard</u>	1) <u>CodeForge</u> . 2) <u>Kdevelop</u> + <u>Qt3 Designer</u> . 3) <u>Eclipse</u> . 4) <u>Glade</u> + <u>Motor</u> or + <u>Xwpe</u> or + any text editor. 5) <u>Emacs</u> , <u>XEmacs</u> . 6) <u>Vim</u> + <u>ctags</u> + <u>scripts from vim.sf.net</u> . 7) <u>Boa Constructor</u> . 8) <u>PythonCard</u> .
Visual C++ IDE	Borland C++ Builder, MS Visual C	1) <u>Anjuta</u> + <u>Glade</u> + <u>Devhelp</u> . 2) <u>KDE Studio Gold</u> . [Prop] 3) <u>Dev-C++</u> . 4) <u>Kylix</u> . [Prop] (Kylix Personal Edition is free). 5) <u>vtkBuilder</u> . 6) <u>foxBuilder</u> . 7) <u>wxDesigner</u> . 8) <u>Arriba</u> . [Prop] 9) <u>Code Crusader</u> . [Prop] 10) <u>CodeWarrior</u> . [Prop]

		11) <u>Gbuilder</u> . 12) <u>Source Navigator</u> . 13) <u>TimeStorm</u> . [Prop] 14) <u>Understand for C++</u> . [Prop] 15) <u>SlickEdit</u> . [Prop] 16) <u>Vide</u> .
C++ IDE	Borland Turbo C++ 3.0 for DOS, , <u>Minimalist GNU For Windows (mingw32-gcc.exe)</u>	1) <u>GCC</u> (+ <u>Motor</u> or + <u>Xwpe</u>). 2) <u>LinEdit</u> . 3) <u>Rhide</u> . 4) <u>Wxstudio</u> . 5) <u>Eclipse</u> .
Object Pascal IDE	Delphi	1) <u>Kylix</u> . [Prop] (Kylix Personal Edition is free). 2) <u>Lazarus</u> + <u>FPC</u> .
Pascal	Pascal, BP	1) <u>Freepascal</u> . 2) <u>GNU Pascal</u> . 3) <u>RShell</u> (in style of Borland Pascal 7.0)
Basic	Basic	1) <u>Hbasic</u> . 2) <u>X-basic</u> . 3) <u>Yabasic</u> . 4) <u>SmallBASIC</u> .
Prolog	<u>VisualProlog</u> , <u>Mercury</u> , <u>SICStus Prolog</u> [Prop]	1) <u>GNU Prolog</u> . 2) <u>Mercury</u> . 3) <u>SWI-Prolog</u> . 4) <u>SICStus Prolog</u> . [Prop] 5) <u>CIAO Prolog</u> .
Assembler	<u>TASM</u> , <u>MASM</u> , <u>NASM</u>	1) <u>NASM</u> . (Intel syntax) 2) <u>FLAT Assembler</u> . 3) <u>gas</u> . (AT&T syntax, part of binutils).
Disassembler, Reverse engineering	Softlce	The source code is open :) 1) <u>ldasm</u> .
Debugger	1) <u>WinDbg</u> 2) <u>Minimalist GNU For Windows</u> (<u>gdb.exe</u>)	<u>gdb</u> . Frontends: 1) <u>ddd</u> . 2) <u>xxgdb</u> , <u>mxgdb</u> . 3) <u>CGDB</u> . 4) <u>Vim scripts</u> . 5) <u>[X]Emacs C-mode</u> . 6) <u>KMD</u> . 7) <u>NANA</u> . (Library)
WYSIWYG html editor	1) <u>Macromedia Dreamweaver</u> 2) <u>MS Frontpage</u>	1) <u>Netscape / Mozilla Composer</u> . 2) <u>Openoffice HTML editor</u> . 3) <u>Amava</u> .

	3) Netscape / Mozilla Composer 4) Openoffice HTML editor	4) GINF (Ginf is not Frontpage) 5) IBM WebSphere Homepage Builder . [Prop] 6) JXHTML EDIT (Java) .
Powerful editor for site creating, contains set of samples and can be complemented with every sort and kind of plug-ins	Dreamweaver Ultradev	???
HTML / DHTML editor	HomeSite , Coffeecup	1) Quanta Plus . 2) Bluefish . 3) WebMaker . 4) Screem . 5) Toppage . 6) WebDesigner . 7) ScriptEditor . 8) August . 9) Coffeecup / Linux . 10) FCKeditor .
HTML / DHTML editor	Arachnofilia	Arachnofilia . [Prop]
XML Editor	XML Spy [Prop]	1) XMLMind XML Editor . 2) Vim . 3) Emacs .
Perl/Python/Tcl IDE	-	1) Komodo . [Prop] 2) Perl Dev Kit .
Java IDE	JBuilder , IDEA	Link: Java Tools for Linux . 1) Jbuilder . 2) NetBeans . 3) Eclipse . 4) Sun ONE Studio . [formerly Forte] 5) Vide .
IDE for Oracle Database development	T.O.A.D. , SQL Navigator , PL/SQL Developer	Tora .
CASE-facility for UML	ArgoUML , Together ControlCenter [Prop]	1) Umbrello UML Modeller . 2) Dia+Dia2Code . 3) PoceidonCE (community edition) . 4) ArgoUML . 5) Together ControlCenter [Prop]
Top-level CASE system	Rational Rose .	Rational Rose . [Prop]

HEX-editor	<u>Hiew</u>	1) <u>Biew</u> . 2) <u>KHexEdit</u> . 3) <u>hexedit</u> (console). 4) <u>GHex</u> .
Clipper compiler and preprocessor	CA-Clipper, <u>The Harbour Project</u>	1) <u>Clip</u> . 2) <u>The Harbour Project</u> . 3) <u>xHarbour</u> .
Platform in dot-net style	M\$.Net	1) <u>Mono</u> . 2) <u>DotGNU/Portable.NET</u>
Work with CVS	WinCVS, TortoiseCVS, cvs for Windows, <u>BitKeeper</u> [Prop]	1) <u>cvs</u> (console). 2) <u>Cervisia</u> (KDE). 3) <u>Lincvs</u> . (Front-end for CVS) 4) <u>BitKeeper</u> . [Prop] 5) <u>SubVersion</u> . (enhanced CVS-like platform + WebDAV -> SCM)
IDE for Interbase/Firebird development	IBExpert	1) <u>IBAccess</u> 2) <u>IBAdmin</u> [prop] 3) <u>IBWebAdmin</u> (apache / php)
Visual Basic	Visual Basic	<u>Phoenix</u> .
Graphical libraries	1) WinAPI, MFC, VCL (C, C++) 2) <u>Tk</u> (Tcl, C) 3) <u>Tkinter</u> (over Tk for Python, Perl, etc) 4) <u>wxWindows</u> (C++) (over winapi) 5) <u>wxPython</u> (Python) (over wxWindows) 6) <u>GTK+</u> (link #2) (C, C++) 7) <u>Qt</u> (C++) 8) <u>FLTK</u> (C++) 9) AWT, Swing (Java) 10) Xaw - part of X-server (C)	If toolkit is oriented on C, usually it's possible to use it from the other languages. There is object bindings for some of them to use them with C++ and other OO-languages. Toolkits, oriented on C++ from the beginning, are impossible to use from C, and quite often - from the other languages. 1) X11/Xtext (C) - low level libraries, used by others. 2) Xt - X Toolkit (C) - reference X11 toolkit 3) Xaw - MIT Athena (C) - reference X11 toolkit 4) <u>Xaw3d</u> - MIT Athena <u>3D</u> (C) - Athena with 3D-view 5) <u>LessTif</u> - opensource analog of Motif 1 (C) 6) <u>OpenMotif</u> (C) 7) <u>Tk</u> (Tcl, C) 8) <u>Tkinter</u> (over Tk for Python, Perl, etc) 9) <u>wxWindows</u> (C++) (over GTK+ - wxGtk or Motif - wxMotif) 10) <u>wxPython</u> (Python) (over wxWindows) 11) <u>Ot</u> (C++)

		12) <u>GTK+</u> (C, C++) 13) <u>PyQt</u> (over Qt for Python) 14) <u>PyGTK</u> (over GTK+ for Python) 15) <u>Gtk::Perl</u> (over GTK+ for Perl) 16) Qt for Perl (over Qt for Perl) 17) <u>GtkAda</u> (over GTK+ for Ada95) 18) <u>FLTK</u> (C++) 19) <u>XView</u> (C) 20) <u>FOX</u> (C++) 21) AWT, Swing (Java) 22) <u>WinAPI and MFC through wine</u> (C)
Source code documentation system	???	1) <u>Doxygen</u> . 2) <u>CWEB</u> .
Memory leak tracing	1) Numega Bounds Checker 2) Rational Purify	1) <u>MallocDebug</u> . 2) <u>Valgrind</u> . 3) <u>Kcachegrind</u> . 4) <u>ElectricFence</u> . 5) <u>dmalloc</u> . 6) <u>ccmalloc</u> . 7) <u>LeakTracer</u> . 8) <u>memprof</u> . 9) <u>BoundsChecker</u> . [prop] 10) <u>mprof</u> . 11) <u>Insure</u> . [prop] 12) <u>dbx</u> . (for Sparc) 13) <u>YAMD</u> . 14) <u>Njamd</u> . 15) <u>Mpatrol</u> .
Application development profiling (tests code performance)	???	1) <u>gProf</u> . 2) <u>JUnit</u> . (Java)
Software projecting	1) Rational Rose 2) Enterprise Architect 3) Visio	???
Game programming	1) <u>DirectX</u> 2) <u>libSDL</u> 3) <u>ClanLib</u>	1) <u>libSDL</u> 2) <u>ClanLib</u>
Everything needed for work with XML Schemas. DTD.	1) Altova XMLSpy Suite	???

XSL/XSLT, SOAP, WSDL (edit, debug, check, etc)		
Source code -> HTML-document with highlighted syntax	???	1) <u>Webcpp</u> .
Bug Tracking System	???	1) <u>Bugzilla</u> .
Object Request Broker (ORB)	1) <u>ORBIT</u>	1) <u>ORBIT</u> .
Portability	???	1) <u>Autoconf</u> , <u>Automake</u> & <u>Libtool</u> .
Source code indexer and cross-referencer	???	1) <u>GNU GLOBAL</u> . 2) <u>LXR</u> .
Dynamic tracer of system calls	???	1) <u>Syscalltrack</u> .
XML C parser	1) <u>libxml</u> (???)	1) <u>Libxml2</u> .
8) Server software.		
Web-server	1) <u>Apache</u> 2) <u>IIS</u> 3) <u>Roxen</u> 4) <u>wn</u> 5) <u>cern-httpd</u> 6) <u>dhttpd</u> 7) <u>caudium</u> 8) <u>aolserver</u> 9) <u>Boa</u>	1) <u>Apache</u> . 2) <u>Xitami</u> . 3) <u>Thttp</u> . 4) <u>TUX (Red Hat Content Accelerator)</u> . 5) <u>PublicFile</u> . 6) <u>Boa</u> . 7) <u>Caudium</u> . 8) <u>Roxen</u> . 9) <u>Zeus</u> . [Prop] 10) <u>Thy</u> .
FTP-server	Internet Information Server, <u>ServU</u> , War FTP, BulletProof FTP server	1) <u>pure-ftpd</u> . 2) <u>vsftpd</u> . 3) <u>wu-ftpd</u> . 4) <u>proftpd</u> . 5) <u>gl-ftpd</u> . 6) <u>ftp</u> . 7) <u>PublicFile</u> . 8) <u>Teepeedee</u> .
Language for Web-development	PHP	<u>PHP</u> .
Language for Web-development	Perl	<u>Perl</u> .
Language for Web-development	ASP	ASP module for Apache.

Database engine	MS SQL, <u>MySQL</u>	1) <u>Sybase Adaptive Server Enterprise</u> . [Prop] 2) <u>PostgreSQL</u> . The most advanced open source database. 3) <u>MySQL</u> . The most popular open source database. 4) <u>mSQL</u> . 5) <u>SAP DB</u> .
Database engine	IBM DB2	<u>IBM DB2</u> . [Prop]
Database engine	Oracle	1) <u>Oracle</u> . [Prop] 2) <u>PostgreSQL</u> . 3) <u>Linter</u> . (cyrillic)
Database engine	Informix [Prop]	Informix. [Prop]
Database engine	Borland Interbase, <u>FireBird</u>	<u>FireBird</u> .
Email server	<u>MDaemon</u> , Hamster	1) <u>Sendmail</u> . 2) <u>Qmail</u> . 3) <u>Postfix</u> . 4) <u>Exim</u> .
Email / PIM / Groupware server	Microsoft Exchange	1) <u>CommuniGate Pro</u> . [Prop] 2) <u>Bynari's Insight GroupWare Suite</u> . [Prop] 3) <u>Samsung Contact</u> . [Prop] 4) <u>Teamware Office</u> . [Prop] 5) <u>Novell Netmail</u> . [Prop] 6) <u>Amphora</u> . Based on zope and qmail. 7) <u>Tutos</u> . Based on Linux / Apache / PHP / Mysql / Sendmail. 8) <u>Kroupware</u> . The project from the KDE PIM developers, which is being financed by the government of Germany. 9) <u>SuSe Linux Openexchange Server</u> . [Prop] 10) <u>PHPGroupware</u> . 11) <u>SCOoffice Mail Server</u> . [Prop] (SCO - m.d. :). 12) <u>LinuXchangE</u> . 13) <u>OpenOffice.org Groupware Project</u> . (New!)
Mail filter / spam killer	cygwin+Exim port	1) <u>SpamAssassin</u> . 2) <u>Procmail</u> . 3) <u>Mailfilter</u> . 4) яvrus-imap.

		5) Exim.
Mail downloader	MDaemon	<u>Fetchmail</u> .
???	Lotus Domino	<u>Lotus Domino</u> . [Prop]
Server / router on one diskette.	Impossible	As much as you want :). 1) <u>muLinux</u> . 2) <u>Dachstein</u> (firewall / dhcp). 3) <u>Serverdisk</u> (http / ftp). 4) <u>Fl4!</u> .
Proxy server	MS Proxy Server, WinGate	1) <u>Squid</u> . 2) <u>Paco</u> . 3) <u>Privoxy</u> . 4) <u>Wwwoffle</u> . 5) <u>OOPS</u> .
Server for supporting Java Servlets and JSP, can work with Apache	<u>Tomcat</u>	<u>Tomcat</u> .
Advanced server statistics	1) <u>AWStats</u>	1) <u>AWStats</u> . (All web-, ftp-, proxy-, mail-, wap- and streaming-servers). 2) <u>ANALOG</u> . (Web-server).
Cluster of servers	???	1) <u>LVS - The Linux Virtual Server</u> .
Web Mail	???	1) <u>IMP</u> . 2) <u>CAMAS</u> . (for Caudium web-server).

9) Scientific and special programs.

Useful links:	-	<u>Scientific Applications on Linux</u> - many links to both OSS and proprietary applications.
Math system in MathCad style	Mathcad	Gap.
Math system in Matlab style	<u>Matlab</u>	1) <u>Matlab</u> . [FTP] 2) <u>Octave</u> . (+ Gnuplot) 3) <u>Scilab</u> . 4) <u>R</u> . 5) <u>Yorick</u> . 6) <u>rlab</u> . 7) <u>Yacas</u> . 8) <u>Euler</u> .
Math system in Mathematica style	<u>Mathematica</u>	1) <u>Mathematica</u> . [Prop] 2) <u>Maxima</u> . 3) <u>MuPad</u> .

		4) NumExp. 5) <u>Mathomatic</u> .
Math system in Maple style	Maple	1) <u>Maple</u> . [Prop] 2) <u>Maxima</u> . 3) <u>MuPad</u> .
Equation / math editor	Mathtype, MS Equation Editor, <u>OpenOffice Math</u>	1) <u>OpenOffice Math</u> . 2) <u>MathMLed</u> . 3) <u>Kformula (Koffice)</u> . 4) <u>LyX</u> . 5) <u>Texmacs</u> .
Programs for three-dimensional modeling	SolidWorks, ...	<u>ProEngineer Linux</u> . [Prop]
Programs for three-dimensional modeling	CATIA	CATIA. It was designed under Unix, and from version 4 (2000) it was ported under Windows (not too successfully).
Programs for three-dimensional modeling	SolidEdge	SolidEdge (part of more powerful package Unigraphics).
Engineering	ANSYS	ANSYS.
CAD/CAM/CAE	AutoCAD, Microstation, ArchiCAD	1) <u>Varkon</u> . 2) <u>Linuxcad</u> . [Prop, ~100\$] 3) <u>Varicad</u> . [Prop] 4) <u>Cycas</u> . [Prop] 5) <u>Tomcad</u> . 6) <u>Thancad</u> . 7) <u>Fandango</u> . 8) <u>Lignumcad</u> . 9) <u>Giram</u> . 10) <u>Jcad</u> . 11) <u>QSCad</u> . 12) <u>FreeEngineer</u> . 13) <u>Ocadis</u> . 14) <u>PythonCAD</u> . 15) <u>OpenCascade</u> .
CAD/CAM/CAE, simplified	AutoCAD Lite	<u>Qcad</u> .
Desktop Publishing Systems	Adobe PageMaker, QuarkXPress	<u>Adobe Framemaker</u> . [Proprietary, cancelled]
Small desktop publishing systems	MS Publisher	1) <u>Scribus - Desktop Publishing for Linux</u> . 2) <u>KWord</u> .
Diagram and chart	Microsoft Visio	1) <u>Kivio (Koffice)</u> .

designer		2) Dia . 3) KChart . 4) xfig . 5) Tgif + dotty . 6) Tulip . 7) Poseidon for UML . [Prop & free versions]
Geographic image processing software	Erdas Imagine, ER Mapper , ENVI	ENVI .
GIS (Geographical information system)	ArcView	All projects: FreeGIS Project . 1) Grass . 2) Quantum GIS . 3) PostGIS . 4) FreeGIS . 5) MapQuest . 6) MapBlast .
Interactive Geographic Data Viewer	1) Thuban .	1) Thuban .
Vectorization of bitmaps	MapEdit , Easy Trace	1) Autotrace .
Software CNC, controlling machine tools	OpenCNC [Prop]	EMC .
Advanced text processing system in TeX style	MikTeX , emTeX (DOS)	1) TeX . 2) TeX / LaTeX 3) LyX (WYSIWYM). 4) Kile .
Convenient, functional and user-friendly TeX-files / dvi-files editor.	WinEdt	1) Kile (KDE Integrated LaTeX Environment). 2) Ktexmaker2 . 3) Tk LaTeX Editor .
Statistical Computing Language and Environment	S-PLUS	R .
Statistical analysis	SPSS , Statistica , SalStat	Many links - here . 1) PSPP . 2) OpenStat2 . 3) "Probability and Statistics Utilities for Linux users" 4) SalStat .

Econometrics Software	Eviews, Gretl	1) <u>Gretl</u> .
Emulation of the circuit	1) Electronic Workbench 2) Altera MaxPlus+	1) <u>Geda</u> . 2) <u>Oregano</u> . 3) <u>Xcircuit</u> . 4) <u>Gnome Assisted Electronics</u> . 5) <u>SPICE</u> . 6) <u>SPICE OPUS</u> . 7) <u>NG-SPICE</u> .
Program to draw chemical structures	<u>Chemdraw</u> , <u>Isisdraw</u>	<u>Xdrawchem</u> .
Downloader and player for Olympus dictophone	Olympus DSS Player	???
Market analysis	<u>MetaStock</u>	???
Electronics scheme design	1) PCAD 2) OrCad 3) Visio	1) <u>Eagle</u> . 2) <u>Geda</u> .
The oscilloscope emulation	Winoscillo	<u>Xoscope</u> .
Measurement of the temperature and voltages on motherboard	MBMonitor, PCAAlert, Speedfan	1) <u>KSensors</u> . 2) <u>KHealthCare</u> (KDE). 3) <u>Gkrellm</u> + plugins + blackbox addons.
S.M.A.R.T-attributes and temperature of the hard disk	Come on CD with mainboard, <u>Active SMART</u>	1) <u>smartctl</u> . 2) <u>Hddtemp-0.3</u> . 3) <u>IDEload-0.2</u> . 4) <u>Smartsuite-2.1</u> . 5) <u>Smartmontools</u> . 6) <u>lde-smart</u> . 7) <u>Smartsuite</u> .
Memory testing	SiSoft SANDRA	<u>Memtest86</u> .
Program for watching temperatures, fanspeeds, etc	SiSoft SANDRA, SiSoft SAMANTHA	1) <u>Ksensors</u> . 2) <u>Lm_sensors</u> . 3) <u>xsensors</u> . 4) <u>wmsensormon</u> and other applets for AfterStep / WindowMaker / FluxBox.
HDD testing / benchmarking	SiSoft SANDRA, SiSoft SAMANTHA, <u>IOzone</u>	1) <u>hdparm</u> . 2) <u>Bonnie++</u> . 3) <u>IOzone</u> . 4) <u>Dhbench</u> .

		5) <u>Bonnie</u> . 6) <u>IO Bench</u> . 7) <u>Nhfsstone</u> . 8) <u>SPEC SFS</u> . [Prop]
Video testing / benchmarking	Final Reality	1) <u>X11perf</u> . 2) <u>Viewperf</u> .
Realtime Control	<u>SHA Sybera</u> <u>Hardware Access</u>	<u>DIAPM RTAI - Realtime Application Interface</u> .
Simulator of nets	???	1) <u>NS</u> .
Neural network simulation	???	1) <u>Xnbc</u> . 2) <u>Stuttgart Neural Network Simulator (SNNS)</u> .
"Sensor for LCD"	???	1) <u>Sensors-lcd</u> .
Electrocardiogrammas viewer	???	1) <u>ecg2png</u> .
A software technology, that turns x86 computer into a full-function PLC-like process controller	<u>SoftPLC</u>	1) <u>MatPLC</u> .
Catalog of the software for translators	-	<u>Linux for translators</u> .
Translation memory	1) <u>Trados Translators Workbench</u> 2) <u>Deja Vu</u> 3) <u>Star Transit</u> 4) <u>SDLX</u> 5) <u>OmegaT</u>	1) <u>OmegaT</u> .
Catalog of educational software	-	1) <u>SchoolForge</u> . 2) <u>Seul / EDU</u> .
Designing and viewing DTDs	<u>NearFar Designer</u> [Prop]	???
Finity Element Analysis	-	1) <u>FELT (Finity Element Analysis)</u>
10) Emulators.		
Virtual machine emulator	1) <u>VMWare</u> [Prop] 2) <u>Connectix Virtual</u>	1) <u>VMWare</u> . [Prop] 2) <u>Win4Lin</u> . IPron. \$891.

	PC [Prop]	3) Bochs . 4) Plex86 . 5) User Mode Linux .
Linux emulator	1) CygWin . 2) MKS Toolkit. 3) Bash for Windows. 3) Minimalist GNU For Windows .	1) User Mode Linux .
X Window System (XFree) emulator	XFree under CygWin .	-
Windows emulator	-	1) Wine . (GUI: gwine , tkwine) 2) Transgaming WineX . (GUI: tqgui) [NF] 3) Crossover Office .
Sony PlayStation emulator	ePSXe, ...	1) ePSXe . 2) Pcsx .
ZX Spectrum emulator	X128, Speccyal, SpecX, SpecEmu, UnrealSpeccy, ...	1) Xzx . 2) Glukalka . 3) Fuse . 4) ZXSP-X . 5) FBZX . 6) SpectEmu .
Arcade machines emulator	???	1) MAME . 2) Xmame / Xmess . 3) Advancemame . Frontends: advancemenu . ckmame . flynn . gname . gnomame . grok . grustibus . gxname . it . it's quit . fancy . kmamerun . kmamu . qmamecat . startxmame . setcleaner . tkmame .
ST emulator	1) Steem .	1) StonX . 2) Steem .
C64 emulator	???	1) Vice . 2) Frodo .
Amiga emulator	???	1) UAE . 2) WinUAE .
Mac 68k emulator	???	1) Basilisk II .
Game boy emulator	1) Visual Boy Advance	1) Visual Boy Advance . 2) VGBA . (GUI: vgb-gui)
Atari 2600 Video	1) Stella	1) Stella .

Computer System emulator		2) Saint .
NES / SNES emulator	1) Zsnes . 2) Snes9x .	1) Zsnes . 2) Snes9x . 3) FWNes . 4) GTuxNes .
M680x0 Arcade emulator	1) Rainemu .	1) Rainemu .
Multi / other emulators	???	1) M.E.S.S . 2) Zinc .
11) Other / Humour :) 		
Space simulator	1) Openuniverse . 2) Celestia . 3) Zetadeck .	1) Openuniverse . 2) Celestia . 3) Kstars . 4) Zetadeck .
TV driver	-	RivaTV .
System, running from CD without installing (Live CD)	1) Windows PE . 2) PE Builder .	1) Knoppix . 2) Cool Linux . 3) Blin . 4) DemoLinux . 5) DyneBolic . 6) Gentoo (live CD) . 7) Lonix . 8) Virtual Linux . 9) Bootable Business Card (LNX-BBC) . 10) ByzantineOS . 11) FreeLoader Linux . 12) MoviX . 13) Freeduc CD . 14) SuSE live-eval CD . 15) Freedom Linux . 16) Eagle Linux . 17) Kurumin (Brazilian Portuguese only)
Boot rescue/tools diskette	Windows system diskette	1) Linux system diskette. 2) Tomsrtbt . 3) BanShee Linux . 4) RIP .
Creation of LiveCD for system recovery	???	1) Make CD-ROM Recovery .
Local file systems mount	ext2fs (driver), explore2fs	Linux-NTFS . (driver for NTFS partitions mounting)

	(program) - ext2/3 under Windows	
Installing software and uninstalling	InstallShield, WISE, GhostInstaller, Microsoft Installer - the analog of rpm	1) Rpm. 2) Urpmi. 3) GnoRpm. 4) Nautilus RPM. 5) Apt-get & frontends (synaptic, aptitude, ...). 6) Apt-rpm. (for RedHat, SuSE, ALT Linux, etc) 7) yum (Yellowdog Updater Modified) 8) yum enhanced by ASPLinux. 9) Gentoo Portage
Installing software from source and uninstalling	<u>Minimalist GNU For Windows</u>	1) make install, make uninstall 2) CheckInstall. 3) Sinstall. 4) Emerge (Gentoo). 5) Apt-get & frontends (synaptic, aptitude, ...).
System update	Windows Update	1) Ximian Red Carpet. 2) Red Hat Network. 3) MandrakeOnline. 4) SuSE YaST Online Update. 5) Caldera Volution Online. 6) Apt. 7) Gentoo ebuilds (portage). 8) Debian GNU/Linux package search. 9) Yum.
Certification	MCSD, MCT, MCSE	1) Red Hat Certification. 2) Sair Linux and GNU Certification. 3) Linux Professional Institute Certification (LPIC). 4) Linux+. 5) Prometric. 6) VUE.
Icons on desktop	Explorer	1) Desktop File Manager. 2) Idesk.
Work with screensavers	Desktop properties	1) xset. 2) xlockmore. 3) xscreensaver. 4) ksscreensaver.
Place for keeping "removed" files	Trash	1) Trash Can. 2) Libtrash.

Checking the hard disk	Scandisk	fsck -check or reiserfsck -check. Not needed with journaled file systems (reiserfs, ext3, jfs, xfs).
Defragmentation	defrag	Not needed.
GUI of the system	Windows Explorer	Kde, Gnome, IceWM, Windowmaker, Blackbox, Fluxbox, ...
Windows XP GUI	Windows XP	<u>XPde</u> .
Multiple workspaces	-	Yes!!! :).
Fast users / desktop switching	Windows XP feature for non- networked computers	1) Ctrl+Alt+F1, login as new user at command-line interface, start GUI by entering command startx - - :1. Switch between screens using Ctrl+Alt+F7 or F8 depending on user. 2) Command "gdmflexiserver -n". 3) Built-in in <u>KDE 3.1</u> .
Flavors of the system	9x, NT, XP	RedHat, Mandrake, Knoppix, Debian, SuSE, ALT, ASP, Gentoo, Slackware, Linux From Scratch, ...
Tactics	FUD (fear, uncertainty, doubt)	Open Source! "First they ignore you, then they laugh at you, then they fight you, then you win".
Source code of the kernel freely available	No	<u>Of course :)</u>
Command line and scripting	1) <u>command.com</u> :). 2) <u>cmd.exe</u> 3) <u>Windows Scripting Host</u> 4) <u>4DOS / 4NT</u> 5) <u>Minimalist GNU For Windows</u> 6) <u>Unix tools for Windows (AT&T)</u> 7) <u>KiXtart</u> 8) <u>ScriptLogic</u> [Prop]	1) <u>Bash</u> . 2) <u>Csh</u> . 3) <u>Zsh</u> . 4) <u>Ash</u> . 5) <u>Tcsh</u> .
Free of charge operating system	Microsoft Windows. (Imagine yourself that in Russia there are 95% of users having a pirate copy of Windows :).	Linux - the Free operating system!!
-	<u>Nimda</u>	<u>Slapper</u> .

-	Wincih, klez, etc	No analogs
Backdoors and hidden keys	Decide it yourself :).	-
Easter eggs, undocumented possibilities	Logo with Windows developers, Doom in Excel 95, 3D-racing in Excel 2000, etc, etc...	-
The magazines	Windows Magazine	1) Linux Journal . 2) Linux Gazette . 3) Linux magazine . 4) Linux pratico (Italy) . 5) Australian Linux .
-	Blue Screen Of Death (BSOD)	1) Kernel panic. 2) Screensaver "bsod" :).
Whom it is necessary to curse for bugs and defects of the system	M\$, Bill Gates personally	1) Developers of the distribution. 2) All the Linux people and Linus Torvalds personally :). 3) Yourself and your own /dev/hands :)).
-	M\$.com	GNU.org , FSF.org
-	Windows.com	Linux.org
-	Bill Gates, "Road ahead"	Linus Torvalds, "Just for fun" :).
-	Bill Gates, "Business @ the speed of thought"	Richard M. Stallman, "The right to read". The book: "Free Software, Free Society: Selected Essays of Richard M. Stallman"